Monographs in Electrochemistry
*Series Editor:* F. Scholz

Dieter Britz
Jörg Strutwolf

# Digital Simulation in Electrochemistry

*Fourth Edition*

EXTRAS ONLINE

Springer

# Digital Simulation in Electrochemistry

# Monographs in Electrochemistry

**Series Editor**: Fritz Scholz, University of Greifswald, Germany

Surprisingly, a large number of important topics in electrochemistry is not covered by Up-to-date monographs and series on the market, some topics are even not covered at all. The series Monographs in Electrochemistry fills this gap by publishing indepth monographs written by experienced and distinguished electrochemists, covering both theory and applications. The focus is set on existing as well as emerging methods for researchers, engineers, and practitioners active in the many and often interdisciplinary fields, where electrochemistry plays a key role. These fields will range – among others – from analytical and environmental sciences to sensors, materials sciences and biochemical research.

More information about this series at http://www.springer.com/series/7386

Dieter Britz • Jörg Strutwolf

# Digital Simulation in Electrochemistry

Fourth Edition

Springer

Dieter Britz
Aarhus University
Aarhus, Denmark

Jörg Strutwolf
Department of Chemistry
University of Tübingen
Tübingen, Germany

Additional material to this book can be downloaded from http://extras.springer.com.

ISSN 1865-1836 ISSN 1865-1844 (electronic)
Monographs in Electrochemistry
ISBN 978-3-319-30290-4 ISBN 978-3-319-30292-8 (eBook)
DOI 10.1007/978-3-319-30292-8

Library of Congress Control Number: 2016936724

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG Switzerland

*This book is dedicated to our wives,
Sandra and Nooshin.*

The original version of this book was revised. An erratum to this book can be found at DOI 10.1007/978-3-319-30292-8_18

# Preface

This book is a revision of the 2005 3rd Edition with the same title. There is additional material, mainly in the entirely new Chap. 13 on migration and substantial new parts on ultramicroelectrode arrays in Chap. 12, as well as a general overhaul of all other chapters including many references to more recent work.

It has been our aim since the 1st Edition to write on the subject in a didactic manner, and we believe we have succeeded. The first chapters provide the reader with the background needed for the programming of simulations. There are still very few books in this field. Another book on the same subject has recently appeared [1], emphasizing cyclic voltammetry, although other experimental techniques are mentioned. The interesting book [2] by Honeychurch, very wide-ranging, presenting Mathematica [3] solutions to simulation, is now out of print and hard to obtain and in any case serves a different purpose, programming in that environment, whereas we continue to propagate a compiled language, in our case Fortran 95 (or higher).

The book is accompanied by a number of example procedures and programs, all in Fortran 90/95 (available at http://extras.springer.com). These have all been verified as far as possible. While some errors might remain, they are hopefully very few and mainly cosmetic.

We gratefully acknowledge a number of colleagues for fruitful discussions. They are (in alphabetical order) Dr. Lesław Bieniasz (Cracow University of Technology), Dr. Stephen Feldberg (Brookhaven National Laboratory, Upton, New York), Professor Bertel Kastening (Hamburg University), Professor J.A. Manzanares (Valencia University), Professor Keith Oldham (Trent University Canada), Dr. Ole Østerby (Aarhus University), and Professor Bernd Speiser (Tübingen University). If we have left anybody out, we apologize.

Finally, one of us (DB) thanks his wife for mathematical support, and we both thank our wives for their patience during the writing of this book.

Aarhus, Denmark
Tübingen, Germany
January 2016

Dieter Britz
Jörg Strutwolf

## References

1. Compton RG, Laborda E, Ward KR (2014) Understanding voltammetry: simulation of electrode processes. Imperial College Press, London
2. Honeychurch MJ (2004–2006) Simulating electrochemical reactions with mathematica. IBNH, St. Lucia, QLD, Australia
3. https://www.wolfram.com/mathematica/

# Contents

# Chapter 1
# Introduction

This book is about the application of digital simulation to electrochemical problems. The term "simulation" came into wide use with the advent of analog computers, which could produce electrical signals that followed mathematical functions to describe or model a given physical system [1–3], and there was even a digital simulator of an analog control circuit for an electrochemical simulation [4]. When digital computers became common, people began to do these simulations digitally and called this digital simulation. Most commonly we simulate electrochemical transport problems, which are difficult to solve analytically in all but a few model system cases—when things get more complicated, as they do in real electrochemical cells, problems may not be solvable algebraically, yet we still want answers.

Most commonly, the basic equation we need to solve is the diffusion equation, relating concentration $c$ to time $t$ and distance x from the electrode surface, given the diffusion coefficient D:

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} \; . \tag{1.1}$$

This is Fick's second diffusion equation [5], an adaptation to diffusion of the heat transfer equation of Fourier [6]. Technically, it is a second-order parabolic partial differential equation (*pde*). In fact, it will mostly be only the skeleton of the actual equation one needs to solve; there will usually be such complications as convection (solution moving), migration (ion movement in solution due to an electric field) and chemical reactions taking place in the solution, which will cause concentration changes in addition to diffusion itself. Numerical solution may then be the only way we can get numbers from such equations—hence digital simulation.

The numerical technique most commonly employed in digital simulation is (broadly speaking) that of finite differences and this is much older than the digital computer. It dates back at least to 1911 [7] (Richardson). In 1928, Courant et al. [8]

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_1

described what we now take to be the essentials of the method; Emmons [9] wrote a detailed description of finite difference methods in 1944, applied to several different equation types. There is no shortage of mathematical texts on the subject: see, for example, Lapidus and Pinder [10] and Smith [11], two excellent books out of a large number.

It should not be imagined that the technique became used only when digital computers appeared; engineers certainly used it long before that time, and were not afraid to spend hours with pencil and paper. Emmons [9] casually mentions that one fluid flow problem took him 36 h! Not surprisingly, it was during this early pre-computer era that much of the theoretical groundwork was laid and refinements worked out to make the work easier—those early stalwarts wanted their answers as quickly as possible and accurate.

Electrochemical digital simulation is almost synonymous with Stephen Feldberg, who wrote his first paper on it in 1964 [12]. It is not always remembered that Randles [13] used the technique much earlier (in 1948), to solve the linear sweep problem. He did not have a computer and did the arithmetic by hand. The most widely quoted electrochemical literature source is Feldberg's chapter in Electroanalytical Chemistry [14], which describes what will here be called the "box" method. Feldberg is rightly regarded as the pioneer of digital simulation in electrochemistry, and is still active in the field today. This has also meant that the box method has become standard practice among many electrochemists, while what will here be called the "point" method is more or less standard elsewhere. Having experimented with both, the present authors favour the point method for the ease with which one arrives at the discrete form of one's equations, especially when the differential equation is complicated.

A brief description will now be given of the essentials of the simulation technique. Assume (1.1) above. We wish to obtain concentration values at a given time over a range of distances from the electrode. We divide space (the $x$ coordinate) into small intervals of length $h$ and time $t$ into small time steps $\delta t$. Both $x$ and $t$ can then be expressed as multiples of $h$ and $\delta t$, using $i$ as the index along $x$ and $j$ as that for $t$, so that

$$x_i = ih \tag{1.2}$$

and

$$t_j = j\delta t \; . \tag{1.3}$$

Figure 1.1 shows the resulting grid of points. At each drawn point, there is a value of $c$. The digital simulation method now consists of developing rows of $c$ values along $x$, (usually) one $t$-step at a time. Let us focus on the three filled-circle points $c_{i-1}$, $c_i$ and $c_{i+1}$ at time $t_j$. One of the various techniques to be described will compute from these three known points a new concentration value $c'_i = c_i(t =$

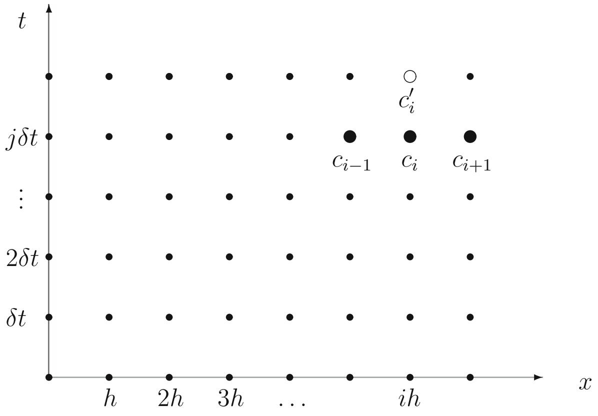


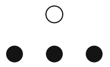
**Fig. 1.1** Discrete sample point grid

$(j + 1)\delta t$) (empty circle) at $x_i$ for the next time value $t_{j+1}$, by expressing (1.1) in discrete form:

$$\frac{c_i' - c_i}{\delta t} = \frac{D}{h^2}\left(c_{i-1} - 2c_i + c_{i+1}\right). \tag{1.4}$$

The only unknown in this equation is $c_i'$ and it can be explicitly calculated. Having obtained $c_i'$, we move on to the next $x$ point and compute $c'$ for it, etc., until all $c$ values for that row, for the next time value, have been computed.

In the remainder of the book, the various schemes for calculating new points will often be graphically described by isolating the marked circles seen in Fig. 1.1; in this case, the scheme would be represented by the following diagram



This follows the convention seen in such texts as Lapidus and Pinder [10] (who call it the "computational molecule"). It is very convenient, as one can see at a glance what a particular scheme does. The filled points are known points while the empty circles are those to be calculated.

Several problems will become apparent. The first one is that of the method used to arrive at (1.4); this will be dealt with later. There is, in fact, a multiplicity of methods and expressions used. The second problem is the concentration value at $x = 0$; there is no $x_{-1}$ point, as would be needed for $i = 0$. The value of $c_0$ is a *boundary value*, and must be handled in some other way. Another boundary value is the last $x$ point we treat. How far out into the diffusion space should (need) we go? Usually, we know good approximations for concentrations at some sufficiently large distance from the electrode (e.g. either "bulk" concentration, or zero for a species generated at the electrode), and we have pretty good criteria for the distance we need to go out to. Another boundary lies at the row for $t = 0$: this is the row of starting values. Again, these are supplied by information other than the diffusional process we are simulating. Boundary problems are dealt with in Chap. 6. They are, in fact, a large part of what this book is about, or what makes it specific to electrochemistry. The discrete diffusion equation we have just gone through could just as well apply to heat transfer or any other diffusional transport problems.

Throughout the book, the following symbol convention will be used: dimensioned quantities like concentration, distance or time will be given lower-case symbols ($c$, $x$, $t$, etc.) and their non-dimensional equivalents will be given the corresponding upper-case symbols ($C$, $X$, $T$, etc.), with a few unavoidable exceptions.

## References

1. Bucur RV, Covaci I, Miron C (1964) Die Modellierung der Elektrolyse bei konstantem Strom im Falle ebener, endlicher Diffusion mit Hilfe des Analogrechners. J Electroanal Chem 8:277–285
2. Holub K, Němec L (1966) The analog method for solution of problems involving diffusion to the electrode. I. Diffusion to a sphere with Langmurian adsorption solved by analog method. J Electroanal Chem 11:1–11
3. Kabakchi SA, Filinovskii VY (1972) Simulation of convection diffusion processes close to a rotating disk electrode by means of a Model MN-7 analog computer. Sov Electrochem 8:1390–1394
4. Klinger J, Conway BE, Angerstein-Kozlowska H (1978) Procedures for computer simulation of kinetics of electrochemical surface processes. Comput Chem 2:117–129
5. Fick A (1855) Ueber diffusion. Pogg Ann 94:59–86
6. Fourier F (1822) Théorie Analytique de la Chaleur, vol I. Didot, Pere et Fils, Paris
7. Richardson LF (1911) The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. Philos Trans R Soc Lond Ser A 210:307–357
8. Courant R, Friedrichs K, Lewy H (1928) Über die partiellen Differenzengleichungen der mathematischen Physik. Math Ann 100:32–74
9. Emmons HW (1944) The numerical solution of partial differential equations. Quart Appl Math 2:173–195
10. Lapidus L, Pinder GF (1982) Numerical solution of partial differential equations in science and engineering. Wiley, New York
11. Smith GD (1985) Numerical solution of partial differential equations, 3rd edn. Oxford University Press, Oxford
12. Feldberg SW, Auerbach C (1964) Model for current reversal chronopotentiometry with second-order kinetic complications. Anal Chem 36:505–509
13. Randles JEB (1948) A cathode-ray polarograph. Part II - the current-voltage curves. Trans Faraday Soc 44:327–338
14. Feldberg SW (1969) Digital simulation: a general method for solving electrochemical diffusion-kinetic problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 3. Marcel Dekker, New York, pp 199–296

# Chapter 2
# Basic Equations

## 2.1 General

In this chapter, we present most of the equations that apply to the systems and processes to be dealt with later. Most of these are expressed as equations of concentration dynamics, that is, concentration of one or more species as a function of time, as well as other variables, in the form of differential equations. Fundamentally, these are transport (diffusion-, convection- and migration-) equations but may be complicated by chemical processes occurring heterogeneously (i.e. at the electrode surface—electrochemical reaction) or homogeneously (in the solution bulk—chemical reaction). The transport components are all included in the general Nernst–Planck equation (see also [1]) for the flux $J_j$ of species $j$

$$\boldsymbol{J}_j = -D_j \nabla c_j - \frac{z_j \mathcal{F}}{\mathcal{R}\mathcal{T}} D_j c_j \nabla \phi + c_j \boldsymbol{v} \tag{2.1}$$

in which $\boldsymbol{J}_j$ is the molar flux per unit area of species $j$ at the given point in space, $D_j$ the species' diffusion coefficient, $c_j$ its concentration, $z_j$ its charge, $\mathcal{F}$, $\mathcal{R}$ and $\mathcal{T}$ have their usual meanings, that is, respectively the Faraday (universal) gas constant and temperature, $\phi$ is the potential and $\boldsymbol{v}$ the fluid velocity vector of the surrounding solution (medium). The symbol $\nabla$ denotes the differentiation operator and it is directional in three-dimensional (3D) space. This equation is a more general form of Fick's first diffusion equation, which contains only the first term on the right-hand side, the diffusion term. The second term on that side is the migration term and the last is the convection term. These will now be discussed individually. At the end of the chapter, we go through some models and electrode geometries, and present some known analytical solutions, as well as dimensionless forms of the equations. There is no term in the equation to take account of changes due to chemical reactions taking place in the solution, since these do not directly give rise to a flux of substance. Such

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_2

terms come in later, in the equations relating concentration changes with time to the above components (see (2.15) and Sect. 2.2.6).

## 2.2 Some Mathematics: Transport Equations

### *2.2.1 Diffusion*

For a good text on diffusion, see the monograph of Crank [2], or Ghez [3]. Consider Fig. 2.1. We imagine a chosen coordinate direction $x$ in a solution volume containing a dissolved substance at concentration $c$, which may be different at different points—i.e., there may be concentration gradients in the solution. We consider a very small area $\delta A$ on a plane normal to the $x$-axis. Fick's first equation now says that the net flow of solute (flux $f_x$, in mol s$^{-1}$) crossing the area is proportional to the negative concentration gradient at the plane, in the $x$-direction

$$f_x = \frac{\mathrm{d}n}{\mathrm{d}t} = -\delta A\, D \frac{\mathrm{d}c}{\mathrm{d}x} \tag{2.2}$$

with $D$ a proportionality constant called the diffusion coefficient and $n$ the number of moles. This can easily be understood upon a moment's thought; statistically, diffusion is a steady spreading out of randomly moving particles. If there is no concentration gradient, there will be an equal number per unit time moving backward and forward across the area $\delta A$, and thus no net flow. If there is a gradient, there will be correspondingly more particles going in one direction (down the gradient) and a net increase in concentration on the lower side will result. Equation (2.2) is of precisely the same form as the first heat flow equation of Fourier [4]; Fick's contribution [5] lay in realising the analogy between temperature and concentration, heat and mass (or number of particles). The quantity $D$ has units m$^2$ s$^{-1}$ (SI) or cm$^2$ s$^{-1}$ (cgs).

Equation (2.2) is the only equation needed when using the box method and this is sometimes cited as an advantage. It brings one close to the microscopic system, as we shall see, and has—in theory—great flexibility in cases where the

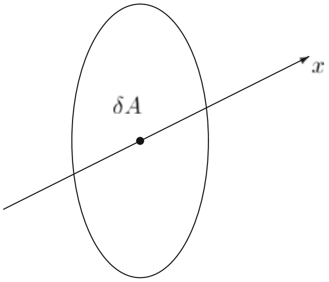


**Fig. 2.1** Diffusion across a small area

diffusion volume has an awkward geometry. In practice, however, most geometries encountered will be—or can be simplified to—one of but a few standard forms such as rectangular, cylindrical or spherical—for which the full diffusion equation has been established (see, e.g., Crank [2]). In Cartesian coordinates this equation, Fick's second diffusion equation, in its most general form, is

$$\frac{\partial c}{\partial t} = D_x \frac{\partial^2 c}{\partial x^2} + D_y \frac{\partial^2 c}{\partial y^2} + D_z \frac{\partial^2 c}{\partial z^2} \, . \tag{2.3}$$

This expresses the rate of change of concentration with time at given coordinates $(t, x, y, z)$ in terms of second space derivatives and three different diffusion coefficients. It is possible for $D$ to be direction-dependent (in anisotropic media) and it can also be space-dependent but for a solute in solution, it is normally equal in all directions and usually the same everywhere, so (2.3) simplifies to

$$\frac{\partial c}{\partial t} = D \left( \frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} + \frac{\partial^2 c}{\partial z^2} \right) , \tag{2.4}$$

that is, the usual three-dimensional form. Even this is rather rarely applied—we always try to reduce the number of dimensions, preferably to one, giving

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} \tag{2.5}$$

(but see Sect. 12.4 for 2D systems and an example of a 3D problem).

If the geometry of the system is cylindrical, it is convenient to switch to cylindrical coordinates: $z$ along the cylinder, $r$ the radial distance from the axis and $\theta$ the angle. In most cases, concentration is independent of the angle and the diffusion equation is then (see Appendix B)

$$\frac{\partial c}{\partial t} = D \left( \frac{\partial^2 c}{\partial z^2} + \frac{\partial^2 c}{\partial r^2} + \frac{1}{r} \frac{\partial c}{\partial r} \right) \, . \tag{2.6}$$

There is no gradient along $z$ (the axis) for a (long) cylinder, so only $r$ remains

$$\frac{\partial c}{\partial t} = D \left( \frac{\partial^2 c}{\partial r^2} + \frac{1}{r} \frac{\partial c}{\partial r} \right) \, . \tag{2.7}$$

For a spherical system, assuming no concentration gradients other than away from the centre (radially), the equation becomes

$$\frac{\partial c}{\partial t} = D \left( \frac{\partial^2 c}{\partial r^2} + \frac{2}{r} \frac{\partial c}{\partial r} \right) \, . \tag{2.8}$$

### 2.2.2 Diffusion Current

Equation (2.2) gives the flux in mol s$^{-1}$ of material as the result of a concentration gradient. If there is such a gradient normal to an electrode/electrolyte interface, then there is a flux of material at the electrode and this takes place via the electron transfer. An electroactive species diffuses to the electrode, takes part in the electron transfer and becomes a new species. The electrical current $i$ flowing is then equal to the molar flux multiplied by the number $n$ of electrons transferred for each molecule or ion (2.2), and the Faraday constant

$$i = nFAD \left( \frac{\partial c}{\partial x} \right)_{x=0} \tag{2.9}$$

for a reduction current. The flux and the current are thus, in a sense, synonymous and will, in fact, profitably be expressed simply in terms of the concentration gradient itself or its dimensionless equivalent, to be discussed later (Sect. 2.3).

### 2.2.3 Convection

If we cannot arrange for our solution to be (practically) stagnant during our experiment, then we must include convective terms in the equations. Figure 2.2 shows a plot of concentration against the $x$-coordinate at a given instant. Let $x_1$ be a fixed point along $x$, with concentration $c_1$ at some time $t$, and let the solution be moving forward along $x$ with velocity $v_x$, so that after a small time interval $\delta t$, concentration $c_2$ (previously at $x_2$) has moved to $x_1$ by the distance $\delta x$. If $\delta t$ and $\delta x$

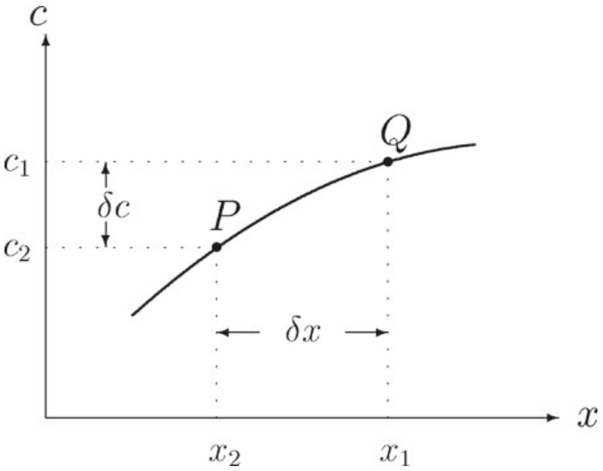


**Fig. 2.2** Convection

are chosen sufficiently small, we may consider the line PQ as straight and we have, for the change $\delta c$ at $x_1$

$$\delta c = -\delta x \frac{\mathrm{d}c}{\mathrm{d}x} \,. \tag{2.10}$$

Dividing by $\delta t$, taking $v_x = \delta x / \delta t$ and going to the infinitesimal limit, we get for the $x$-term

$$\frac{\partial c}{\partial t} = -v_x \frac{\partial c}{\partial x} \,. \tag{2.11}$$

If there is convection in all three directions, this expands to

$$\frac{\partial c}{\partial t} = -v_x \frac{\partial c}{\partial x} - v_y \frac{\partial c}{\partial y} - v_z \frac{\partial c}{\partial z} \,. \tag{2.12}$$

This treatment ignores the diffusional processes taking place simultaneously; the two transport terms are additive in the limit.

Convection terms commonly crop up with the dropping mercury electrode, rotating disk electrodes and in what has become known as hydrodynamic voltammetry, where the electrolyte is made to flow past an electrode in some reproducible way (e.g. the impinging jet, channel and tubular flows, vibrating electrodes, etc.). This is discussed in Chap. 14.

### 2.2.4 Migration

Simulation of migration is treated in Chap. 13. Often, the electrochemist is able to eliminate this transport term (and will do so for practical reasons as well). If our species is charged, that is, it is an ion, then it may experience electrical forces due to potential fields. This will be significant in solutions of ionic electroactive species, not containing a sufficiently large excess of inert electrolyte.

In general (see Vetter [6]), for an electroactive cation with charge $+z_A$ and anion with charge $-z_B$, an inert electrolyte with the same two charges on its ions, and with $r$ the concentration ratio electrolyte/electroactive ion, we have the rather awkward equation

$$\frac{i}{i_0} = \left(1 + \left|\frac{z_A}{z_B}\right|\right)(1 + r)\left(1 - \left(\frac{r}{1 + r}\right)^p\right) \tag{2.13}$$

where

$$p = \left(1 + \left|\frac{z_A}{z_B}\right|\right)^{-1} \tag{2.14}$$

and $i_0$ is the pure diffusion current, without migration effects. To illustrate, let us take $|z_A| = |z_B| = 1$. Then $i/i_0 = 2$ for $r = 0$ (no inert electrolyte), 1.17 for $r = 1$, 1.02 for $r = 10$ and 1.002 for $r = 100$. For very accurate studies, then, inert electrolyte should be in excess by a factor of 100 or more. Cases where this is not so are dealt with in the migration Chap. 13.

There is one situation in which migration can have an appreciable effect, even in the presence of excess inert electrolyte. For the measurement of very fast reactions, one must resort to techniques involving very small diffusion layers (see Sect. 2.4.1.1 for the definition)—either by taking measurements at very short times or forcing the layer thickness down by some means. If that thickness becomes comparable in magnitude with that of the diffuse double layer, and the electroactive species is charged, then migration will play a part in the transport to and from the electrode. The effect has been clearly explained elsewhere [7]. A rough calculation for a planar electrode in a stagnant solution, assuming the thickness of the diffuse double layer to be of the order of $10^{-9}$m and the diffusion coefficient of the electroactive species to be $10^{-12}\ \text{m}^2\ \text{s}^{-1}$ (which is rather slow) shows that migration effects are expected during the first few microseconds or so. This is also gone into in Chap. 13.

## 2.2.5 Total Transport Equation

This section serves merely to emphasise that for a given cell system, the full transport equation is the sum of those for diffusion, convection and migration. We might write, quite generally,

$$\frac{\partial c}{\partial t} = \left(\frac{\partial c}{\partial t}\right)_{\text{diff}} + \left(\frac{\partial c}{\partial t}\right)_{\text{conv}} + \left(\frac{\partial c}{\partial t}\right)_{\text{migr}} \tag{2.15}$$

with the "diff" term as defined by one of the equations (2.3)–(2.8), the "conv" term by (2.11) and "migr" related to (2.13). At any one instant, these terms are simply additive. Digitally, we can "freeze" the instant and evaluate the sum of the separate terms. There may be non-transport terms to add as well, such as kinetic terms, to be discussed next.

## 2.2.6 Homogeneous Kinetics

Homogeneous reactions are chemical reactions not directly dependent upon the electrode/electrolyte interface, taking place somewhere within the electrolyte (or, in principle, the metal) phase. These lead to changes in concentration of reactants and/or products and can have marked effects on the dynamics of electrochemical processes. They also render the dynamic equations much more difficult to solve and it is here that digital simulation sees much of its use. Whereas analytical solutions for kinetic complications are difficult to obtain, the corresponding discrete expressions are obtained simply by extending the diffusion equation by an extra, kinetic, term (although practical problems arise, see Chaps. 5 and 9). The actual form of this depends upon the sort of chemistry taking place. In the simplest case, met with in flash photolysis, we have a single substance generated by the flash, then decaying in solution by a first- or second-order reaction; this is represented by equations of the form

$$\frac{\partial c}{\partial t} = -k_1 c \tag{2.16}$$

or

$$\frac{\partial c}{\partial t} = -2k_2 c^2 \tag{2.17}$$

and these can be added to the transport terms. Very often, we have several substances interacting chemically, as in the example of the simple electrochemical reaction

$$\mathrm{A} + ne^- \rightleftharpoons \mathrm{B} \tag{2.18}$$

followed by chemical decay of the product B. If this is first-order and we have a simple one-dimensional diffusion system, we then have the two equations ($c_A$ and $c_B$ denoting concentrations of, substances A and B, respectively; $D_A$ and $D_B$ the two respective diffusion coefficients)

$$\begin{aligned} \frac{\partial c_A}{\partial t} &= D_A \frac{\partial^2 c_A}{\partial x^2} \\ \frac{\partial c_B}{\partial t} &= D_B \frac{\partial^2 c_B}{\partial x^2} - k_1 c_B \ . \end{aligned} \tag{2.19}$$

There is a great variety of such reactions including dimerisation, disproportionation and catalytic reactions, both preceding and following the electrochemical step(s) and it is not useful to attempt to list them all here but see Bard and Faulkner [1] for a full discussion. The point is merely to stress that they are (with greater or lesser difficulty) digitally tractable, as will be shown in Chaps. 5 and 9.

There is one problem that makes homogeneous chemical reactions especially troublesome. Most often, a mechanism to be simulated involves species generated at the interface, that then undergo chemical reaction in the solution. This leads to concentration profiles for these species that are confined to a thin layer near the interface—thin, that is, compared with the diffusion layer (see Sect. 2.4.1.1, the Nernst diffusion layer). This is called the reaction layer (see [1, 6, 8]). Simulation parameters are usually chosen so as to resolve the space within the diffusion layer and, if a given profile is much thinner than that, the resolution of the sample point spacing might not be sufficient. The thickness of the reaction layer depends on the nature of the homogeneous chemical reaction. In any case, any number given for such a thickness—as with the diffusion layer thickness—depends on how the thickness is defined. Wiesner [9] first derived an expression for the reaction layer thickness $\mu$,

$$\mu = \sqrt{\frac{D}{k}}\,. \tag{2.20}$$

(Wiesner's expression used different symbols, but this is not important.) This expression strictly holds only for a first-order reaction and Vetter [6] provides a more general expression. However, the above expression is sufficient for most simulation purposes. The problem exists only for rather large values of the rate constant $k$; for small values, $\mu$ becomes greater than the diffusion layer thickness, which will then dominate the concentration profile. At the other end of the scale of rate constants, for very fast reactions, $\mu$ can become very small. The largest rate constant possible is about $10^{10}\,\mathrm{s}^{-1}$ (the diffusion limit) and this leads to a $\mu$ value only about $10^{-5}$ the thickness of the diffusion layer, so there must be some sample points very close to the electrode. This problem has been overcome, first by using unequal intervals, then by the use of dynamic grids, both of which are discussed in Chap. 7.

### 2.2.7 Heterogeneous Kinetics

In real (as opposed to model) electrochemical cells, the net current flowing will often be partly determined by the kinetics of electron transfer between electrode and the electroactive species in solution. This is called heterogeneous kinetics, as it refers to the interface instead of the bulk solution. The current in such cases is obtained from the Butler–Volmer expressions relating current to electrode potential [1, 6–8]. We have at an electrode the process (2.18), with concentrations at the electrode/electrolyte interface $c_{A,0}$ and $c_{B,0}$, respectively. We take as positive current that going into the electrode, i.e., electrons leaving it, which corresponds to the reaction (2.18) going from left to right, or a reduction. Positive or forward (reduction) current $i_f$ is then related to the potential $E$ by

$$i_f = nFAc_{A,0}k^0 \exp\left(\frac{-\alpha n\mathcal{F}}{\mathcal{R}\mathcal{T}}(E - E^0)\right) \tag{2.21}$$

with $A$ the electrode area, $k^0$ a standard heterogeneous rate constant, $\alpha$ the so-called transfer coefficient which lies between 0 and 1 and $E^0$ the system's standard potential. For the reverse (oxidation) current $i_b$,

$$i_b = -nFAc_{B,0}k^0 \exp\left((1-\alpha)\frac{n\mathcal{F}}{\mathcal{R}\mathcal{T}}(E - E^0)\right). \tag{2.22}$$

Both processes may be running simultaneously. The net current is then the sum $(i_f + i_b)$ and this will, through (2.9), fix the concentration gradients at the electrode in these cases.

The Butler–Volmer relation which, as Inzelt points out in a thorough review of the history of the formulation of electrode kinetics [10], should really be called the Butler–Volmer–Erdey-Gruz theory. There has been much discussion comparing the theory with the Marcus–Hush theory. This has the unfortunate property that to use it, one must compute integrals, although Zeng et al. [11] have presented some simpler approximations, and Bieniasz [12] showed some ways to get accurate rate constants using the theory. Feldberg [13] advises caution when applying Butler–Volmer to steady state currents at an ultramicrodisk electrode, where the currents can be lower than predicted by Butler–Volmer, compared with the more correct Marcus–Hush theory, whereas Henstridge et al. [14] find that such conditions are unlikely to be met in practice. The Inzelt and Feldberg papers contain numerous references to this complex field, but in this book only Butler–Volmer theory will be dealt with.

If a reaction is very fast, it may be simpler to make the assumption of complete reversibility or electrochemical equilibrium at the electrode, at a given potential $E$. The Nernst equation then applies:

$$E = E^0 - \frac{\mathcal{RT}}{n\mathcal{F}} \ln\left(\frac{c_{B,0}}{c_{A,0}}\right) \tag{2.23}$$

or, for the purpose of computation,

$$\frac{c_{A,0}}{c_{B,0}} = \exp\left(\frac{n\mathcal{F}}{\mathcal{RT}}(E - E^0)\right) . \tag{2.24}$$

Just how this is applied in simulation will be seen in later chapters.

The foregoing ignores activity coefficients. If these are known, they can be inserted. Most often they are taken as unity.

## 2.3 Normalisation: Making the Variables Dimensionless

In most simulations, it will be advantageous to transform the given equation variables into dimensionless ones. This is done by expressing them each as a multiple of a chosen reference value, so that they no longer have dimensions. The time variable $t$, for example, is expressed as a multiple of some characteristic time $\tau$, which may be different things depending upon the experiment to be simulated. Sometimes it might be the total duration of an experiment (the observation time) or, in the case of a linear sweep experiment, the length of time it takes for the voltage to change by some specified amount. The distance from an electrode $x$ can be conveniently expressed as a multiple of some characteristic distance $\delta$, which will be defined below. Concentrations are normally expressed as multiples of some reference concentration, usually the initial bulk concentration of a certain species involved in the reaction, say $c^*$. The convention adopted in the rest of the book is, then, that the new dimensionless variables, written in capitals, are

$$\begin{aligned} C &= c/c^* \\ X &= x/\delta \\ T &= t/\tau \ . \end{aligned} \tag{2.25}$$

The reference time scale $\tau$ depends on the system to be simulated, as will be seen in the next section, where some model systems are described. There, the characteristic distance $\delta$ will also be defined as used in this book (Sect. 2.4.1.1). Other variables that are normalised are the current and electrode potential. Current $i$ is proportional to the concentration gradient, by Fick's first equation (2.2), as expressed in (2.9). We introduce the dimensionless gradient or flux, defined as

$$G = \left.\frac{\partial C}{\partial X}\right|_{X=0} \ . \tag{2.26}$$

This will now represent the current in dimensionless form. The actual current can be calculated by denormalisation, that is,

$$i = nFADG\frac{c^*}{\delta} \ . \tag{2.27}$$

The standard heterogeneous rate constant, seen in Eqs. (2.21) and (2.22), with dimensions m s$^{-1}$, is normalised by

$$K^0 = k^0\sqrt{\tau/D} \ . \tag{2.28}$$

Potential values (in V) are normalised by the $\mathcal{RT}/n\mathcal{F}$ unit and usually by referring to some reference value $E^0$, using (in this book) the symbol $p$:

$$p = \frac{n\mathcal{F}}{\mathcal{RT}}\left(E - E^0\right) \tag{2.29}$$

so that one $p$-unit corresponds to $25.69/n$ mV. Thus, the two Butler–Volmer components in Eqs. (2.21) and (2.22) can be expressed in terms of dimensionless current $G$ as

$$G = K_f C_{A,0} - K_b C_{B,0} \tag{2.30}$$

with

$$\begin{aligned} K_f &= k^0 \exp\{-\alpha p\} \\ K_b &= k^0 \exp\{(1-\alpha)p\} \end{aligned} \tag{2.31}$$

and the Nernst equation very simply as

$$\frac{C_{A,0}}{C_{B,0}} = e^p \, . \tag{2.32}$$

With certain rules and tricks, as will be shown, this will lead to equations whose solutions are much more general and useful than if we solve the dimensioned equation for our particular parameter set of values.

## 2.4 Some Model Systems and Their Normalisations

When developing a new simulation method, it is good to have a number of model systems at hand, for which there are known results, whether these be in the form of analytical solutions (concentration profiles, current) or well-established series solutions (as in the case of linear sweep voltammetry (LSV), where some parameters have been calculated to quite high precision). The test models should be chosen, as far as possible, to challenge the method. If the new method's primary purpose, for example, is simply greater efficiency, then a simple model like the Cottrell system and chronopotentiometry may be enough to demonstrate that; these two differ fundamentally in their boundary conditions, the Cottrell system having a so-called Dirichlet boundary condition (given concentrations at the boundary), while chronopotentiometry has a derivative or Neumann condition, where gradients are specified at the boundary. If a method under development is expected to give high resolution (small intervals) along $x$—usually at the boundary—a model that provides marked concentration changes very close to the boundary is the best for testing that.

Along with a group of models that have shown themselves useful, their particular normalisations will be presented. The first model, the Cottrell system, will also serve to introduce the concept of the Nernst diffusion layer.

### *2.4.1 Potential Steps*

Potential step experiments are a popular way to look at electrochemical kinetics. The oldest known is the Cottrell system, where the potential stepped to is so far negative that the resulting current is limited by the transport of the active substance. If the step is not so far negative, one then has either Nernstian boundary conditions, or those for quasireversible or irreversible systems. All of these cases have been analytically solved. As well, there are two systems involving homogeneous chemical reactions, from flash photolysis experiments, for which there exist solutions to the potential step experiment, and these are also given; they are valuable tests of any simulation method, especially the second-order kinetics case.

**Fig. 2.3** A semi-infinite one-dimensional cell

#### 2.4.1.1 Cottrell System

We introduce here the diffusion-controlled potential-step experiment, hereafter called the Cottrell experiment [15]. Consider Fig. 2.3, showing a long thin tube representing an electrochemical cell, bounded at one end by an electrode and filled with electrolyte and an electroactive substance initially at concentration $c^*$ (the bulk concentration). We place the electrode at $x = 0$ and the other, counter-electrode (not shown), at a large distance so that what happens there is of no consequence to us. We apply, at $t = 0$, a potential such that our electroactive substance reacts at the electrode infinitely fast—that is, its concentration $c_0$ at the electrode ($x = 0$) is forced to zero and kept there. Clearly, there will be flow of substance towards the electrode by diffusion (we assume no convection here) and we will gradually cause some depletion of material in the solution near $x = 0$; this depletion region will grow out from the electrode with time. Mathematically, this is described by the diffusion equation

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} \tag{2.33}$$

with the boundary conditions

$$\begin{aligned} t = 0, \text{all } x: &\quad c = c^* \\ t > 0, x = 0: &\quad c = 0 \\ \text{all } t, x \to \infty: &\quad c = c^* \;. \end{aligned} \tag{2.34}$$

This classical equation with the boundary conditions as shown has an analytical solution (Cottrell [15], see also standard texts such as Bard and Faulkner [1] or Galus [8]):

$$c(x,t) = c^* \operatorname{erf}\left(\frac{x}{2\sqrt{Dt}}\right) \;. \tag{2.35}$$

In electrochemical experiments, we usually want the current or, since it is related simply by (2.9) to $\partial c/\partial x$ at $x = 0$, we want $(\partial c/\partial x)_0$. This is obtained by differentiating (2.35) and setting $x = 0$, resulting in

$$\left(\frac{\partial c}{\partial x}\right)_0 = \frac{c^*}{\sqrt{\pi D t}} \tag{2.36}$$

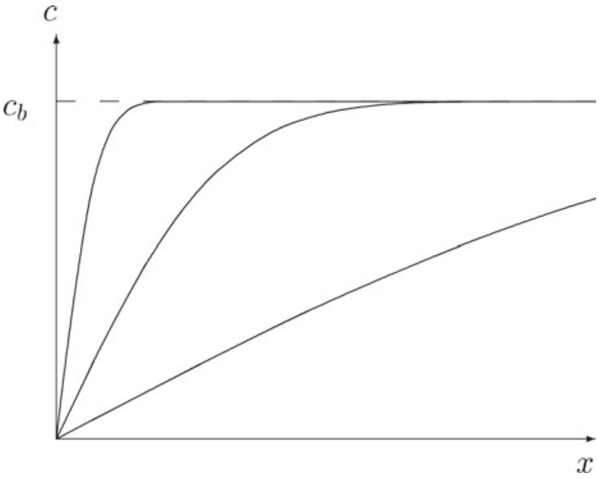


**Fig. 2.4** Concentration profile changing with time for the Cottrell experiment

and the current itself is given by

$$i = \frac{nFA\sqrt{D}c^*}{\sqrt{\pi t}}\ , \tag{2.37}$$

the **Cottrell equation**.

The function erf is the error function, for which tables exist [16], and which can be numerically computed (see the function ERF discussed in Appendix E). The solution, (2.35), is shown in Fig. 2.4 for three values of $t$, increasing as the curves go to the right.

These so-called concentration profiles agree with our intuitive picture of what should happen. Note that the concentration gradient at $x = 0$ decreases with time. The current function declines with the inverse square root of time (2.36). If, for a particular $t$ value, we wish to know the current, we can insert $c^*$, $D$ and $t$ into this equation and use (2.9) to get it.

It is clear from Fig. 2.4 that we should be able to define a distance that roughly corresponds, at a given time, to the distance over which much of the concentration change has taken place. One possible choice for this is the distance $\delta$ as shown in Fig. 2.5, obtained by continuing the concentration gradient at $x = 0$ straight up to $c^*$. Since this tangent line has the equation

$$c = \left(\frac{\partial c}{\partial x}\right)_0 x = \frac{c^* x}{\sqrt{\pi D t}}\ , \tag{2.38}$$

$\delta$ will be obtained by substituting $c = c^*$ and $x = \delta$; this leads to

$$\delta = \sqrt{\pi D \tau} \tag{2.39}$$

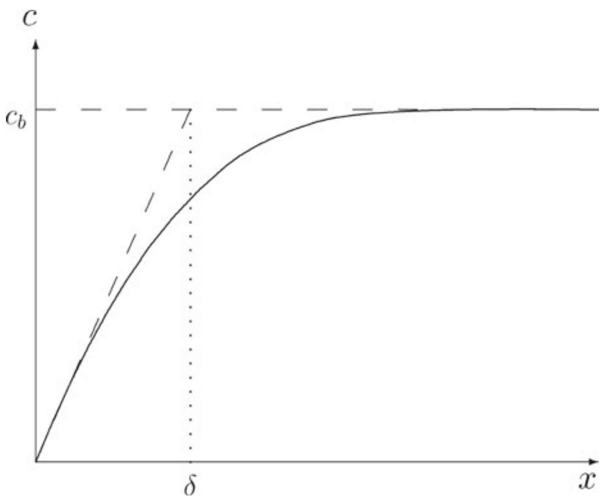


**Fig. 2.5** The diffusion layer thickness $\delta$

now expressed for the particular observation time $\tau$. This quantity—a length scale—was defined by Nernst (and Brunner) in 1904 [17, 18], and is named after the former. We find that, at any given time, there will be noticeable concentration changes in the solution within a space extending only a few multiples of $\delta$.

This definition of $\delta$ is one of several possible. The way it is defined above yields that distance for which the concentration has moved from zero to $c^*$ by a fraction $\mathrm{erf}(\frac{1}{2}\sqrt{\pi}) \approx 0.8$ or in other words, about 80 % of the change has happened at that point. Although this might be the most rational definition, others can be agreed upon. In the present context, it turns out that a smaller distance is the most convenient:

$$\delta = \sqrt{D\tau}\ . \tag{2.40}$$

At this distance, about 52 % of the total change has happened. This definition of $\delta$ will be used in the remainder of the book. It is customary, since Feldberg's seminal chapter [19], to use a multiple of $6\sqrt{T_m}$ as the outer limit, with $T_m$ being the maximum dimensionless time in a given simulation. The number 6 is normally used without comment. Looking at the solution of the potential jump experiment [see below, (2.43)], at this multiple of $\delta$, the concentration changes are less than $10^{-4}$ the bulk value and for simulations aiming at an accuracy of no better than about 0.01 %, this is a sufficient distance. On occasion it is however better to increase the distance somewhat, as has been the case with work by the present authors [20–22], where multiples of 8 or even 10 have been used to ensure accuracy.

This scale is now used. The three variables $c$, $x$ and $t$ are rendered dimensionless by the normalisations in (2.25) and applying these to (2.33) results in the new dimensionless diffusion equation

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} \tag{2.41}$$

and for the Cottrell system in these terms, the dimensionless boundary conditions,

$$\begin{aligned} T = 0, \text{all } X: &\quad C = 1, \\ T > 0, X = 0: &\quad C = 0, \\ \text{all } T, X \to \infty: &\quad C = 1\ . \end{aligned} \tag{2.42}$$

From both the diffusion equation and the boundary conditions, such parameters as $D$ and $c^*$ have now been eliminated. The solution is then

$$C(X,T) = \operatorname{erf}\left(\frac{X}{2\sqrt{T}}\right) \tag{2.43}$$

for the concentrations and

$$\left(\frac{\partial C}{\partial X}\right)_0 = G = \frac{1}{\sqrt{\pi T}}\ . \tag{2.44}$$

This might be called the dimensionless Cottrell equation, for "current" $G$, which in fact is the dimensionless concentration gradient at $X = 0$.

#### 2.4.1.2 Potential Step, Reversible System

In the Cottrell experiment, as described in the last section, we have a step to a very negative potential, so that the concentration at the electrode is kept at zero throughout. It is possible also to step to a less extreme potential. If the system is reversible, and we consider the two species A and B, reacting as in (2.18), then we have the Nernstian boundary condition as in (2.24). Using (2.29) and assigning the symbols $C_A$ and $C_B$, respectively, to the dimensionless concentrations of species A and B, we now have the new boundary conditions for the potential step,

$$\begin{aligned} T = 0, \text{all } X: &\quad C_A = 1, C_B = 0, \\ T > 0, X = 0: &\quad C_A/C_B = e^p, \\ \text{all } T, X \to \infty: &\quad C_A = 1, C_B = 0\ , \end{aligned} \tag{2.45}$$

in which species B is not initially present. Note that substance A is now the reference species and the values of its diffusion coefficient $D_A$ and its initial bulk concentration

$c_A^*$ are the ones used in the normalisations (2.25) and (2.40). Similarly, if the diffusion coefficients are different for the two species, we also define the ratio

$$d = D_B/D_A \; . \tag{2.46}$$

There is the additional boundary condition (flux condition),

$$f_A + f_B = 0 \tag{2.47}$$

or, in terms of concentration gradients at the electrode,

$$D_A \left. \frac{\partial c_A}{\partial x} \right|_{x=0} + D_B \left. \frac{\partial c_B}{\partial x} \right|_{x=0} = 0 \tag{2.48}$$

which, in its dimensionless form and using (2.46), becomes

$$\left. \frac{\partial C_A}{\partial X} \right|_{X=0} + d \left. \frac{\partial C_B}{\partial X} \right|_{X=0} = 0 \; . \tag{2.49}$$

The solution to all this is, as given in Galus [8], is

$$C_A(X,T) = \frac{d^{-1}e^p + \operatorname{erf}\left(\frac{X}{2\sqrt{T}}\right)}{1 + d^{-1}e^p} \tag{2.50}$$

and for $C_B$

$$C_B(X,T) = \frac{d \operatorname{erfc}\left(\frac{X}{2\sqrt{dT}}\right)}{1 + d^{-1}e^p} \tag{2.51}$$

and the current (expressed as the dimensionless gradient for A) is

$$G = G_{Cott}/(1 + d^{-1}e^p) \tag{2.52}$$

where $G_{Cott}$ is the $G$-value for the simple Cottrell case as in (2.44). This equation is also seen in Bard and Faulkner [1, p. 178].

If the two species' diffusion coefficients are assumed equal ($d = 1$), the above equations simplify in an obvious way. In fact, then the problem is mathematically equivalent to the simple Cottrell case. Cottrell pointed out [15] that then, initially the concentrations at the electrode of the two species will instantly change to their Nernstian values and remain there after that.

A final point concerns the fact that, if indeed $d = 1$, then at any point $X$,

$$C_A(X,T) + C_B(X,T) = C_A(X,0) + C_B(X,0) \; . \tag{2.53}$$

This equation could be used to simplify the simulation, reducing it to only a single species to be simulated. Agreeing with Feldberg however (private communication), this is not a good idea. Rather, the above equation should be used as a check on a given simulation, to make sure that all is well.

#### 2.4.1.3 Potential Step, Quasi- and Irreversible System

For the **quasireversible** case, two species A and B must again be considered and the two boundary conditions are the flux condition (2.49) and the dimensionless form of the Butler–Volmer equation. The forward and backward heterogeneous rate constants $k_f$ and $k_b$ are normalised:

$$K_f = k_f \sqrt{\frac{\tau}{D_A}} \tag{2.54}$$

$$K_b = k_b \sqrt{\frac{\tau}{D_A}} \tag{2.55}$$

and the dimensionless current $G$ is as given in (2.30). With suitable discretisation, $G$ becomes one of the two boundary conditions, the other one being the usual flux expression (2.47).

This case was studied and published in 1952–1953 by several groups independently, some giving the solution for the case of both $K_f$ and $K_b$ being nonzero and some treating the totally irreversible case, $K_b = 0$. See [8] for the references. Texts tend to give only the solution for the current, but by continuing the treatments in [8, p. 235] or [1] and using form 170 in the tables in Doetsch [23] the solution, in dimensionless form, is [1, p. 192]

$$C_A(X,T) = \\ 1 - \frac{K_f}{K_f + K_b}\left\{ - \exp(HX + H^2T)\,\mathrm{erfc}\left(H\sqrt{T} + \frac{X}{2\sqrt{T}}\right) + \mathrm{erfc}\left(\frac{X}{2\sqrt{T}}\right)\right\} \tag{2.56}$$

for the concentration profile of species A and

$$C_B(X,T) = \\ \frac{K_f}{K_f + K_b}\left\{ - \exp(HX + H^2T)\,\mathrm{erfc}\left(H\sqrt{T} + \frac{X}{2\sqrt{dT}}\right) + \mathrm{erfc}\left(\frac{X}{2\sqrt{dT}}\right)\right\} \tag{2.57}$$

for species B, where the dimensionless variable $H$ is defined as

$$H = k_f \sqrt{\frac{\tau}{D_A}} + k_b \sqrt{\frac{\tau}{D_B}} \tag{2.58}$$

(deviating slightly from the notation in [24]). The dimensionless current then is

$$G = K_f \exp(H^2 T)\,\mathrm{erfc}(H\sqrt{T})\ . \tag{2.59}$$

Modification to the totally irreversible case ($k_b = 0$) is trivial, as is the simplification to equal diffusion coefficients ($d = 1$).

#### 2.4.1.4 Potential Step, Homogeneous Chemical Reactions

Three examples are popular here. The first two start with flash photolysis, where an intense flash irradiates the whole cell at $t = 0$, instantly producing an electrochemically active species that decays chemically in time, either by a first-order reaction, or a second-order reaction. The labile substance is assumed to be formed uniformly in the cell space with a bulk concentration of $c^*$. These are cases where the concentration at the outer boundary is not constant, falling with time. The third case, the catalytic or **EC**$'$ system (see [1]), is of special interest because of the reaction layer it gives rise to.

The **Reinert–Berg** system is the one in which the reactions are

$$\begin{aligned} \mathrm{A} + e^- &\rightarrow \mathrm{B} \\ \mathrm{A} &\rightarrow prod\ , \end{aligned} \tag{2.60}$$

substance A having been generated by an intense flash. The two reactions take place simultaneously. We need only consider the single species A. This system poses no special problems. Reinert and Berg solved it [25] for a potential step to very negative potentials, that is, doing a Cottrell experiment on this system. The diffusion equation becomes

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - kc \tag{2.61}$$

where $k$ is the rate constant of the homogeneous chemical reaction and boundary conditions are

$$\begin{aligned} t = 0, \text{all } x: &\quad c = c^* \\ t > 0, x = 0: &\quad c = 0 \\ \text{all } t, x \rightarrow \infty: &\quad c = c^* e^{-kt}\ . \end{aligned} \tag{2.62}$$

Note the difference from (2.34). Normalising as usual, with the additional normalisation of rate constant $k$ to $K$,

$$K = k\tau \tag{2.63}$$

so that these equations become

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} - KC \tag{2.64}$$

and

$$\begin{aligned} T = 0, \text{all } X : &\quad C = 1 \\ T > 0, X = 0 : &\quad C = 0 \\ \text{all } T, X \to \infty : &\quad C = e^{-KT} \; . \end{aligned} \tag{2.65}$$

The solution of this is the one for the simple Cottrell system, multiplied by the decay factor

$$C(X,T) = \exp(-KT)\,\text{erf}\left(\frac{X}{2\sqrt{T}}\right) \tag{2.66}$$

and [25]

$$G(T) = \exp(-KT)\frac{1}{\sqrt{\pi T}} \tag{2.67}$$

(the solution of Reinert and Berg [25], is for a dropping mercury electrode but can be transferred to the planar Cottrell case). Obviously, one must choose the characteristic (observation) time $\tau$ reasonably—several multiples of the half-life—so that even out in the bulk, there is still some substance left at that time, or else the calculation will be operating on values very close to zero. This will depend on the value of $K$. When simulating the plain Cottrell experiment, it is customary to simulate to $T = 1$, but here, one might only go to $T = n/K$, with $n$ some smallish number, so that $\exp(-KT)$ does not become too small.

In the Reinert–Berg system, the homogeneous chemical reaction involves a bulk species, and there is no reaction layer (Sect. 2.2.6).

The **Birk–Perone** system, a flash photolysis experiment with subsequent second-order decay, is a little more interesting because it can, with an unsuitable simulation method, lead to negative concentration values. The simultaneous reactions are

$$\begin{aligned} \text{A} + e^- &\to \text{B} \\ 2\text{A} &\to \textit{prod} \end{aligned} \tag{2.68}$$

and this has the governing equation

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - 2kc^2 \; , \tag{2.69}$$

where $k$ is the rate constant of the homogeneous chemical reaction and boundary conditions are

$$\begin{aligned} t = 0, \text{all } x: &\quad c = c^* \\ t > 0, x = 0: &\quad c = 0 \\ \text{all } t, x \to \infty: &\quad c = c^*/(1 + 2ktc^*)\ . \end{aligned} \tag{2.70}$$

The boundary condition at $X \to \infty$ is the solution of the simple homogeneous reaction taking place there. Normalising all variables, and $k$ normalised using

$$K = 2kc^*\tau\ , \tag{2.71}$$

these become

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} - KC^2\ , \tag{2.72}$$

with

$$\begin{aligned} T = 0, \text{all } X: &\quad C = 1 \\ T > 0, X = 0: &\quad C = 0 \\ \text{all } T, X \to \infty: &\quad C = (1 + KT)^{-1}\ . \end{aligned} \tag{2.73}$$

A solution for this system was first attempted by Birk and Perone [26], who however oversimplified their assumptions. This was pointed out later [27] and the more rigorous solution (current only) was found to be

$$\frac{i_k}{i_{k=0}} = \frac{1}{1+\theta} \left\{ 1 + \sum_{n=1}^{\infty} a_n \left( \frac{\theta}{1+\theta} \right)^n \right\}\ , \tag{2.74}$$

in which $i_{k=0}$ is the plain Cottrell solution, $\theta$ is defined as $2kc^*t = KT$ and the first 10 coefficients $a_n$ are [27]

$$\begin{aligned} a_1 &= 4/\pi - 1 = 0.27324 \\ a_2 &= 0.08327 \\ a_3 &= 0.02893 \\ a_4 &= 0.01162 \\ a_5 &= 0.00540 \\ a_6 &= 0.00286 \\ a_7 &= 0.00169 \\ a_8 &= 0.00108 \\ a_9 &= 0.00074 \\ a_{10} &= 0.00053\ . \end{aligned}$$

Another system of interest in connection with potential steps (and, see below, LSV) is the **catalytic** or **EC**$'$ system (classified as $\mathrm{E}_r\mathrm{C}_i'$ in [1]), described in simplified form by

$$\begin{aligned} \mathrm{A} + e^- &\rightleftharpoons \mathrm{B} \\ \mathrm{B} &\rightarrow \mathrm{A}\,, \end{aligned} \tag{2.75}$$

where the product B reverts, with pseudo-first-order rate constant $k$, to the original A. The first reaction is conveniently taken to be diffusion limited (that is, the potential is very negative as in the Cottrell experiment). Normalising as usual [rate constant $k$ as above, (2.63)] and assuming equal diffusion coefficients ($d = 1$), the boundary conditions are

$$\begin{aligned} T = 0, \text{all } X: &\quad C_A = 1, C_B = 0 \\ T > 0, X = 0: &\quad C_A = 0 \\ \text{all } T, X \to \infty: &\quad C_A = 1, C_B = 0\,. \end{aligned} \tag{2.76}$$

The solution, derived by Delahay and Stiehl [28] and, independently in the same year by Miller [29], in dimensionless form, is

$$\begin{aligned} C_A(X,T) &= 1 - \tfrac{1}{2}\exp(X\sqrt{K})\,\mathrm{erfc}\left(\frac{X}{2\sqrt{T}} + \sqrt{KT}\right) \\ &\qquad -\tfrac{1}{2}\exp(-X\sqrt{K})\,\mathrm{erfc}\left(\frac{X}{2\sqrt{T}} - \sqrt{KT}\right) \\ C_B(X,T) &= \ \tfrac{1}{2}\exp(X\sqrt{K})\,\mathrm{erfc}\left(\frac{X}{2\sqrt{T}} + \sqrt{KT}\right) \\ &\qquad +\tfrac{1}{2}\exp(-X\sqrt{K})\,\mathrm{erfc}\left(\frac{X}{2\sqrt{T}} - \sqrt{KT}\right) \end{aligned} \tag{2.77}$$

and the current

$$G = \sqrt{K}\,\mathrm{erf}\,\sqrt{KT} + \frac{\exp(-KT)}{\sqrt{\pi T}}\,. \tag{2.78}$$

This system will be discussed again in a later chapter, because it is of special interest, both species A and B forming a reaction layer. The thickness $\mu$ of this layer was given in Sect. 2.2.6 and this can now be normalised, for a first-order homogeneous reaction, for which we have the dimensionless rate constant as in (2.63), giving the dimensionless reaction layer thickness

$$\mu^* = \frac{1}{\sqrt{K}}\,. \tag{2.79}$$

Note that for large $K$ and $T$, the current $G$ approaches the constant value $\sqrt{K}$ [24, p. 504] citing [30], converting to our dimensionless units.

### 2.4.2 Constant Current

While the Cottrell system might be regarded as the simplest possible model with a Dirichlet boundary condition (that is, in which boundary concentrations are specified), the constant current case is the simplest possible for the Neumann boundary condition, in which a concentration gradient is specified at the boundary. This model can also be called the chronopotentiometric experiment since here, the current is given and it is the electrode potential that is measured against time. Mathematically this model is defined by the usual equation (2.33), here with the boundary conditions

$$\begin{aligned} t = 0, \text{all } x: &\quad c = c^* \\ t > 0, x = 0: &\quad \mathrm{d}c/\mathrm{d}x = const\ . \\ \text{all } t, x \to \infty: &\quad c = c^*\ . \end{aligned} \tag{2.80}$$

The solution to this, that is, the concentration profile as a function of $x$ and $t$, is [1]

$$c(x,t) = c^* - \frac{i}{nFAD}\left\{2\sqrt{\frac{Dt}{\pi}}\exp\left(-\frac{x^2}{4Dt}\right) - x\,\mathrm{erfc}\left(\frac{x}{2\sqrt{Dt}}\right)\right\}\ , \tag{2.81}$$

where $i$ is the constant current that is applied, $D$ is the diffusion coefficient of the electroactive species. Some concentration profiles at three time values are shown in Fig. 2.6 and the constant concentration gradient at $x = 0$ can be seen. Also, the concentration $c(0,t)$ decreases with time $t$; it is in fact

$$c(0,t) = c^* - \frac{2i\sqrt{t}}{nFA\sqrt{\pi D}} \tag{2.82}$$

and reaches zero at some time, as shown in the figure. This time is the transition time (so named because the electrode potential undergoes a sharp transition at this point). It is given the symbol $\tau$ and is related to the current $i$ by the **Sand equation**:

$$\frac{i\sqrt{\tau}}{c^*} = \frac{nFA\sqrt{\pi D}}{2} \tag{2.83}$$

first given by Sand [31] and, with more detail, by Karaoglanoff [32].

To normalise this system, the previous definition (2.40) is used for the distance $x$, and $c^*$, the bulk concentration, for the concentration $c$; for the time unit, it is natural

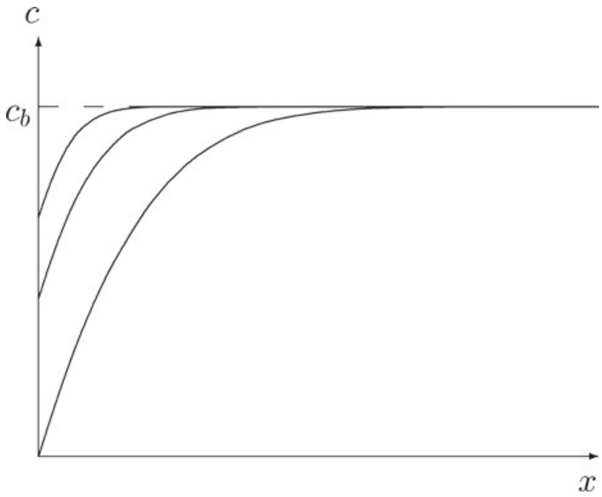


**Fig. 2.6** Concentration profile changing with time for chronopotentiometry

to use the transition time $\tau$ itself. This makes the boundary conditions

$$\begin{aligned} T = 0, \text{all } X: &\quad C = 1, \\ T > 0, X = 0\ : &\quad \mathrm{d}C/\mathrm{d}X = \tfrac{1}{2}\sqrt{\pi} \\ \text{all}, X \to \infty\ T: &\quad C = 1\ . \end{aligned} \tag{2.84}$$

Interestingly, here the constant current becomes the dimensionless constant concentration gradient at the electrode, with the value $\frac{1}{2}\sqrt{\pi}$. The dimensionless concentration profile is

$$C(X,T) = 1 - \sqrt{T}\left\{\exp\left(-\frac{X^2}{4T}\right) - \frac{\sqrt{\pi}X}{2\sqrt{T}}\,\mathrm{erfc}\left(\frac{X}{2\sqrt{T}}\right)\right\} \tag{2.85}$$

and, very simply,

$$C(0,T) = 1 - \sqrt{T}\ . \tag{2.86}$$

In Fig. 2.6, the profiles shown are for $t = 0.1\tau$, $t = 0.3\tau$ and $t = \tau$; that is, for $T = 0.1$, $T = 0.3$ and $T = 1$.

### 2.4.3 Linear Sweep Voltammetry

This is another useful system with which methods can be tested, one reason being that it demands more iterations than those mentioned above and is thus notoriously time-consuming. We again consider the simple reaction

$$A + ne^- \rightleftharpoons B \tag{2.87}$$

and assume reversibility. The electrode potential $E(t)$ is time-dependent,

$$E(t) = E_1 - vt \tag{2.88}$$

in which $E_1$ is the starting potential and $v$ is the scan rate in V s$^{-1}$. The diffusion equations are as for the potential step with a reversible system (2.18) with the boundary conditions, for the classical case,

$$\begin{aligned} t = 0: &\quad E = E_1 \\ t = 0, \text{all } x: &\quad c_A = c^*, c_B = 0 \\ t > 0: &\quad E(t) = E_1 - vt \\ t > 0, x = 0: &\quad c_A/c_B = \exp\left\{\tfrac{n\mathcal{F}}{\mathcal{RT}}(E(t) - E^0)\right\} \\ t > 0, x \to \infty: &\quad c_A = c^*, c_B = 0\ , \end{aligned} \tag{2.89}$$

where $c^*$ is the initial bulk concentration of species A and species B is not present initially. A common diffusion coefficient for both species, $D$, is assumed. In practice, the sweep terminates at some (more negative) potential $E_2$, but this is not part of the description. This system is interesting in that it was in fact the first to be simulated, by Randles, in 1948 [33] using hand calculations. In the same year, Ševčík [34] worked towards an analytical solution, ending in an integral equation he was forced to solve numerically. The current function is therefore called the Randles–Ševčík function. The integral equation was developed in 1964 by Nicholson and Shain [35] and solved numerically with greater accuracy. Their calculations were later improved by Oldham [36], Mocak [37] and Mocak and Bond [38] who used series solutions. The Oldham values have not been improved upon. The current function (which will be seen below to be the dimensionless flux for species A at the electrode), given the symbol $\chi$, was found by Oldham to have a peak value at the dimensionless potential $p_{max}$ [for the definition see (2.29)] of $-1.1090$, corresponding to $-28.493/n$ mV (at 25 °C and using the Diehl value [39] for the Faraday, 96486.0 C/mol), the peak $\chi$ (or $G$) value there being 0.44629. These numbers are useful to know as standards for comparing simulations, and refer only to the LSV case (that is, no reverse sweep is described).

To render the LSV system dimensionless, the usual reference values for concentration, time and distance from the electrode are needed, as well as that for potential (2.29) (and thus, sweep rate). Both species' concentrations are normalised by the initial bulk concentration of A, $c^*$, as always, and the potential to dimensionless $p$ as in (2.29), (2.88) thus becoming

$$p = p_1 - at \tag{2.90}$$

with $p_1$ being the dimensionless starting potential, and the variable $a$ given by

$$a = \frac{n\mathcal{F}}{\mathcal{R}\mathcal{T}} v \ , \tag{2.91}$$

the sweep rate (and now sweeping in the cathodic direction). A reference time $\tau$ can now be defined, being simply the time it takes to sweep through one $p$-unit,

$$\tau = a^{-1} \tag{2.92}$$

so that we have, as usual,

$$T = t/\tau \tag{2.93}$$

and also the reference distance $\delta$, as before,

$$\delta = \sqrt{D\tau} \tag{2.94}$$

and thus

$$X = x/\delta \tag{2.95}$$

so that we now have the two diffusion equations

$$\begin{aligned} \frac{\partial C_A}{\partial T} &= \frac{\partial^2 C_A}{\partial X^2} \\ \frac{\partial C_B}{\partial T} &= \frac{\partial^2 C_B}{\partial X^2} \end{aligned} \tag{2.96}$$

with the boundary conditions

$$\begin{aligned} T = 0 : &\quad p = p_1 \\ T = 0, \text{all } X : &\quad C_A = 1, C_B = 0 \\ T > 0 : &\quad p(T) = p_1 - T \\ T > 0, X = 0 : &\quad C_A/C_B = \exp(p(T)) \\ T > 0, X \rightarrow \infty : &\quad C_A = 1, C_B = 0 \ . \end{aligned} \tag{2.97}$$

Note the rather simple form of the Nernst equation here, and the fact that the dimensionless sweep rate is now unity, that is, one $p$-unit per one $T$-unit.

When solving this system by computer, the dimensionless results can then be translated back into dimensioned values. The above $\chi$ function is the same as dimensionless $\partial C_A/\partial X$ $(X = 0)$ or $G$, and becomes a real current via the equation

$$i(t) = n\mathcal{F}Ac_b\sqrt{\frac{n\mathcal{F}Dv}{\mathcal{RT}}}\,\chi \tag{2.98}$$

at the actual potential

$$E(t) = \frac{\mathcal{RT}}{n\mathcal{F}}\,p(t) + E^0\ . \tag{2.99}$$

With LSV, the quasireversible and irreversible cases might also be interesting models, both of which have mixed boundary conditions, lying somewhere between the extremes of Dirichlet and Neumann conditions, because here we have fluxes at the electrode, determined by heterogeneous rate constants (depending on potential) and concentrations at the electrode. Also, as will be seen in a later chapter, these models can give rise, with some simulation methods, to surprising instabilities. These models are described in the standard texts such as [1, 8].

The quasireversible LSV case was treated by Matsuda and Ayabe [40], who used a series sum as an approximation to the integral equation obtained from the Laplace-transform solution of the problem. The result depends on the heterogeneous rate constant, both the peak current and the peak potential varying with this parameter. Basha et al. [41] tried to improve on the results but it seems that those of Nicholson and Shain [35] were better. These also provided results for the totally irreversible case, first described by Delahay [42]. For this, the $\chi(at)$-function has a constant maximum, given to four figures in [1], 0.4958, from the tables in [35]. Peak potential varies with rate constant, as with the quasireversible case.

Thus, for these two cases, we do not have high-precision comparisons. In such a case, one recourse is to do convergence computations, going to finer and finer intervals until there is no further change in the values, which can then be used for reference. This of course rests on the assumption that one has a method that is guaranteed to work; and often, this is reasonable. One uses a tried-and-true method with such a guarantee but which is not necessarily highly efficient. The values are then used to check whatever method one is working on that might be more efficient. If this strikes the reader as somewhat unsatisfactory, note that, for LSV, there are in fact no analytical solutions at all, except those based on numerical methods of some kind, by either solving the associated integral equation numerically, or evaluating various series sums that are considered good approximations to the solution of such integral equations. It can be considered that digital simulation is one further method for doing this job.

Another case of interest with LSV is the catalytic EC' system, described above in the section on potential steps. The equations are the same except that the potential here is not constant nor very negative, following (2.29) in its dimensionless form. For small and intermediate rates of the homogeneous chemical reaction

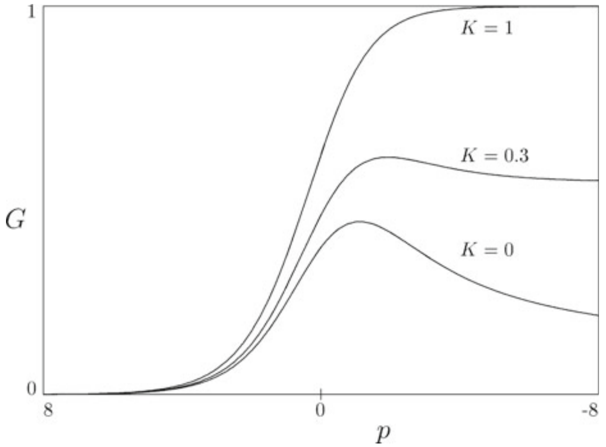


**Fig. 2.7** LSV curves, catalytic system

(dimensionless constant $K$), the same procedure as mentioned above, that is, convergence simulations, must be used. For large $K$, however, the LSV curves become sigmoid, with a plateau equal to the current for the potential step, $G = \sqrt{K}$. This can be used to test methods. Figure 2.7 shows LSV curves for some $K$-values, where this effect is seen.

Finally, it should be pointed out [43, 44], concentrations must be set to their equilibrium values prior to the sweep. This will most often be simple, as most sweeps start at a potential where the product is at zero concentration, but this may not always be the case.

## 2.5 Adsorption Kinetics

Adsorption is often present in electrochemical systems, both unintended and intended for certain purposes. The rate at which adsorbed layers form is of interest and must be simulated in most cases, as it is mathematically difficult, although a few simple cases have been solved. Modern texts [1, 8] give adsorption kinetics quite brief treatment, and the classic literature is rather old. Textbooks devoted to the subject (Damaskin et al. [45] or Jehring [46]) deal very briefly with the kinetics of adsorption. The recent focus is on self-assembled monolayers (SAMs; for a review, see, for example, [47]), and the kinetics of their formation are complicated by changes that take place after adsorption [1]. It used to be thought that the adsorption step itself is fast on mercury, but slower on solid metals [8]. The insight gained from the study of SAMs suggests that on solid metals, too, adsorption as such is fast, but the rearrangement that takes place afterwards, is a slow process [47].

There are some important general relations for a substance adsorbed from solution on an electrode. These pertain to the equilibrium state and the kinetics of the process leading to equilibrium. Adsorption kinetics receives rather intermittent attention in the electrochemical literature. One of the clearest discussions is by Mohilner [48]; see also Delahay [49], Bard and Faulkner [1].

The degree of adsorption is expressed either by $\varGamma$, the surface concentration, in units of moles per unit area, or in terms of the fractional coverage $\theta$:

$$\theta = \varGamma / \varGamma_m \,, \tag{2.100}$$

where $\varGamma_m$ is the maximum possible surface concentration at saturation, in many cases corresponding to a complete monolayer of the substance on the electrode. At equilibrium, $\varGamma$ or $\theta$ are related to the adsorbed substance's concentration $c_0$ adjacent to the electrode, by the adsorption isotherm $I$, customarily written in the inverse form

$$bc_0 = I(\theta) \tag{2.101}$$

or, in dimensionless terms,

$$BC_0 = I(\theta) \tag{2.102}$$

with $B = bc^*$. It may be that while the state of adsorption is in momentary equilibrium with $c_0$, there are nevertheless concentration gradients in the solution. The isotherms take many forms; a large list was presented by Mohilner [48]. A few examples are

$$I(\theta) = \theta \tag{2.103}$$

which is the linear or Henry isotherm, sometimes applicable for $\theta \ll 1$; or the Langmuir isotherm

$$I(\theta) = \frac{\theta}{1-\theta} \,. \tag{2.104}$$

If there is interaction between the particles (attractive or repulsive), the Frumkin isotherm may apply:

$$I(\theta) = \frac{\theta}{1-\theta} \exp(-2a\theta) \tag{2.105}$$

with $a$ the attraction parameter. The logarithmic Temkin isotherm [48] is

$$I(\theta) = \exp(a\theta) \,. \tag{2.106}$$

There are other, more complicated isotherms but the above examples suffice.

In order to reach a certain surface concentration $\Gamma$ or fractional coverage $\theta$, the substance in question must first arrive at the electrode by some transport process. The diffusion equation applies for this part. The rate of increase of $\Gamma$ (per unit area) is proportional to the unit area flux at the electrode

$$\frac{d\Gamma}{dt} = -f \tag{2.107}$$

(flux being regarded as away from the electrode). From (2.2) this becomes

$$\frac{d\Gamma}{dt} = D \left.\frac{\partial c}{\partial x}\right|_{x=0} . \tag{2.108}$$

We prefer to work with $\theta$ and so get [using (2.100)]

$$\frac{d\theta}{dt} = D \left.\frac{\partial c}{\partial x}\right|_{x=0} \Gamma_m . \tag{2.109}$$

In dimensionless form, using the transformations (2.25) as before, this is

$$\frac{d\theta}{dT} = \frac{c^* \sqrt{D\tau}}{\Gamma_m} G , \tag{2.110}$$

$\tau$ being the observation time as before. Some model systems with solutions exist for adsorption kinetics. The simplest case is that of a very large $B$ parameter in (2.102), and a Cottrellian experiment. For $B \rightarrow \infty$, there is then the set of boundary conditions

$$\begin{aligned} T = 0 : &\quad C(X,T) = 1; \quad \theta = 0 \\ T > 0 : &\quad C(0,T) = 0; \quad \theta = \tfrac{c^* \sqrt{D\tau}}{\Gamma_m} \int_0^T G \, \mathrm{d}T \\ \text{all } T, X \rightarrow \infty : &\quad C = 1 . \end{aligned} \tag{2.111}$$

In other words, adsorption is so strong, that every particle of the substance arriving at the electrode is adsorbed immediately, forcing the concentration there, in solution, to zero. So the adsorbate simply accumulates by the Cottrellian flux given by the potential step experiment, and the solution, given first by Koryta [50], is

$$\theta(T) = \frac{2c_b \sqrt{D\tau}}{\Gamma_m \sqrt{\pi}} \sqrt{T} . \tag{2.112}$$

If adsorption is fast but not sufficiently strong to justify the assumption $C_0 \approx 0$, then $C_0$ will, at any instant, be determined by the adsorption isotherm (2.102). This boundary condition leads to mathematical problems; the integral equation resulting from (2.109) then becomes a Volterra equation. This has been solved for only some very simple isotherms. Delahay and Trachtenberg [51] solved it for the Henry isotherm (2.103), the solution being

$$C = 1 - \exp\left(\frac{X}{K} + \frac{T}{K^2}\right) \operatorname{erfc}\left(\frac{X}{2\sqrt{T}} + \frac{\sqrt{T}}{K}\right) , \tag{2.113}$$

in which the dimensionless $K = b\Gamma_m/\sqrt{D\tau}$.

Reinmuth [52] arrived at a solution for the Langmuir isotherm in the form of a series, involving the beta function. Levich et al. [53] had an approximate solution for a general adsorption isotherm.

If the adsorption step itself is rate-limiting, one must have available rate expressions for the adsorption and the desorption steps. The flux in (2.107) is then split into two opposing components. Using the notation of Delahay and Mohilner [48, 54], there is a forward flux $v_f$, adding to the adsorbate's surface concentration and backward flux $v_b$, the rate of redissolution of adsorbed substance. These obey rate equations rather analogous to those for electron transfer, the Butler–Volmer equation, in the sense that there are rate constants that are potential dependent. For the forward and backward rates, we have

$$\begin{aligned} v_f &= k_f f_1(c_0, \Gamma) \\ v_b &= k_b f_2(\Gamma) , \end{aligned} \tag{2.114}$$

where $k_f$ and $k_b$ are the forward and backward rate constants and $f_1$ and $f_2$ are functions whose forms depend on the adsorption isotherm assumed. Note that the two constants have different units. If the forward and backward rates are equal, then there is equilibrium and the isotherm is obtained by equating the right-hand sides of (2.114). If the rates are not equal, there will be a net flux of substance between the two phases.

Various workers [55–59] have presented particular forms of the functions $f_1$ and $f_2$, for various isotherms. For all these equations, one needs to know the $k$ parameters and possibly more. In all cases, we have

$$\frac{d\Gamma}{dt} = v_f - v_b \tag{2.115}$$

which we would usually normalise to the dimensionless form

$$\frac{d\theta}{dT} = V_f - V_b \tag{2.116}$$

with $V = v\tau/\Gamma_m$, $\tau$ being, as before, a chosen experimental time scale.

Two examples are given here, to be followed up in Chap. 10. Lovrić and Komorsky-Lovrić [58] expressed (2.114) for the simple Henry isotherm (2.103), as

$$\begin{aligned} v_f &= k_f c_0 \\ v_b &= k_b \Gamma \ , \end{aligned} \tag{2.117}$$

which is linear in both $c_0$ and $\Gamma$, while Lorenz [60] expressed the equation for the Langmuir isotherm (2.104) as

$$\begin{aligned} v_f &= k_f c_0 (\Gamma_m - \Gamma) \\ v_b &= k_b \Gamma \end{aligned} \tag{2.118}$$

where we now have a nonlinear term in the first equation. These can be normalised conveniently. The $v$ are normalised to $V$ as above. Letting $K_f = k_f \tau c^* / \Gamma_m$ and $K_b = k_b \tau c^*$ (note the different normalisations of the two rate constants), we obtain

$$\begin{aligned} V_f &= K_f C_0 \\ V_B &= K_b \theta \end{aligned} \tag{2.119}$$

for the Henry isotherm, and

$$\begin{aligned} V_f &= K_f C_0 (1 - \theta) \\ V_b &= K_b \theta \end{aligned} \tag{2.120}$$

for the Langmuir isotherm. These two cases—one linear and the other nonlinear—are good examples for the simulation process. More complex isotherms such as the Frumkin isotherm also lead to nonlinear equations, so that this one nonlinear example suffices to point the way. How all this is simulated is described in Chap. 10.

## References

1. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
2. Crank J (1975) The mathematics of diffusion, 2nd edn. Clarendon Press, Oxford
3. Ghez R (1988) A primer of diffusion problems. Wiley, New York
4. Fourier F (1822) Théorie Analytique de la Chaleur, vol I. Didot, Pere et Fils, Paris
5. Fick A (1855) Ueber diffusion. Pogg Ann 94:59–86
6. Vetter K (1961) Elektrochemische Kinetik. Springer, Berlin
7. Bauer HH (1972) Electrodics. Thieme, Stuttgart
8. Galus Z (1994) Fundamentals of electrochemical analysis, 2nd edn. Ellis Horwood, New York (trans: Chalmers RA, Bryce WAJ (eds))
9. Wiesner K (1947) Pokus o výpočet absolutnich hodnot rychlostnich konstant pro disociaci slabých kyselin. Chem Listy 41:6–8
10. Inzelt G (2011) Milestones of the development of kinetics of electrode reactions. J Solid State Electrochem 15:1373–1389
11. Zeng Y, Smith RB, Bai P, Bazant MZ (2014) Simple formula for Marcus-Hush-Chidsey kinetics. J Electroanal Chem 735:77–83

12. Bieniasz LK (2012) A procedure for rapid and highly accurate computation of Marcus-Hush-Chidsey rate constants. J Electroanal Chem 683:112–118
13. Feldberg SW (2010) Implications of Marcus-Hush theory for steady-state heterogeneous electron transfer at an inlaid disk electrode. Anal Chem 82:5176–5183
14. Henstridge MC, Rees NV, Compton RG (2012) A comparison of the Butler-Volmer and asymmetric Marcus-Hush models of electrode kinetics at the channel electrode. J Electroanal Chem 687:79–83
15. Cottrell FG (1903) Der Reststrom bei galvanischer Polarisation, betrachtet als ein Diffusionsproblem. Z Phys Chem 42:385–431
16. Abramowitz M, Stegun IA, editors (1969) Handbook of mathematical functions. Dover, New York
17. Nernst W (1904) Theorie der Reaktionsgeschwindigkeit in heterogenen Systemen. Z Phys Chem 47:52–55
18. Brunner E (1904) Reaktionsgeschwindigkeit in heterogenen Systemen. Z Phys Chem 47:56–102
19. Feldberg SW (1969) Digital simulation: a general method for solving electrochemical diffusion-kinetic problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 3. Marcel Dekker, New York, pp 199–296
20. Britz D, Østerby O, Strutwolf J (2008) Comparison of flux approximations in electrochemical digital simulation. Part 2: complications due to homogeneous chemical reactions, charge estimation and application to the ultramicrodisk electrode. J Electroanal Chem 622:51–58
21. Britz D, Østerby O, Strutwolf J (2010) Reference values of the chronoamperometric response at cylindrical and capped cylindrical electrodes. Electrochim Acta 55:5629–5635
22. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376
23. Doetsch G (1967) Anleitung zum praktischen Gebrauch der Laplace-Transformation und der Z-Transformation. Oldenburg Verlag, München, Wien
24. Bard AJ, Mirkin MV (2001) Scanning electrochemical microscopy. Marcel Dekker, New York
25. Reinert KE, Berg H (1962) Theorie der polarographischen Verfolgung schneller chemischer Reaktionen in Lösung mittels reaktionsbedingter Diffusions-Zeit-Kurven. Monatsber Deut Akad Wiss Berlin 4:26–32
26. Birk JR, Perone SP (1968) Electrochemical studies of rapid photolytic processes. A theoretical and experimental evaluation of potentiostatic analysis in flash photolysed solutions. Anal Chem 40:496–500
27. Britz D, Kastening B (1974) On the electrochemical observation of a second-order decay of radicals generated by flash photolysis or pulse radiolysis. J Electroanal Chem 56:73–90
28. Delahay P, Stiehl GL (1952) Theory of catalytic polarographic currents. J Am Chem Soc 74:3500–3505
29. Miller SL (1952) Polarographic currents from a combination of diffusion and reaction. J Am Chem Soc 74:4130–4134
30. Saveant JM, Vianello E (1965) Potential-sweep chronoamperometry: kinetic currents for first-order chemical reaction parallel to electron-transfer process (catalytic currents). Electrochim Acta 10:905–920
31. Sand HJS (1900) Über die Konzentration an den Elektroden in einer Lösung mit besonderer Berücksichtigung der Wasserstoffentwicklung durch Elektrolyse einer Mischung von Kupfersulfat und Schwefelsäure. Z Phys Chem 35:641–651
32. Karaoglanoff Z (1906) Über Oxidations- und Reduktionsvorgänge bei der Elektrolyse von Eisensalzen. Z Elektrochem 12:5–16
33. Randles JEB (1948) A cathode-ray polarograph. Part II - the current-voltage curves. Trans Faraday Soc 44:327–338
34. Ševčík A (1948) Oscillographic polarography with periodical triangular voltage. Collect Czechoslov Chem Commun 13:349–377
35. Nicholson RS, Shain I (1964) Theory of stationary electrode polarography. Single scan and cyclic methods applied to reversible, irreversible, and kinetic systems. Anal Chem 36:706–723

36. Oldham KB (1979) Analytical expressions for the reversible Randles-Sevcik function. J Electroanal Chem 105:373–375
37. Mocak J (2002) Voltammetric current-potential calculations using infinite series solution. Electrochem Commun 4:803–807
38. Mocak J, Bond A (2004) Use of MATHEMATICA software for theoretical analysis of linear sweep voltammograms. J Electroanal Chem 561:191–202
39. Diehl H, Biggs DL (1983) The density of 4-aminopyridine, voids in crystals, and precision weighing. Talanta 30:894–898
40. Matsuda H, Ayabe Y (1955) Zur Theorie der Randles-Sevčikschen Kathodenstrahl-Polarographie. Z Elektrochem 59:494–503
41. Basha CA, Sangaranarayanan MV (1989) On the evaluation of the current function in linear sweep voltammetry. J Electroanal Chem 261:431–436
42. Delahay P (1953) Theory of irreversible waves in oscillographic polarography. J Am Chem Soc 75:1190–1196
43. Amatore C, Klymenko OV, Svir I (2012) Importance of correct prediction of initial concentrations in voltammetric scans: contrasting roles of thermodynamics, kinetics, and natural convection. Anal Chem 84:2792–2798
44. Luo W, Feldberg SW, Rudolph M (1994) Ensuring self-consistent assignment of thermodynamic parameters in simulations of electrochemical-chemical systems. J Electroanal Chem 368:109–113
45. Damaskin BB, Petrii OA, Batrakov VV (1975) Adsorption organischer Verbindungen an Elektroden. Akademie-Verlag, Berlin
46. Jehring H (1974) Elektrosorptionsanalyse mit der Wechselstrompolarographie. Akademie-Verlag, Berlin
47. Finkley HO (1996) Electrochemistry of organized monolayers of thiols and related molecules on electrodes. In: Bard AJ, Rubinstein I (eds) Electroanalytical chemistry, vol 19. Marcel Dekker, New York, pp 109–335
48. Mohilner DM (1966) The electrical double layer. I. Elements of the double-layer theory. In: Bard AJ (ed) Electroanalytical chemistry, vol 1. Marcel Dekker, New York, pp 241–409
49. Delahay P (1966) Double layer and electrode kinetics. Interscience, New York
50. Koryta J (1953) Über den Einfluss der Farbstoffe der Eosin-Gruppe auf die reversible Oxydo-Reduktion an der tropfenden Qecksilberelektrode. Collect Czechoslov Chem Commun 18:206–213
51. Delahay P, Trachtenberg I (1957) Adsorption kinetics and electrode processes. J Am Chem Soc 79:2355–2362
52. Reinmuth WH (1961) Diffusion to a plane with Langmuirian adsorption. J Phys Chem 65:473–476
53. Levich VG (1962) Physicochemical hydrodynamics. Prentice-Hall, Englewood Cliffs, NJ
54. Delahay P, Mohilner DM (1962) Rate equation for adsorption of a neutral substance at a metal-electrolyte interface. J Am Chem Soc 84:4247–4252
55. Armstrong RD, Race WP, Thirsk HR (1968) The kinetics of adsorption of neutral organic compounds at a mercury electrode. J Electroanal Chem 16:517–529
56. Baret JF (1968) Kinetics of adsorption from a solution. Role of the diffusion and of the adsorption-desorption antagonism. J Phys Chem 72:2755–2758
57. Engelman EE, Evans DH (1992) Explicit finite-difference digital simulation of the effects of rate-controlled product adsorption or deposition in double-potential-step chronocoulometry. J Electroanal Chem 331:739–749
58. Lovrić M, Kormorsky-Lovrić Ŝ (1981) O adsorpciji kontroliranoj difuzijom. Bull Soc Chim Beograd 46:93–98
59. Miller R (1980) Zur Adsorptionskinetik an der Oberfläche wachsender Tropfen. Colloid Polym Sci 258:179–185
60. Lorenz W (1958) Über die Geschwindigkeit der Adsorption und der zweidimensionalen Assoziation höherer Fettsäuren an der Grenzfläche Quecksilber-Elektrolytlösung. Z Elektrochem 62:192–200

# Chapter 3
# Approximations to Derivatives

In this chapter, all the discrete approximations required for simulation are established, that is, for first and second derivatives, both central and asymmetric forms, equally or randomly spaced points, and for a range of numbers of points used.

## 3.1 Approximation Order

Consider Fig. 3.1 and assume that the intervals between the marked points on the $x$-axis are equidistant with length $h$ between them. For the moment, consider a simple first derivative "around the region" between $x_1$ and $x_2$ (to be made more precise in later sections). The function shown as a curve is known only in the form of the fat points on it. Intuitively, one thinks of the approximation in that region as

$$\frac{\mathrm{d}y}{\mathrm{d}x} = \frac{y_2 - y_1}{h} \tag{3.1}$$

and this can be used to define the concept of *order*. The above expression is that for the slope of the straight line drawn from the first point (at $x_1$) to the second (at $x_2$), and is slightly in error. The error depends on where on the curve we mean the expression to apply. In later sections all this will be developed precisely, but here it suffices to say that the error will usually be a function of the interval $h$, and will become smaller, as we make $h$ smaller. The order then tells us how much smaller. The relation between the error $e$ and $h$ is approximately given by

$$e = const \times h^p \tag{3.2}$$

and the power $p$ is the order (while the constant is of lesser importance). For example, it is seen below that, if we mean (3.1) to apply at the position $x_1$ or $x_2$, then $p$ is equal to unity, and one says that the approximation is first-order with respect

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_3

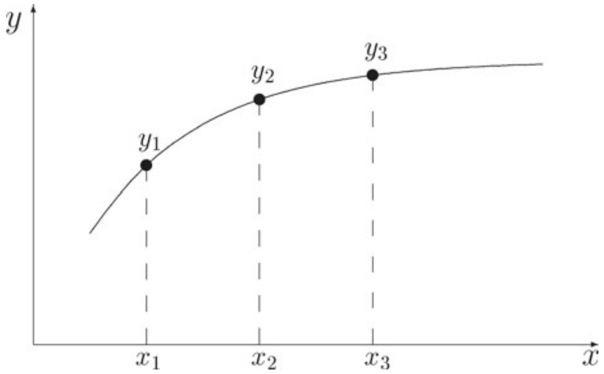


**Fig. 3.1** Arbitrary function with three points marked

to $h$, expressed as $O(h)$. In other words, if we attempt to make the error smaller by halving $h$, then we also halve the error. If we mean the approximation to apply to the point midway between the interval, that is, at $x = (x_1 + x_2)/2$, it turns out that $p = 2$, and the error is $O(h^2)$. Thus, if we halve $h$, the error becomes a quarter as large. This is better than first-order and generally, one seeks approximations of high order.

In digital simulation, when discretising the diffusion equation, we have a first derivative with respect to time, and one or more second derivatives with respect to the space coordinates; sometimes also spatial first derivatives. Efficient simulation methods will always strive to maximise the orders.

## 3.2 Two-Point First Derivative Approximations

Consider again Fig. 3.1, and the point at $x_2$, expressed as a Taylor series development going from $x_1$. Note that the symbols $y_2$, $f(x_2)$ and $f(x_1 + h)$ are all synonymous. The Taylor expansion is

$$y_2 = y_1 + hy'_1 + \frac{h^2}{2!}y''_1 + \frac{h^3}{3!}y'''_1 + \ldots \tag{3.3}$$

where $y'_1$ etc. are the progressively higher spatial derivatives of the function at the point $(x_1, y_1)$. This equation can be rearranged to

$$y'_1 = \frac{y_2 - y_1}{h} - \frac{h}{2!}y''_1 - \frac{h^2}{3!}y'''_1 - \ldots \tag{3.4}$$

where the first term on the right-hand side is in fact (3.1) above. However, now we know more about this approximation: we note that it refers to the point $(x_1, y_1)$, and that if we write

$$y_1' = y'(x_1) = \frac{y_2 - y_1}{h} \tag{3.5}$$

then this has an error equal to the sum of the further terms on the right-hand side of (3.4). That is, the error is

$$e = -\frac{h}{2!}y_1'' - \frac{h^2}{3!}y_1''' - \dots\ . \tag{3.6}$$

This is a polynomial in $h$, and since $h$ is normally rather small, the lowest power in $h$ will contribute most to the sum. Thus we see that the error is $O(h)$ (the actual coefficients do not matter as much as the order).

The approximation (3.5) is called a *forward difference* because the values used to approximate it lie forward of the point $(x_1)$ where it is meant to apply.

It is possible to develop the point at $x_2$ going backward, again using the Taylor expansion:

$$y_1 = y_2 - hy_2' + \frac{h^2}{2!}y_2'' - \frac{h^3}{3!}y_2''' + \dots \tag{3.7}$$

where there is now an alternation of sign because of the negative value $-h$. Rearranging this yields

$$y_2' = \frac{y_2 - y_1}{h} + \frac{h}{2!}y_2'' - \frac{h^2}{3!}y_2''' + \dots \tag{3.8}$$

giving the approximation

$$y_2' = y'(x_2) = \frac{y_2 - y_1}{h} \tag{3.9}$$

which is also $O(h)$ (with different polynomial coefficients) and, as it refers to the point $(x_2, y_2)$ using a preceding other point, is a *backward difference*.

It will now become clear why Fig. 3.1 has three points on it. We focus on the point at $x_2$ and use Taylor's expansions around this point for both $y_1$ [see (3.7)] and $y_3$:

$$y_3 = y_2 + hy_2' + \frac{h^2}{2!}y_2'' + \frac{h^3}{3!}y_2''' + \dots\ . \tag{3.10}$$

Subtracting (3.7) from (3.10) and rearranging yields

$$y_2' = \frac{y_3 - y_1}{2h} + \frac{h^2}{3!}y_2''' + \dots \tag{3.11}$$

that is,

$$y'_2 = \frac{y_3 - y_1}{2h} + O(h^2) \tag{3.12}$$

a second-order *central difference* approximation to the first derivative, which is much better than either the forward or backward formulae in the above. Clearly, we could have done this, focussing on a point midway between $x_1$ and $x_2$ (let us call it $x_{1.5}$) and Taylor-expanding around it for the two points $y_1$ and $y_2$, thus arriving at the approximation, $y_{1.5}$ at $x_{1.5}$,

$$y'_{1.5} = \frac{y_2 - y_1}{h} + O(h^2) \tag{3.13}$$

which is also second-order with respect to $h$.

We now have three two-point approximations for a first derivative, all in fact being the same expression, $(y_2 - y_1)/h$, but depending on where this formula is intended to apply, being, respectively a forward difference of $O(h)$ if applied at $x_1$, a backward difference of $O(h)$ if applied at $x_2$ and a central difference of $O(h^2)$ if applied at $(x_1 + x_2)/2$. In subsequent chapters, all these will be used to approximate, among others, the various derivatives in Eqs. (2.3)–(2.8) on page 7.

## 3.3 Multi-Point First Derivative Approximations

The above approximations to a first derivative used only two points, which sets a limit on the approximation order. By using more points, higher-order approximations can be achieved. In the context of this book, forward and backward multi-point formulae are of special interest, as well as some asymmetric and central multi-point ones. To this end, a notation will be established here. Figure 3.2 shows the same

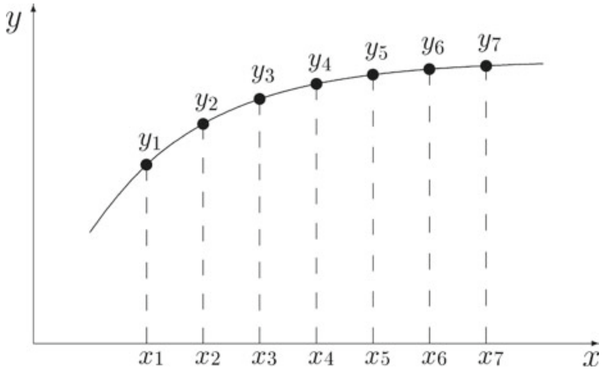


**Fig. 3.2** Arbitrary function with seven points marked

curve as Fig. 3.1 but now seven points are marked on it. The notation to be used is as follows. If a derivative is approximated using the $n$ values $y_1 \dots y_n$, lying at the $x$-values $x_1 \dots x_n$ (intervals $h$) and applied at the point$(x_i, y_i)$, then it will be denoted as $y_i'(n)$ (for a first derivative) and $y_i''(n)$ (for a second derivative).

For a given number $n$ of points to be included in an approximation for $y'$ applied at point index $i$ within the group, the procedure is to calculate the $\beta$ coefficients in the general expression

$$y_i'(n) = \frac{1}{h}\sum_{i=1}^{n} \beta_i y_i \,. \tag{3.14}$$

This is done by writing the Taylor expansions around the point at index $i$ for all the other $(n-1)$ points, to a sufficient number of terms, and solving for the derivatives, discarding all but the numbers for the first derivative. A single example will illustrate the method. Assume that we want $y_2'(4)$, that is, the derivative $y'$ at point $(x_2, y_2)$ out of points at $x_1 \dots x_4$ in Fig. 3.2. Taylor expansions are written for points at $x_1, x_3, x_4$, going to the third derivative (in general, for $n$ points, to the $(n-1)$st derivative):

$$\begin{aligned} y_1 &= y_2 - hy_2' + \frac{h^2 y_2''}{2!} - \frac{h^3 y_2'''}{3!} + O(h^4) \\ y_3 &= y_2 + hy_2' + \frac{h^2 y_2''}{2!} + \frac{h^3 y_2'''}{3!} + O(h^4) \\ y_4 &= y_2 + 2hy_2' + \frac{4h^2 y_2''}{2!} + \frac{8h^3 y_2'''}{3!} + O(h^4) \,. \end{aligned} \tag{3.15}$$

The above can be rewritten as a matrix equation,

$$\begin{bmatrix} -1 & \frac{1}{2!} & -\frac{1}{3!} \\ 1 & \frac{1}{2!} & \frac{1}{3!} \\ 2 & \frac{4}{2!} & \frac{8}{3!} \end{bmatrix} \begin{bmatrix} h & 0 & 0 \\ 0 & h^2 & 0 \\ 0 & 0 & h^3 \end{bmatrix} \begin{bmatrix} y_2' \\ y_2'' \\ y_2''' \end{bmatrix} = \begin{bmatrix} y_1 - y_2 \\ y_3 - y_2 \\ y_4 - y_2 \end{bmatrix} \tag{3.16}$$

(remembering the $O(h^4)$ terms but not writing them). This can be written as

$$\mathbf{AHd} = \mathbf{b} \tag{3.17}$$

where $\mathbf{A}$ is the main matrix, $\mathbf{H}$ the diagonal matrix of terms in $h$, $\mathbf{d}$ the solution vector of derivatives $\left[y_2'\ y_2''\ y_2'''\right]^T$ and $\mathbf{b}$ the right-hand side vector of knowns. The next step is to multiply by the inverses of the two left-hand matrices

$$\mathbf{d} = \mathbf{A}^{-1}\mathbf{H}^{-1}\mathbf{b} \,. \tag{3.18}$$

All that is wanted here is the top row of the inverted matrix, since

$$y_2' = h^{-1}\begin{bmatrix}c_{11} & c_{12} & c_{13}\end{bmatrix}\begin{bmatrix}y_1 - y_2\\ y_3 - y_2\\ y_4 - y_2\end{bmatrix} \tag{3.19}$$

(with $c_{11}$ etc. being the first row elements of the inverse $\mathbf{C} = \mathbf{A}^{-1}$). When inverting matrix $\mathbf{A}$, the numbers come out as decimal fractions, in this case $\begin{bmatrix}-\frac{1}{3} & 1 & -\frac{1}{6}\end{bmatrix}$. We prefer whole-number fractions and, in the case of equidistant points, these exist. It is an easy programming job to find a multiplier that makes whole numbers out of all entries in the top row of $\mathbf{A}^{-1}$; in this case, it is 6 and the result of the computation is

$$y_2' = \frac{1}{6h}\begin{bmatrix}-2 & 6 & -1\end{bmatrix}\begin{bmatrix}y_1 - y_2\\ y_3 - y_2\\ y_4 - y_2\end{bmatrix} \tag{3.20}$$

which, when multiplied out and after sorting, gives the result

$$y_2' = \frac{1}{6h}\{-2y_1 - 3y_2 + 6y_3 - y_4\} + O(h^3) \tag{3.21}$$

in which the order term indicates that this approximation is third-order with respect to $h$.

In this way, the coefficients for any $y_i'(n)$ can be calculated. Table A.1 in Appendix A shows a number of these, as whole numbers $m\beta_i$, where $m$ is the multiplier mentioned above. For each $n$, the table shows forward differences (at index 1), backward derivatives (at index $n$) and derivatives applying at points between the two ends. In case the reader wonders why all this is of interest: the forms $y_1'(n)$ will be used to approximate the current or, in general, the concentration gradient, in simulations (see the next section); the backward forms $y_n'(n)$ will be used in the section on the BDF method in Chaps. 4 and 9, and the intermediate forms shown in the table will be used for the Kimble and White (high-order) start of the BDF method, also described in these chapters. The coefficients have a long history. Collatz [1] derived some of them in 1935 and presents more of them in [2]. Bickley tabulated a number of them in 1941 [3]. The three-point current approximation, essentially $y_1'(3)$ in the present notation, was first used in electrochemistry by Randles [4] (preempted by 2 years by Eyres et al. [5] for heat flow simulations), then by Heinze et al. [6]; Newman [7, p. 554] used a five-point current approximation, and schemes of up to seven-point were provided in [8].

## 3.4 The Current Approximation

As shown in Chap. 2, Eq. (2.26), the current in its dimensionless form $G$ is the dimensionless gradient of $C$ with respect to $X$ at $X = 0$. This implies that a forward difference must be used, as we normally have $C$-values starting at $X = 0$. There are algorithms with points at negative $X$ values, but they are not generally very successful or popular. The approximation can therefore be expressed as the $n$-point approximation

$$G \approx \frac{1}{H} \sum_{i=0}^{n-1} \beta_i C_i \,. \tag{3.22}$$

The symbol $H$ is the interval along the normalised spatial axis. The symbol $G_n$ will sometimes be used, to mark $n$, the number of points used. The simplest formula is the two-point form,

$$G \approx \frac{1}{H}(C_1 - C_0) \tag{3.23}$$

This seems a poor, low-order approximation. It can be justified, however, in cases where $H$ is very small, as is in fact so with most useful programs these days, since these use unequal intervals, usually spaced very closely near the electrode. As will be seen, this two-point form makes the discretisation of boundary conditions much easier. There are even cases in which the current approximation becomes worse as more points are introduced. This happens with severely stretched grids (see “unequal intervals”, elsewhere), so the $n$-point formula should probably be used only with equal intervals. It has also been argued [9] that the three-point formula for equal intervals,

$$G \approx \frac{1}{2H}(-3C_0 + 4C_1 - C_2) \tag{3.24}$$

is most compatible with the usual three-point second-order approximation to the second space derivative, being itself second-order. This is a matter of taste.

## 3.5 The Current Approximation Function $\mathcal{G}$

The above (3.22) is now generalised to operate on any array or vector $\mathbf{v} = [v_0, v_1, \ldots, v_{n-1}]^T$, that is, we define the function

$$\mathcal{G}(\mathbf{v}, n, H) = \frac{1}{H} \sum_{i=0}^{n-1} \beta_i v_i \tag{3.25}$$

which will be used extensively in this book. Mostly, the second and third arguments will be taken as understood, and the function will then simply be written as $\mathcal{G}(\mathbf{v})$ for the general vector $\mathbf{v}$, which in many cases will be concentration $C$ (but, as will be seen in Chap. 6, not always). The function is implemented in the Fortran function `G0FUNC` and is included in the example routines in Appendix E.

### *3.5.1 Unequal Intervals*

Equation (3.25) holds for equal intervals but also for arbitrary (unequal) intervals, if the coefficients are computed accordingly. For unequal intervals, the coefficients must be computed (probably precomputed in a given program), as they cannot be tabulated, and this is best done using the Fornberg algorithm [10], to be described in the later Chap. 7. It is implemented in the routine `G0FORN`, also described in Appendix E.

## 3.6 High-Order Compact (Hermitian) Current Approximation

There is a trick by which one can increase the order, and thereby the accuracy, of current approximations for a given number of points used. It is related mathematically to the Numerov method, to be discussed in a later chapter. The device is based on the particular form of the *pde* that we are trying to solve for. It was introduced to electrochemistry by Bieniasz [11, 12], referring to some earlier work in other fields. The device belongs to a class of schemes given various names, among them "compact stencil" or "high-order compact (HOC) scheme", or Hermitian discretisation. The latter term is possibly the best. It refers to Hermite's interpolation method, clearly described by Kopal [13]. Its essence is the use in an approximation, not only of function values at grid points but also function derivatives. This is now generally applied in other contexts outside interpolation. The English translation of Collatz' book [2] uses the term to translate the original "Mehrstellenverfahren" and notes that this does not imply that Hermite used the method in this way. The Hermitian method can not only be used to obtain better current approximations, but also for simulations with derivative boundary conditions, to be described in Chap. 9. An example of a Hermitian method is the Numerov method [14], also described in Chap. 9.

The information on derivatives that the device makes use of is the *pde* itself, which can be written in the form

$$\frac{\partial^2 C}{\partial X^2} = F(X, T, C, \partial C/\partial T) \tag{3.26}$$

writing it out for a normalised equation for simplicity (Bieniasz makes it very general, as a system of such equations, each one with its own diffusion coefficient). The $F$ term always contains the time derivative, but may also contain, for example, homogeneous chemical terms in concentration. In what follows, the function will be simply written as $F_i$, where it is understood that this refers to the point at $X = iH$.

First we consider the current approximation presented in the above two sections. A question left untouched, for example, the equation for the current approximation (3.25) above, is just what terms were dropped when generating a particular form. The order of what was dropped is given in Sect. 3.3, but not extended to actual higher terms. This must be done now. Bieniasz [11] presents a table of these and we can write the first few. For this, it is convenient to use a more compact notation for the higher derivatives: let

$$D_X^k \equiv \left.\frac{\partial^k C}{\partial X^k}\right|_X \tag{3.27}$$

and recall that $G_n$ denotes a current approximation using $n$ points as defined above (3.25), that is, neglecting higher terms. This gives us, for $n = 2$,

$$D_0^1 = G_2 - \tfrac{1}{2}HD_0^2 - \tfrac{1}{6}H^2D_0^3 - \dots \tag{3.28}$$

and for $n = 3$,

$$D_0^1 = G_3 + \tfrac{1}{3}H^2D_0^3 + \tfrac{1}{4}H^3D_0^4 + \dots \tag{3.29}$$

and so on for the higher-$n$ forms. An extended table is seen in Bieniasz's paper [11], but these two will suffice here. In order to improve the two approximations $G$, clearly we need information on the higher derivatives.

One further new notation is useful here, used by Bieniasz. A given current approximation is denoted as $n(m)$, where $n$ is the number of points used to approximate it, and $m$ is the order with respect to $H$, the intervals in $X$. Thus, the formulae used so far make, for example, $G_2$ a 2(1) form and $G_3$ a 3(2) form. It will be seen that we can easily obtain, for example, 2(3) and 3(4), etc.

In order to obtain the missing higher derivatives, or approximations to them, we write (3.26) for $X = 0$:

$$\left.\frac{\partial^2 C}{\partial X^2}\right|_0 = D_0^2 = F(0, T, C_0, \partial C_0/\partial T) = F_0\ . \tag{3.30}$$

This can be applied directly to (3.28), neglecting the term in $D_0^3$ there and replacing the term in $D_0^2$ as in (3.30), obtaining a new approximation,

$$D_0^1 \approx G_2 - \tfrac{1}{2}HF_0 \tag{3.31}$$

which is 2(2), an improvement on the old form. Bieniasz' treatment results in a general equation that can be written as

$$D_0^1 = G_n + H \sum_{i=0}^{n-1} \phi_i F_i \tag{3.32}$$

or, in words, with the old formulae of the above two sections, all $n(n-1)$, improved by the addition of (up to) an equal number $n$ of $F_i$ values with weighting coefficients $\phi_i$. In the above simple 2(2) example, we have $\phi_o = -\frac{1}{2}, \phi_1 = 0$. In the table of $\phi$ coefficients in Bieniasz [11], it is seen that for all $n$, there is a set of coefficients that give $n(n)$, and they all have the last one, $\phi_{n-1}$, equal to zero. These all fail to make use of the last $F_{n-1}$ value, but do have the advantage of an easy calculation of the $\phi$ coefficients, and easy implementation.

We can go further with the above two-point case, to get 2(3). For this, expressions for $F_i$ are generated from (3.30) by Taylor expansion. Here we use just one:

$$F_1 = F_0 + HD_0^3 + \frac{H^2}{2} D_0^4 + \ldots \tag{3.33}$$

and cutting this off (for the moment) from the fourth derivative onwards, the third derivative is obtained:

$$D_0^3 \approx \frac{1}{H}(F_1 - F_0) \tag{3.34}$$

which, together with (3.30) can be inserted in (3.29) (neglecting the fourth-order derivative term) and upon rearranging, we get

$$D_0^1 \approx G_2 + H\left(-\tfrac{1}{3}F_0 - \tfrac{1}{6}F_1\right) \tag{3.35}$$

which is 2(3), with coefficients $\phi_0 = -\frac{1}{3},\ \phi_1 = -\frac{1}{6}$.

We can go further, taking the three-point approximation. As well as (3.33), we write the Taylor expansion for $F_2$:

$$F_2 = F_0 + 2HD_0^3 + \frac{4H^2}{2} D_0^4 + \ldots\ . \tag{3.36}$$

The simpler formula simply makes use of (3.34) and inserting it into (3.36), $D_0^4$ is obtained and after some rearrangement of the $F$ terms, we get the 3(3) form

$$D_0^1 \approx G_3 + H\left(-\tfrac{1}{3}F_0 + \tfrac{1}{3}F_1\right)\ . \tag{3.37}$$

Here, $\phi_0 = -\frac{1}{3},\ \phi_1 = +\frac{1}{3},\ \phi_2 = 0$ and we thus have the 3(3) form. Again, the last (third) point in $F$ is unused. To involve it as well, write out the Taylor expansions for both $F_1$ and $F_2$ to the fourth derivative term:

$$
\begin{aligned}
F_1 &= F_0 + HD_0^3 + \frac{H^2}{2}D_0^4 + \dots \\
F_2 &= F_0 + 2HD_0^3 + \frac{4H^2}{2}D_0^4 + \dots
\end{aligned}
\tag{3.38}
$$

and solve this little system for both derivative terms. One way to do this is to express $D_0^3$ from the first equation in terms of the other terms and to substitute that in the second. This is a recursive process as described by Bieniasz [11], but what one does is in fact to solve such systems. Doing this, one obtains for this case the 3(4) form

$$
D_0^1 \approx G_3 + H\left(-\tfrac{1}{4}F_0 + \tfrac{1}{6}F_1 + \tfrac{1}{12}F_2\right) \ , \tag{3.39}
$$

with the coefficients obvious from the formula. This treatment can be extended to higher numbers of points and the reader is invited to look up Table 3 in Bieniasz [11], where this has been done up to $n = 5$, up to the 5(6) form. The results of using these are given in that paper for a number of different electrochemical problems and, not unexpectedly, it seems that the $n(n)$ forms are inferior to the $n(n + 1)$ forms, so the latter seem to be the logical choice. There are arguments for the 2(3) form in particular. It is third-order, and this goes well together with higher-order methods, which rarely recommend themselves, in terms of computing time and programming effort, above that order. Also, with unequal intervals, there are no ready-calculated coefficients for more than two points and thus two points recommend themselves. The 2(3) formula can be applied as it stands in (3.35).

There remains one problem, that of the values of $F_i$ needed for the approximations. Their determination depends on the simulation method used, but at this point, it can be said that the major term, $\partial C/\partial T$, always present, can be approximated simply as

$$
\frac{\partial C_i}{\partial T} \approx \frac{C_i -' C_i}{\delta T} \tag{3.40}
$$

where $C_i$ is the present value, just calculated, and $'C_i$ is the last value before the step taken. This is a backward difference, and something better than this can be achieved and is described in Chap. 8. If other terms are contained in $F$, their most recent values are simply used.

In the above discussion we assumed that the *pde* is in the form of (3.26) and that the spatial grid is uniform. For more general *pde*s involving first spatial derivatives as well, Bieniasz [15] developed a three-point, fourth order accurate compact boundary gradient approximation. In the case of nonuniform spatial grids the derivation of compact gradient approximations becomes more complicated (see, for example, Bieniasz [16]) and relevant boundary gradient formulae are not yet available.

## 3.7 Second Derivative Approximations

Clearly, some approximations to second derivatives are also needed. The most widely used approximation is derived as follows. Regard Fig. 3.2 and focus on the point at $x_2$, where the derivative is to apply. We already have Taylor expansions for the points at $x_1$ and $x_3$ [Eqs. (3.7) and (3.10)], both of which have a neglected term of $O(h^4)$. Adding the two equations and rearranging leads to the approximation formula

$$y_2'' = \frac{y_1 - 2y_2 + y_3}{h^2} + O(h^2) \,, \tag{3.41}$$

a second-order approximation. Until recently, this has always been used in digital simulation. In 1990, Kimble and White [17] suggested a higher-order formula using five points (and six at the electrode), together with an unusual way of simulating, described in a later chapter. While the method itself is somewhat demanding in terms of computer memory and has not become popular (it solves the whole grid in $(X,T)$ as one large system), the five-point approximation for the second derivative does seem promising, and has been explored [18, 19]. Therefore, both the central five-point scheme and a few asymmetrical multipoint schemes are needed. The procedure is the same as described above for the first derivative. For example, in the case of the central five-point scheme, centered on the point $(x_3, y_3)$ in Fig. 3.2, we write Taylor expansions for the surrounding four points, going out to terms in $h^4 y_3''''$, and solving the system of four equations. In this case, it is the second row of the matrix inverse that provides the coefficients, since we seek $y_3''$ of the unknowns vector. The result is the matrix equation

$$\begin{bmatrix} -2 & \frac{4}{2!} & \frac{-8}{3!} & \frac{16}{4!} \\ -1 & \frac{1}{2!} & \frac{-1}{3!} & \frac{1}{4!} \\ 1 & \frac{1}{2!} & \frac{1}{3!} & \frac{1}{4!} \\ 2 & \frac{4}{2!} & \frac{8}{3!} & \frac{16}{4!} \end{bmatrix} \begin{bmatrix} h & 0 & 0 & 0 \\ 0 & h^2 & 0 & 0 \\ 0 & 0 & h^3 & 0 \\ 0 & 0 & 0 & h^4 \end{bmatrix} \begin{bmatrix} y_3' \\ y_3'' \\ y_3''' \\ y_3'''' \end{bmatrix} = \begin{bmatrix} y_1 - y_3 \\ y_2 - y_3 \\ y_4 - y_3 \\ y_5 - y_3 \end{bmatrix} . \tag{3.42}$$

Inverting the matrices and multiplying out the second row with the coefficient vector finally yields the approximation, presented in Table A.2 in Appendix A, together with a few others. It turns out that in the process, the terms in $h^5$ drop out and the final approximation is of $O(h^4)$, arising from the neglected terms in $h^6$. The formula has been given as early as 1935 by Collatz [1], who also presented some asymmetric forms in his 1960 book [2], and Bickley in 1941 [3]. Noye [20] also provides a number of multipoint second derivatives for use in the solution of *pde*s.

For reasons that become clear in Chap. 9, we also need an asymmetric form, centered on the point $(x_2, y_2)$, since in a simulation (index 1 being the electrode) this point is also subject to diffusional changes. The obvious course here is to use an asymmetric five-point formula, but this, as pointed out by Collatz [2], is only of

$O(h^3)$. Presumably for this reason, Kimble and White chose a six-point asymmetric scheme here, not provided by Collatz (who goes to a seven-point scheme). The six-point scheme is indeed $O(h^4)$ and is included in Table A.2. At the outer end of the diffusion space, this is needed again at the second-last point and it is given as $y''_5(6)$ in the table, meaning the second derivative applied to point $(x_5, y_5)$. The coefficients are those for $y''_2(6)$ in reverse order but without sign flip.

## 3.8 Derivatives on Unevenly Spaced Points

Some simulation techniques make use of points along $x$, and indeed sometimes along $t$, that are spaced unevenly, either in some smooth transformational progression or more or less arbitrarily. A general treatment is given in this section, as well as an example of a particular algebraic solution.

The need for such formulae and algorithms became clear upon publication of the paper by Rudolph [21], showing that direct discretisation of derivatives on an exponentially expanding grid is in fact better than discretisation on an equally spaced transformed grid. This is in contrast with what the computer science community is agreed upon, based on early works [22–24]. One reason for this is, as Rudolph points out, that the concentration profiles in electrochemical work are normally almost linear near the electrode, so that current approximations or other first derivative expressions in that region in fact operate on a curved function in transformed space. This does not explain why second derivatives in the diffusion space, too, are more accurate when discretised directly on an unequal grid, as they have been found to be by some numerical experiments. There have been a few publications lately presenting derivative approximations on unequal grids in the form of algebraic solutions [21, 25, 26]. Bieniasz, in his introductory paper [27] on adaptive grids, used a four-point formula for a second derivative, which was that of Blom [28]. This was found later to be inconsistent, and Bieniasz presented a corrected expression [29]. Britz and Strutwolf [25] showed a derivation of such formulae and a few particular examples. For approximations on just a few (3 or 4) points, such formulae can be useful but for higher-order forms, a numerical approach is better.

Figure 3.3 shows a few points along some function $u = f(x)$. The symbol $u$ is used for this, to indicate that intervals are unequal. The points are again numbered from 1 to $n$ in general, and a given derivative might be referred to a point at index $i$ among the $n$ points. They will, as above, following the convention for equal intervals, be denoted by the symbols $u'_i(n)$ for first derivatives or $u''_i(n)$ for second derivatives. A one-sided first derivative, for example a current approximation, will then be $u'_1(n)$ and a central second derivative as is often employed, referring to the middle point, is $u''_2(3)$. First derivatives will be given in terms of linear sums of the

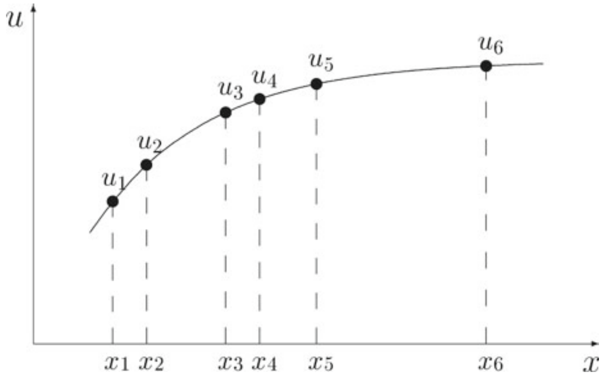


**Fig. 3.3** Arbitrary function with unevenly spaced points

form

$$u' = \sum_{i=1}^{n} \beta_i \, u_i \tag{3.43}$$

and second derivatives by a similar expression,

$$u'' = \sum_{i=1}^{n} \alpha_i \, u_i \ . \tag{3.44}$$

The coefficients are used to compute the derivatives, but can also be useful in the discretisation of derivative boundary conditions or in the setting up of discretisation matrices in some problems.

Define a sequence of displacements $h_k, k = 1 \dots n$, given by

$$h_k = x_k - x_i \tag{3.45}$$

where the reference point $x_i$ is fixed. Clearly, $h_i = 0$. This is the set of displacements from the reference point. Taylor expansions are written for all points to be involved around the $i$th point. Derivatives higher than the second are involved in the expansion, and $D^j u$ denotes the $j$th derivative operator on $u$. For the $k$th point, we have

$$u_k = u_i + h_k D u_i + \frac{h_k^2}{2!} D^2 u_i + \frac{h_k^3}{3!} D^3 u_i + \dots , + \frac{h_k^{n-1}}{(n-1)!} D^{n-1} u_i \quad + O(h_k^n) \ . \tag{3.46}$$

Just $n-1$ derivatives are needed on the right-hand side, and the dominant error term is indicated. The resulting system can be cast in vector/matrix form

$$\mathbf{Hd} = \mathbf{r} + \mathbf{e} \tag{3.47}$$

where

$$\mathbf{H} \equiv \begin{bmatrix} h_1 & h_1^2 & h_1^3 & \dots & h_1^{n-1} \\ h_2 & h_2^2 & h_2^3 & \dots & h_2^{n-1} \\ & & \dots & & \\ h_{i-1} & h_{i-1}^2 & h_{i-1}^3 & \dots & h_{i-1}^{n-1} \\ h_{i+1} & h_{i+1}^2 & h_{i+1}^3 & \dots & h_{i+1}^{n-1} \\ & & \dots & & \\ h_n & h_n^2 & h_n^3 & \dots & h_n^{n-1} \end{bmatrix}, \tag{3.48}$$

**d** denotes the vector of derivatives $[Du_i \quad \frac{D^2u_i}{2!} \quad \dots \quad \frac{D^{n-1}u_i}{(n-1)!}]^T$, **r** stands for the vector of the knowns $(u_k - u_i)$, and **e** for the vector of the last, error terms in (3.46). The vector **d** has been chosen as it was, with the factorials glued to the terms, because this avoids the very small inverse factorials in the matrix. The matrix **H**, as it is, already has elements of greatly varying magnitudes because of the powers of intervals, which can be small; this leads to inaccurate inversion, and the problem might be compounded by the factorials. These must be multiplied appropriately after inversion.

Matrix **H** can be automatically generated and its inverse then yields the solution for the derivatives. We require only the first two derivatives, which arise from the first two rows of the inverse. If the inverse be **V** and its elements at indices $i, j$ be $v_{i,j}$, then we have, by expanding $\mathbf{d} = \mathbf{Vr}$ (neglecting the error terms **Ve** for the moment),

$$u_i'(n) = Du_i = v_{1,1}(u_1 - u_i) + v_{1,2}(u_2 - u_i) + \dots + v_{1,n-1}(u_n - u_i)\ . \tag{3.49}$$

There is no term in $v_{1,i}(u_i - u_i)$ and so there is a break at index $i$ in the sequence of terms. For $k < i$, the terms are $v_{1,k}(u_k - u_i)$, while for $k > i$, the terms are $v_{1,k-1}(u_k - u_i)$. Comparing with (3.43), it is clear that the row of $v_{1,k}$ represents the $\beta_k$ coefficients, bar $\beta_i$. We have

$$\beta_k = \begin{cases} v_{1,k} \ , & k < i \\ v_{1,k-1}, & k > i \end{cases} \tag{3.50}$$

(omitting $\beta_i$). Finally,

$$\beta_i = -\sum_{\substack{k=1 \\ (k \neq i)}}^{n} \beta_k\ . \tag{3.51}$$

Similarly, the $\alpha$ coefficients in (3.44) are obtained from the second row of the inverse, so that

$$\alpha_k = \begin{cases} 2!\ v_{2,k} \quad , & k < i \\ 2!\ v_{2,k-1} , & k > i \end{cases} \tag{3.52}$$

(remembering to multiply by 2!) and

$$\alpha_i = - \sum_{\substack{k=1 \\ (k \neq i)}}^{n} \alpha_k \ . \tag{3.53}$$

The above has been rendered superfluous by the Fornberg algorithm [10], which produces weighting coefficients of great accuracy. It has been implemented by the general subroutines FORN and FORNBERG (see Appendix E), which return both the wanted derivative (first or second or in fact any) and the coefficients that produced it. Section 3.9 also discusses this. The special routine G0FORN specifically computes the gradient at the interface, as mentioned in Sect. 3.5.1.

### 3.8.1 Error Orders

From the above treatment, the error orders of the approximations can be determined. First, a definition of what is meant here is required. With equal intervals of length $h$, orders are expressed as powers of that length. Here we have arbitrarily spaced points, and thus a set of different intervals. In computations to confirm error order expectations, the following scheme can serve. Refer all $h_k$ as displacements from point $i$, as above (3.45). A given derivative can then be computed. Then, all points around the reference point $x_i$ are moved to a given fraction $a$ of their original displacements from the reference point, so that now there is a new set of displacements,

$$h'_k = a h_k \ , \tag{3.54}$$

new values are set at the new set of points and a new derivative is computed. The two derivative estimates then yield the order, as usual.

Another way of expressing this is to take the average of all $h_k$, calling this simply $h$. The Taylor expansions (3.46) then each contain an error term of $O(h^n)$, which becomes the vector **e** in (3.47). When producing the respective derivative by multiplying the first or second row of the inverse matrix with the vector of knowns, an error term will arise by the multiplication of the same row with vector **e**. Some consideration of matrix **H** and its inverse **V** reveals that the top row of **V** (which gives the first derivative) consists of elements, all of which are $O(h^{-1})$ and this, multiplied with the error terms, results in an error of $O(h^{n-1})$. Similarly, the second row, which gives the second derivative, has all terms of $O(h^{-2})$, so that second derivatives are $O(h^{n-2})$ accurate.

Some numerical tests show that for first derivatives using $n$ points, the error is indeed of order $n-1$, while for second derivatives, it is of order $n-2$. If the intervals are equal and the approximations are central (this is possible only for odd $n$), the order goes up by unity for both derivatives.

Theoretically, there is no limit on the number $n$ of points used in the approximations. In practice, however, a limit is set by roundoff in the computations, making an increase in $n$ useless, and the factorials in the matrix $\mathbf{H}$ will increase to impractical levels. In any case, there seems little point in $n$ values greater than about 8, although for the usual 32-bit computers in use today, up to 12-point formulas can be accommodated, and up to 15 if the Fornberg algorithm (Sect. 3.9) is used.

### 3.8.2 A Special Case

The above has considered arbitrarily spaced grids, whereas in practice, the spacing is often the exponentially expanding sequence suggested by Seeber and Stefani [30] and by Feldberg [31]. In terms of points, the special case is a sequence of positions given by an exponentially expanding series of spatial intervals. This will be detailed in Chap. 7. Here, it is sufficient to mention that this special case makes the derivation of the coefficients for various derivative approximations easier and the expressions themselves more compact, as was reported by Martínez-Ortiz [32]. That author also found that there is a particular value for the expansion parameter, $\gamma = \sqrt{2}$, for which the asymmetric four-point second derivative, referred to the second of the four points $u''_2(4)$, is third-order accurate, rather than second-order as for other parameter values or arbitrary placement of points. This might be considered of use in simulation. The four-point approximation has some good properties besides this, as will be explained in Chap. 7. On the other hand, an expansion factor of $\sqrt{2}$ is rather large. It has been found [33] that the expansion factor should not exceed about 1.2 for best results.

### 3.8.3 Current Approximation

The above treatment includes the current approximation on an unequal grid, and the subroutine `G0FORN` in Appendix E can compute it.

It should be noted here that Bieniasz [27] used what amounts to our $\mathcal{G}$, designed for equal intervals, to calculate current approximations on an arbitrary grid. The idea is that the spatial axis is mapped onto an imagined, equally spaced, new axis and the approximation then becomes, using the notation of Sect. 3.5,

$$G = \left( \sum_{i=0}^{n-1} \beta_i u_i \right) \left( \sum_{i=0}^{n-1} \beta_i x_i \right)^{-1} = \frac{\mathcal{G}(\mathbf{u}, n, 1)}{\mathcal{G}(\mathbf{x}, n, 1)} . \tag{3.55}$$

Interesting though this trick is, the results using it are disappointing.

### 3.8.4 An Example

For a small number $n$ of points, it may be worthwhile using the algebraic solutions for the coefficients. The procedure is as described above, but instead of inserting actual $h_k$ values into the matrix in (3.48), that matrix is inverted algebraically and the coefficients expressed as a general formula. These are given, both for first and second derivatives on three and four points in Appendix A. We have derived forms up to $n = 6$, and it soon becomes clear that the disadvantage of these is that every specific $u_i'(n)$ or $u_i''(n)$ requires its own expression set, and for the larger $n$, the resulting subroutines become rather long.

To give an idea of how the tabulated formulae are derived, the derivation for the "central" second derivative on three points, $u_2''(3)$ is shown here. We have the two displacements $h_1$ and $h_3$ [see their definitions in (3.45)], and the equations to solve are

$$\begin{aligned} u_1 - u_2 &= h_1 u_2' + h_1^2 \frac{u_2''}{2!} + O(u_2''') \\ u_3 - u_2 &= h_3 u_2' + h_3^2 \frac{u_2''}{2!} + O(u_2''') \end{aligned} \tag{3.56}$$

(still keeping the factorials glued to the derivatives) so that the matrix $\mathbf{H}$ to be inverted is

$$\mathbf{H} = \begin{bmatrix} h_1 & h_1^2 \\ h_3 & h_3^2 \end{bmatrix} . \tag{3.57}$$

The determinant is

$$\Delta = h_1 h_3 (h_3 - h_1) \tag{3.58}$$

and the inverse $\mathbf{V}$ is

$$\mathbf{V} = \frac{1}{\Delta} \begin{bmatrix} h_3^2 & -h_1^2 \\ -h_3 & h_1 \end{bmatrix} . \tag{3.59}$$

To get the second derivative, the second row of $\mathbf{V}$ is multiplied by the left-hand side vector in (3.56), giving

$$u_2'' = \frac{2!}{\Delta}\left(-h_3(u_1 - u_2) + h_1(u_3 - u_2)\right) \tag{3.60}$$

which when reorganised then yields the coefficients for the approximation $u_2'' \approx \alpha_1 u_1 + \alpha_2 u_2 + \alpha_3 u_3$,

$$\begin{aligned}
\alpha_1 &= \frac{-2}{h_1(h_3 - h_1)} \\
\alpha_2 &= -(\alpha_1 + \alpha_3) = \frac{2}{h_1 h_3} && (3.61) \\
\alpha_3 &= \frac{2}{h_3(h_3 - h_1)}\,. && (3.62)
\end{aligned}$$

In all these cases, the coefficient for the reference point is the negative sum of all the others, as is clear from the form of (3.52) and (3.53). A similar formula has been given by Gavaghan [26] and Rudolph [21], except that their notation for the displacements were such that all distances from the reference point are positive, so that the final formulae differ in the signs of some of the terms.

## 3.9 The Fornberg Algorithm

All of the above approximations can readily be obtained as a set of coefficients computed using the Fornberg algorithm [10], implemented as two subroutines and a function using these, in Appendix E. It applies to any spacing of the points to be involved, whether equal or unequal, and can compute any degree derivative and indeed even the zeroth degree, which is interpolation. Furthermore, it is highly accurate and can handle up to 15-point approximations, which however never are needed.

Fornberg described his algorithm in algorithmic language, and Bieniasz implemented it using the programming language C++ (Bieniasz, private communication), from which we generated the version in Fortran 90. In the present authors' more recent programs, the algorithm is always used in a preliminary calculation of $\beta$ and $\alpha$ coefficients for derivative approximations on a given grid, one set of each for each point on the grid, both central and one-sided forms. Equally, currents can be approximated on unequally spaced points using such precomputed first derivative coefficients, or on the spot, since current computation is a small part of the total cpu use in a given program.

The two Fornberg Fortran 90 subroutines and the function for the current are described in Appendix E.

## References

1. Collatz L (1935) Das Differenzverfahren mit höheren Approximationen für lineare Differentialgleichungen. Schriften Math Sem Inst Ang Math Univ Berlin 3:1–35
2. Collatz L (1960) Numerische Behandlung von Differentialgleichungen. Springer, Heidelberg
3. Bickley WG (1941) Formulae for numerical differentiation. Math Gaz 25:19–27
4. Randles JEB (1948) A cathode-ray polarograph. Part II - the current-voltage curves. Trans Faraday Soc 44:327–338
5. Eyres NR, Hartree DR, Ingham J, Jackson R, Sarjant RJ, Wagstaff JB (1946) The calculation of variable heat flow in solids. Philos Trans R Soc Lond A 240:1–57
6. Heinze J, Störzbach M, Mortensen J (1984) Digital simulation of cyclic voltammetric curves by the implicit Crank-Nicolson scheme. J Electroanal Chem 165:61–70
7. Newman J (1991) Electrochemical systems, 2nd edn. Prentice-Hall, Englewood Cliffs, NJ
8. Britz D (1987) Investigation of the relative merit of some n-point current approximations in digital simulations. Application to an improved algorithm for quasireversible systems. Anal Chim Acta 193:277–285
9. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
10. Fornberg B (1988) Generation of finite difference formulas on arbitrarily spaced grids. Math Comput 51:699–706
11. Bieniasz LK (2003) High order accurate one-sided finite-difference approximations to gradients at the boundaries, for the simulation of electrochemical reaction-diffusion problems in one-dimensional space geometry. Comput Biol Chem 27:315–325
12. Bieniasz LK (2004) Improving the accuracy of the spatial discretisation in finite-difference electrochemical kinetic simulations, by means of the extended Numerov method. J Comput Chem 25:1075–1083
13. Kopal Z (1955) Numerical analysis. Chapman & Hall, London
14. Noumerov BV (1924) A method of extrapolation of perturbations. Mon Not R Astron Soc 84:592–601
15. Bieniasz LK (2007) A fourth-order accurate, three-point compact approximation of the boundary gradient, for electrochemical simulations by the extended Numerov method. Electrochim Acta 52:2203–2209
16. Bieniasz LK (2007) A set of compact finite-difference approximations to first and second derivatives, related to the extended Numerov method of Chawla on nonuniform grids. Computing 81:77–89
17. Kimble MC, White RE (1990) A five-point finite difference method for solving parabolic differential equations. Comput Chem Eng 14:921–924
18. Britz D, Strutwolf J (2000) Higher-order spatial discretisations in electrochemical digital simulation. 1. Combination with the BDF algorithm. Comput Chem 24:673–684
19. Strutwolf J, Britz D (2001) Use of high-order discretisations in digital simulation. 2. Combination with the extrapolation algorithm. Comput Chem 25:511–520
20. Noye J (1984) Finite difference techniques for partial differential equations. In: Noye J (ed) Computational techniques for differential equations. Elsevier, Amsterdam, pp 95–354
21. Rudolph M (2002) Digital simulation on unequally spaced grids. Part 1. Critical remarks on using the point method by discretisation on a transformed grid. J Electroanal Chem 529:97–108
22. Crowder HJ, Dalton C (1971) Errors in the use of nonuniform mesh systems. J Comput Phys 7:32–45
23. Kálnay de Rivas E (1972) On the use of nonuniform grids in finite-difference equations. J Comput Phys 10:202–210
24. Noye J (1982) Finite difference methods for partial differential equations. In: Noye J (ed) Proceedings of the 1981 conference on the numerical solution of partial differential equations, Queen's College, Melbourne, Australia. North Holland, Amsterdam, pp 3–137

25. Britz D, Strutwolf J (2003) Higher-order spatial discretisations in electrochemical digital simulation. Part 4. Discretisation on an arbitrarily spaced grid. Comput Biol Chem 27:327—337
26. Gavaghan DJ (1998) An exponentially expanding mesh ideally suited to the fast and efficient simulation of diffusion processes at microdisc electrodes. 1. Derivation of the mesh. J Electroanal Chem 456:1–12
27. Bieniasz LK (1993) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 1. Introductory exploration of the finite-difference adaptive moving grid solution of the one-dimensional fast homogeneous reaction-diffusion problem with a reaction layer. J Electroanal Chem 360:119–138
28. Blom JG, Sanz-Serna JM, Verwer JG (1988) On simple moving grid methods for one-dimensional evolutionary partial differential equations. J Comput Phys 74:191–213
29. Bieniasz LK (1994) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 2. An improved finite-difference adaptive moving grid technique for fast homogeneous reaction-diffusion problems with reaction layers at the electrodes. J Electroanal Chem 374:1–22
30. Seeber R, Stefani S (1981) Explicit finite difference method in simulating electrode processes. Anal Chem 53:1011–1016
31. Feldberg SW (1981) Optimization of explicit finite-difference simulation of electrochemical phenomena utilizing an exponentially expanded space grid. Refinement of the Joslin-Pletcher algorithm. J Electroanal Chem 127:1–10
32. Martínez-Ortiz F (2005) On the use of a real time application interface under Linux in the electrochemistry laboratory. Application to chronopotentiometry. J Electroanal Chem 574:239–250
33. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376

# Chapter 4
# Ordinary Differential Equations

In this chapter, the numerical solution of ordinary differential equations (*ode*s) will be described. There is a direct connection between this area and that of partial differential equations (*pde*s), as noted in, for example, [1]. The *ode* field is large; but here we restrict ourselves to those techniques that appear again in the *pde* field. Readers wishing greater depth than is presented here can find it in the great number of texts on the subject, such as the classics by Lapidus and Seinfeld [2], Gear [3], Jain [4] or the very detailed volumes by Hairer et al. [5, 6]. There is a very clear chapter in Gerald [7].

We begin with single *ode*s. At the end of this chapter, systems of *ode*s are dealt with; they are in fact one way of handling *pde*s, using the Method of Lines (MOL, see Chap. 9), which has a system of *ode*s as an intermediate stage, or something close to it.

The kind of *ode*s most relevant in the present context is of the form

$$y' = f(y) \tag{4.1}$$

with the boundary condition

$$y(t = 0) = y_0\ . \tag{4.2}$$

There is a more general form in which the time variable $t$ also appears in the brackets on the right-hand side of (4.1), but in the present context, it almost never does. The simplified form will be our model.

In what follows below, the discussion assumes that a point $y(t_n)$ at time $t_n = n\delta t$ is known (as well as previous points), and that we wish to calculate the next point $y(t + \delta t)$ or $y(t_{n+1})$. These will also be denoted by $y_n$, $y_{n+1}$, etc., interchangeably with the other notations.

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_4

## 4.1 An Example *ode*

In what follows, the following specific *ode* will often be used as an example:

$$y' = -y \tag{4.3}$$

with the single boundary condition

$$y(0) = 1\ . \tag{4.4}$$

In numerical texts, the right-hand side of (4.3) often has a multiplier, but this can be normalised out. We note that this is an instance of (4.1) with $f(y) = -y$. The *ode* (4.3) together with boundary condition (4.4) has a known solution,

$$y(t) = \exp(-t) \tag{4.5}$$

and is a very convenient *ode* with which to test methods. Here it will be used to illustrate the implementations.

## 4.2 Local and Global Errors

A note is in order here on errors in the numerical solution of an *ode*. There are (regarding errors in a certain light) two kinds of errors. One is the **local error**, being the error added by a single step. The solution is always carried forward to a final point in $t$, using a number $N$ of steps, and at that point we have a final, or **global error**. Unfortunately, this is always of a lower order than the local error.

## 4.3 What Distinguishes the Methods

For most of the methods used to solve *ode*s (at least, those described here), the way in which the methods differ hinges on how the following three questions are answered:

1. How is $y'$ approximated (how many points are used)?
2. To what value of $t$ is that approximation intended to apply?
3. How is the right-hand side of (4.1) expressed or approximated?

It is very important to be clear on these points in devising simulation strategies, especially (when going on to *pde*s) the boundary conditions, which must conform to these points as well.

It will be seen that for the three methods Euler, BI and the trapezium method, the same approximation expression is used for the left-hand side of (4.1) but because of points made in questions (2) and (3) above, the methods are very different.

## 4.4 Euler Method

Consider Fig. 4.1. The curve is the underlying function $f(y)$ that we are trying to find and we have two pieces of information: one point on the curve, here at $t = 0$, the fat point marked in the figure, and the gradient at any point, for example, at the same point, drawn as a tangent. The procedure is now to find a point $y_1$, at a subsequent chosen time $t$, for example, $\delta t$, as shown. The picture represents the mathematical problem of finding the solution to the *ode* (4.1).

The simplest way to find other points on the curve, or approximations to such points, is to move along the tangent drawn, to $t = \delta t$, as shown in the figure. Clearly, this will not land on the curve, if it is indeed curved as shown, but hopefully somewhere close to it, producing the new point $y_1$. This will then be repeated, because from (4.1) we can obtain a new tangent, using $y_1$ and so on. If we have chosen $\delta t$ not too large (much smaller, for example, than in Fig. 4.1), this will result in a series of discrete points $y_i, i = 1, 2, \ldots, N$, that will be an approximation to the desired solution. The method just described is the **Euler method** and is the basis for what in digital simulation is called the **explicit method**.

Expressing this mathematically, $y'$ is approximated by the simple two-point formula (3.1), written as

$$y'(t) = \frac{y(t + \delta t) - y(t)}{\delta t} \, . \tag{4.6}$$

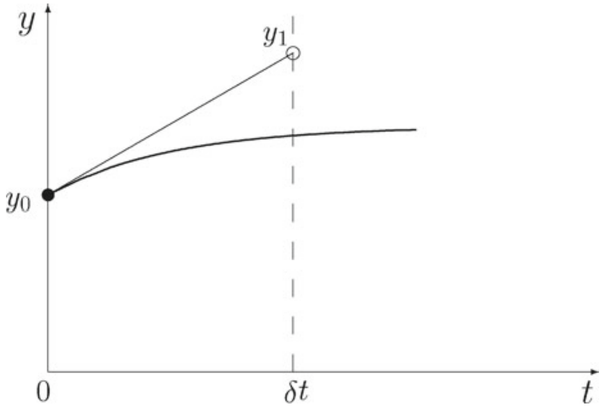


**Fig. 4.1** The Euler method

An alternative way to proceed is to go from the Taylor expansion for $y(t + \delta t)$, as in (3.3),

$$y_{n+1} = y_n + \delta t\, y'_n + O(\delta t^2) \tag{4.7}$$

and substituting for $y'$ from (4.1),

$$y_{n+1} = y_n + \delta t\, f(y_n) + O(\delta t^2) \tag{4.8}$$

which we note is $O(\delta t^2)$. The order refers to the local error due to a single step. In [7] there is clear derivation of the order of the global error from that of the local error, which is not reproduced here. Broadly, the idea is that when taking $N$ steps of length $\delta t$, each contributing a local error of $O(\delta t^2)$, the order is reduced to $O(\delta t)$.

What we have here, with the Euler method, is the definition of the derivative as pertaining to time $t$ (or $n\delta t$) and thus $f(y(t))$ or $f(y_n)$ on the right-hand side. For our specific example (4.3), this becomes approximately

$$y_{n+1} = y_n - \delta t\, y_n \tag{4.9}$$

or

$$y_{n+1} = y_n(1 - \delta t) \; . \tag{4.10}$$

## 4.5 Runge–Kutta (RK)

Figure 4.2 illustrates an improvement on the Euler method. The point marked by × is the same as point $y_1$ in Fig. 4.1, having moved up the tangent from $t = 0$, line 1 in the present figure. This point is just an intermediate result. Using (4.1), we calculate the slope $y'$, that is, $f(y_1)$ and draw that slope, line 2. We can hope that it will be an approximation to the slope at $\delta t$. We now have two slopes, lines 1 and 2. Drawing a third slope midway between these two, (dashed) line 3, might be a better approximation to the slope we should have used, and we use it now, line 4, parallel to line 3, starting from the point $y_0$, and this line hits the $\delta t$ line at the new point $y_1$. This turns out to be a much better approximation to the underlying curve. The reason for this is that the slope of line 4 closely approximates the slope which, if we had known it, would have given us the exact solution, namely that of the line drawn from $y_0$ to the point on the underlying curve at $\delta t$. The above describes the second-order **Runge–Kutta** (**RK**) method (there are other, higher-order variants of Runge–Kutta). This is still an explicit method; the word "explicit" means that at each step, the new point is calculated from previously calculated points only.

For the mathematics of this, consider the discrete equation resulting from the Euler method, as in (4.8). Note that the new point, $y_{n+1}$, is formed from the old point $y_n$ by the addition of a term, here $\delta t f(y_n)$. With RK, these terms are given the

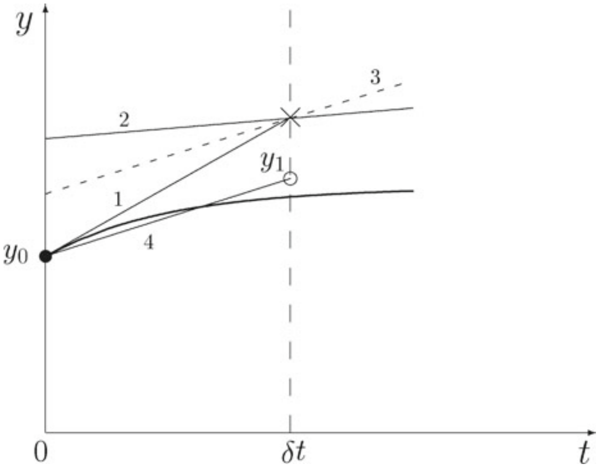


**Fig. 4.2** Second-order Runge–Kutta

symbols $k_i$; there are from one to several of them, and they are added in a weighted manner. The procedure is to generate a number of these $k$'s. One begins with an Euler step,

$$k_1 = \delta t f(y_n)\ . \tag{4.11}$$

(Note that the Euler method can be regarded as a first-order RK form, if we write Eq. (4.8) as

$$y_{n+1} = y_n + k_1 \tag{4.12}$$

which is the same thing). Following the description of Fig. 4.2, the mathematical procedure for second-order RK is then to generate $k_2$ from the tentative point at $y_n + k_1$:

$$k_2 = \delta t f(y_n + k_1) \tag{4.13}$$

and then the final corrected point is

$$y_{n+1} = y_n + \tfrac{1}{2}(k_1 + k_2)\ . \tag{4.14}$$

This formula has a global error of $O(\delta t^2)$ and will here be called **RK2**. We can do even better, generating more $k$'s and getting higher orders. All these RK formulae, including RK2, have variants that have the same error orders. For example, RK2 can also be carried out by generating $k_2$ as

$$k_2 = \delta t f(y_n + \tfrac{1}{2}k_1) \tag{4.15}$$

following with

$$y_{n+1} = y_n + k_2 \ . \tag{4.16}$$

Here, only some of all the variants are given. One variant of third-order RK uses $k_1$ as defined above (4.11), then (4.15) for $k_2$, and finally a third,

$$k_3 = \delta t f(y_n - k_1 + 2k_2) \tag{4.17}$$

giving the third-order scheme **RK3**

$$y_{n+1} = y_n + \tfrac{1}{6}(k_1 + 4k_2 + k_3) \ . \tag{4.18}$$

Numerical professionals, when using the term “Runge–Kutta”, usually mean fourth-order RK, and the classical scheme, here **RK4**, is $k_1$ as above, then $k_2$ as in (4.15), then

$$k_3 = \delta t f(y_n + \tfrac{1}{2}k_2) \ , \tag{4.19}$$

then a fourth $k$,

$$k_4 = \delta t f(y_n + k_3) \tag{4.20}$$

and finally the result, of global fourth-order

$$y_{n+1} = y_n + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \ . \tag{4.21}$$

These formulae can all be applied to *pde*s in a simple manner, easy to program, but have certain drawbacks, as described in a later chapter.

## 4.6 Backwards Implicit (BI)

Another possibility is to let the same derivative approximation pertain to the next time; this is the **backward implicit** (**BI**) method, also called backward Euler [8]:

$$y_{n+1} = y_n + \delta t f(y_{n+1}) \tag{4.22}$$

which again leads to a global error order $O(\delta t)$, and becomes, for our specific example (4.3), after rearranging,

$$y_{n+1} = y_n \frac{1}{1+\delta t} \,. \tag{4.23}$$

This method seems at first sight unpromising, because of its low error order, the same as that for Euler. However, it has some very useful stability properties (see later) and forms the basis for several high-order methods, as will be seen.

## 4.7 Trapezium Method

We know from (3.13) in Chap. 3, how that same derivative approximation is of higher order $O(\delta t^2)$ when applied at the midpoint, and this leads to the **trapezium** method, in which we must find an expression for the right-hand side of (4.1) at time $t + \frac{1}{2}\delta t$. This can be approximated as the average of the values at both ends:

$$f\left(y(t+\tfrac{1}{2}\delta t)\right) \approx \frac{f(y(t)) + f(y(t+\delta t))}{2} \tag{4.24}$$

giving

$$y_{n+1} = y_n + \delta t \frac{f(y_n) + f(y_{n+1})}{2} \,. \tag{4.25}$$

This can be awkward to go on with, being implicit in $y_{n+1}$; in our specific example (4.3), however, there is no problem, the above equation becoming

$$y_{n+1} = y_n - \delta t \, \frac{y_n + y_{n+1}}{2} \tag{4.26}$$

or, after rearranging,

$$y_{n+1} = y_n \frac{1 - \frac{1}{2}\delta t}{1 + \frac{1}{2}\delta t} \tag{4.27}$$

which turns out to be (global) $O(\delta t^2)$. It is the basis of the Crank–Nicolson method when applied to *pde*s, as will be seen.

## 4.8 Backward Differentiation Formula (BDF)

The **BDF** method is ascribed to Curtiss and Hirschfelder [9], who described it in 1952, although Bickley [10] had essentially, albeit briefly, mentioned it already in 1941. Considering Fig. 4.3, the method can be seen as a multi-point extension of BI; the derivative $y'$ is formed by using a number $k$ of points from $y_{n-k+2}$ to $y_{n+1}$, but referred to the new point $y_{n+1}$. This implies a backward derivative, with formulas of the form $y'_n(n)$ as in Appendix A, Table A.1. For example, using the three points shown in Fig. 4.3 (in other words, $k = 3$), the table yields the formula

$$y'_{n+1} \approx \frac{\frac{1}{2}y_{n-1} - 2y_n + \frac{3}{2}y_{n+1}}{\delta t}\,. \tag{4.28}$$

For this value of $k$, then, (4.1) is discretised as

$$y_{n-1} - 4y_n + 3y_{n+1} = 2\delta t f(y_{n+1}) \tag{4.29}$$

(note that the case $k = 2$ is equivalent to BI). This is implicit, as with BI. For our example (4.3), the function on the right-hand side is simply $-y_{n+1}$ and rearrangement then produces

$$y_{n+1} = \frac{-y_{n-1} + 4y_n}{3 + 2\delta t} \tag{4.30}$$

which is of (global) $O(\delta t^2)$. The order can be increased by increasing $k$, the number of time levels (points) used for the backward difference approximation (see Table A.1 for the coefficients $y'_n(n)$).

It turns out that although the BDF schemes achieve higher and higher orders as $k$ increases, the solution begins to oscillate (certainly when the method is adapted

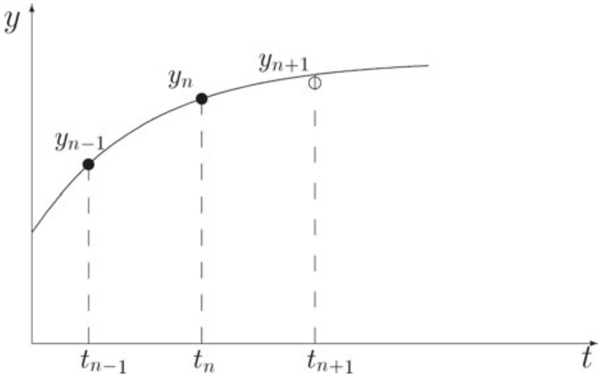


**Fig. 4.3** Backward differentiation formula

to *pde*s) at about $k = 5$ and becomes unstable for $k > 7$ [9, 11, 12]. As applied to diffusion simulations by the Feldberg school [13, 14], a value of 5 is normally used. The choice of this parameter is discussed in a later chapter. The present authors often use the three-point BDF for its $O(\delta t^2)$ property and convenience of use.

Note that the parameter $k$ as defined here, being the number of time points used for the backward difference, which is the convention in electrochemistry since [13], differs from the usage in computer science, where $k$ refers to the number of intervals ("levels") between these points, and is thus smaller by one. It is the electrochemical usage that is adhered to in this book.

### *4.8.1 Starting BDF*

BDF presents the problem of how to start it. If using, for example, a five-point formula, it is not possible to use it for time points earlier than $4\delta t$. At earlier times, an insufficient number of points for the application of the formula are known. This problem is mentioned in, among others, [5, 15, 16]. There are various ways of dealing with this, and four possible strategies will be described here. The simplest way is to ignore the problem, and to artificially assume some points at times $t < 0$, all equal to the initial value given for $t = 0$, and starting directly, generating the point at $\delta t$. This is the **simple start**, which is favoured by Feldberg and coworkers [13], originally without justification other than convenience (private communication, Feldberg 2001). It is illustrated graphically in Fig. 4.4. In the figure, the vertical height of the points indicates the time level, the base line being the level $t = 0$. Filled points are known values while empty points are those to be calculated. Four steps are shown for a sequence using $k = 4$. The sequence starts with the set to the far left, where the three known points are those for $t = 0$. The next set uses two of these and the one just calculated. After this, all required points for subsequent iterations are available. Surprisingly, it turns out that although this yields rather poor results in itself (as will be seen below), a small trick used by Feldberg et al. [13, 14] turns it into a highly accurate method. The trick consists in correcting the time value given to each completed iteration by subtracting from it half a time interval. That is, the iteration numbers $1, 2, \ldots$ which normally are taken as indicating the times $\delta t, 2\delta t, \ldots$ are taken to indicate the values $\frac{1}{2}\delta t$, $\frac{3}{2}\delta t, \ldots$. This will be called the **simple start with correction**, to be described later.

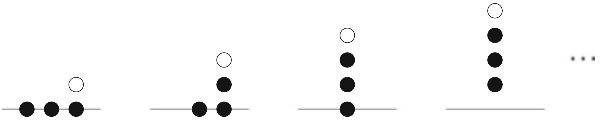


**Fig. 4.4** Simple BDF start schematic

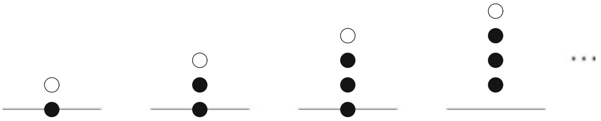


Fig. 4.5 Rational BDF start schematics

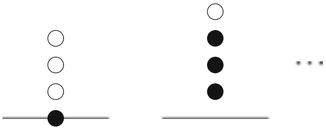


Fig. 4.6 KW BDF start schematics

Among computing professionals solving *ode*s, the usual practice has been what might be called the **rational start**, see Fig. 4.5. This starts with the method BI, which can be regarded as 2-point BDF, to generate the first new point, then uses 3-point BDF to generate the next, then 4-point, and so on, until the desired $k$ has been reached, and continues from there. Inevitably, the first few points will then have errors of a lower order than later points. This does a little better in terms of accuracy than the simple start (without the correction).

There is a high-order start provided by the method of Kimble and White, which will be called the **KW** method here. The method was originally designed [17] for the solution of PDEs, computing a whole grid in space and time in one large system of equations. It is described as applied to *ode*s below, in detail, in Sect. 4.10. In the context of starting a BDF iteration, it can be applied to solve for the first $k - 1$ new values as a system of equations. It is illustrated in Fig. 4.6. The figure shows that (again, $k = 4$) all three unknown points are calculated at once. Briefly, what is done is to apply, at each of the unknown time levels, a four-point approximation to the derivative referred to that level; in the present case (Fig. 4.6), all three are asymmetrical forms, being the three $y'_i(4), i = 2, 3, 4$, of Table A.1 in Appendix A. This gives $k - 1$ (here, 3) equations in as many unknowns. This is a truncated application of the general Kimble and White method described below.

### 4.8.1.1 Time Shifts

We allow ourselves a short digression here, in order to make a special point. There are two ways of presenting an error in a numerical solution of a differential equation. The usual way is to refer to the error in the quantity computed at each new time interval; that is, the difference between the numerical approximation and the underlying exact solution. Another way is to compute, for each calculated value, the time at which that value is exact, and to express the error as a time shift, the difference between the calculated time and the time at that iteration number. It is called a time shift because in many kinds of simulations dealt with in this book, time itself does not enter the equations and, once a simulated sequence of values has become shifted along in time, that shift is permanent. Putting this another way, there is no clock inherent in the method. It will be seen (Chap. 8) that in fact, in linear voltammetry, it does enter the equations indirectly, and no time shifts are encountered in these simulations.

It turns out that there are characteristic time shifts associated with the various methods for solving differential equations [18]. For example, both the Euler method and backwards implicit have a linearly increasing time shift, reaching, respectively, $+\frac{1}{2}\delta t$ and $-\frac{1}{2}\delta t$ at (normalised) $t = 1$, whereas a higher-order method such as the trapezium method has a time shift close to zero. It was found [14, 18] that for BDF with the simple start, the time shift appeared to converge to $-\frac{1}{2}\delta t$. This seemed to empirically justify the Feldberg time correction, but there was no explanation for the effect at the time. Feldberg and Goldstein [14] called it "a fortuitous artefact". The explanation was found [19, 20] and the proof is reproduced in Appendix C. Remarkably, for any function on the right-hand side of a differential equation such as (4.1), whether ordinary or partial (as long as time is not a parameter in the function), and for any BDF value of $k$, there is a convergence to a time shift of exactly $-\frac{1}{2}\delta t$. It will be seen that this makes the Feldberg starting protocol for BDF a very efficient way of applying BDF.

#### 4.8.1.2 Testing the Starting Protocols

To illustrate the points made above, some test computations were run, solving the usual *ode* (4.3). Ten steps were taken over the interval $0 < t \leq 1$, using BDF with $k = 4$. Figure 4.7 shows the results, plotting the error against iteration number. The simple start produces the largest error (curve 1), followed by the rational start

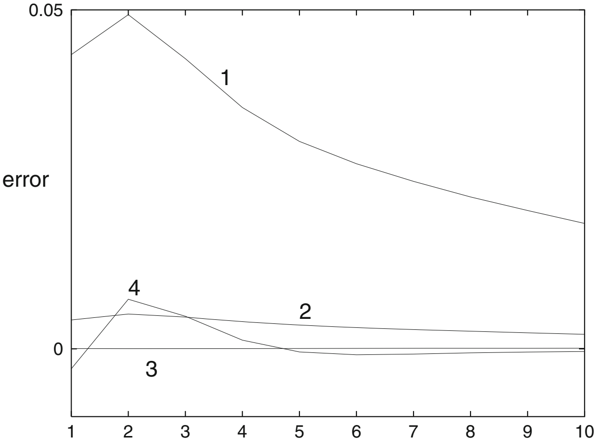


**Fig. 4.7** Errors for the four starting methods (see text)

(curve 2). The KW start (see Sect. 4.10.1 on page 76) is a line that might be mistaken for the zero axis (curve 3). Remarkably, curve 4, showing the simple start with the time correction, converges to the second-best error of the four. Since this is also as easy to program as the simple method, it might be preferable. It was found by some numerical experiments that this method is also the most efficient for electrochemical digital simulations [21]. In that work, the KW method was optimised, and indeed produced highly accurate results. However, this was at considerable computational cost. Efficiency is determined in terms of achieving a target accuracy using the shortest possible computing time and in these terms, the simple start with the correction was the most efficient. This should be received with some relief, as the KW method is not trivial to implement with *pde*s for $k > 3$, especially when optimised for speed, using sparse matrix techniques.

Feldberg and Goldstein [14] extended BDF to exponentially expanding time intervals, by using a general method for computing the coefficients for any time sequence. This also yielded good results.

Finally, Lambert [22] describes a high-order start for general multi-level methods, based on Taylor expansions using higher derivatives. This seems less practical to use as, for example, KW.

## 4.9 Extrapolation

**Extrapolation** is an old technique in numerical analysis invented by Richardson in 1927 [23]. Generally it makes use of known error orders to increase accuracy. In the present context, its application is based on the first-order method BI, mentioned above. One defines a notation in terms of operations $L$ on the variable $y(t)$, the operation being that of taking a step forward in time. Thus, the notation $L_1 y(t)$, or simply $L_1 y_n$, means a single step of one interval (the 1 being indicated by the subscript on $L$). The simplest variant consists of two steps; an application of operation $L_1$ on $y_n$ and as a second step two operations, $L_{1/2}^2$, that is, two consecutive steps of half $\delta t$ (again starting the first from $y_n$), and finally a linear combination of the two results:

$$y_{n+1} = \left(2L_{1/2}^2 - L_1\right) y_n \; . \tag{4.31}$$

The reason why this provides a better estimate of $y_{n+1}$ is that the (global) error $e$ from a series of single steps of size $\delta t$ is a polynomial

$$e = a_1 \delta t + a_2 \delta t^2 + \dots \tag{4.32}$$

with the shown (unknown) coefficients. Clearly, (4.31) will eliminate the first term in that polynomial, leaving only higher terms. The above scheme thus provides an estimate for $y_{n+1}$, that is, $O(\delta t^2)$.

Likewise it is possible to eliminate even higher-order error terms as well, by using more complicated combinations of step sequences. A full description of these is given by Lawson and Morris [24] (second order only) and Gourlay and Morris [25]; these authors adapted the method to the solution of *pde*s and more is said on that in a later chapter. With the higher-order forms, there are again variants, as with Runge–Kutta. Gourlay and Morris carried out some analyses and numerical experiments and the two "winning" schemes are as follows. The third-order scheme is

$$y_{n+1} = \left(\tfrac{9}{2}L_{1/3}^3 - \tfrac{9}{2}L_{2/3}L_{1/3} + L_1\right) y_n \tag{4.33}$$

where the sequence $L_{2/3}L_{1/3}$ means one step of $\frac{2}{3}\delta t$ followed by one more of $\frac{1}{3}\delta t$. The best fourth-order scheme is

$$y_{n+1} = \left(8L_{1/4}^4 + \tfrac{40}{9}L_{3/4}L_{1/4} - \tfrac{32}{3}L_{1/2}L_{1/4}^2 - \tfrac{7}{9}L_1\right) y_n\ . \tag{4.34}$$

In the literature, the schemes are usually described not in terms of fractional steps but with a number of whole-interval steps; the two descriptions are equivalent, however, and it seems that a combination of fractional steps, ending with a new value at the next time interval, is more convenient.

## 4.10 Kimble and White (KW)

The method due to Kimble and White [17] is not actually a method designed for *ode*s, but was devised by the authors for electrochemical *pde*s. The method can however be easily adapted to *ode*s and in fact might be more appropriate there. The method described in 1990 had a precursor in 1987 [26] and this section will start with a description of its expression for *ode*s, because it is simpler and makes the point more clearly. A cut-down application of it has already been outlined in Sect. 4.8.1.

The essence of KW is that multi-point central differences are used as derivatives along most of the $t$ scale, with some asymmetric expressions necessarily added at the ends. Rather than using the time-marching method that is common to all the methods described in previous sections, KW puts all the approximations into one large system of equations, and solves the lot. It turns out that this results in a fortuitous stability [27].

The method is based on another time-marching scheme not mentioned in the above sections: the **leapfrog method** [28, 29], also called the midpoint rule by Hairer and Wanner [6], using central differences. Equation (4.1) can be approximated as

$$\frac{y_{n+1} - y_{n-1}}{2\delta t} = f(y_n) \tag{4.35}$$

where the derivative $y'$ is formed from the central difference spanning two time intervals, and is referred to time $t_n$. This seems intuitively satisfactory and indeed the resulting formula,

$$y_{n+1} = y_{n-1} + 2\delta t f(y_n) \ , \tag{4.36}$$

has a local error of $O(\delta t^3)$ and a global error of $O(\delta t^2)$. The method is conditionally stable [29, 30] or "weakly unstable", according to Hairer and Wanner [6]. Used for parabolic *pde*s, it is unconditionally unstable [29]. Used in the Kimble and White manner, applied at (almost) all points in time, it generates an equation system. Clearly, if we have $N$ points in time for which we want values of $y_i$, starting with the initial value $y_0$, we can write expressions like (4.35) for all $y_i, i = 1 \dots N-1$. There is an expression missing for the point at $N\delta t$, for which there is no central difference. Here, Nguyen and White [26] would use a BDF form and in this case, a three-point form is appropriate, as it has the same error order as the central difference.

The equations then are the following. For $i = 1$ we have, for the general *ode* (4.1),

$$\frac{y_2 - y_0}{2\delta t} = f(y_1) \tag{4.37}$$

which includes the boundary value $y_0$. For points $y_i, i = 2 \dots N - 1$, we have

$$\frac{y_{i+1} - y_{i-1}}{2\delta t} = f(y_i) \tag{4.38}$$

and for the last point, the BDF form

$$\frac{y_{N-2} - 4y_{N-1} + 3y_N}{2\delta t} = f(y_N) \tag{4.39}$$

completing the system, which can now be arranged suitably, writing out the functions $f(y)$. In the present example *ode* (4.3), we have $f(y_i) = -y_i$, and the system becomes

$$\begin{bmatrix} 2\delta t & 1 & & & \\ -1 & 2\delta t & 1 & & \\ & \ddots & & \ddots & \\ & & -1 & 2\delta t & 1 \\ & & 1 & -4 & (3 + 2\delta t) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix} = \begin{bmatrix} y_0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} . \tag{4.40}$$

The interesting thing is that all but the last equation are leap-frog forms, which by themselves result in an unstable solution if using them in a time-marching manner. The mere addition of the last equation (4.39) renders the system stable, and the solution is of $O(\delta t^2)$.

In their 1990 paper, Kimble and White [17] extended this idea to a higher-order formula. They replaced the second-order central difference with a fourth-order central difference using five points (and, as well, for the *pde* they were solving a five-point second derivative with respect to $x$). This necessitates more asymmetric formulae, both for the start and at the end. As with the former scheme, the five-point central difference approximation for $y'$ (see Table A.1, form $y'_3(5)$), if used as a time march, is unstable [27]; and as with that scheme, the BDF form at the top end stabilises it and yields a high-order solution [27]. The formulae, again for a row of points $y_i, i = 1 \ldots N$ (given the boundary or initial value $y_0$), are the following.

For $i = 1$, a 5-point asymmetrical form, called $y'_2(5)$ in the table, is applied to the point $y_1$:

$$\frac{-3y_0 - 10y_1 + 18y_2 - 6y_3 + y_4}{12\delta t} = f(y_1) \tag{4.41}$$

and from there on for all $i = 2, \ldots, N-2$, the central form $y'_3(5)$ is used,

$$\frac{y_{i-2} - 8y_{i-1} + 8y_{i+1} - y_{i+2}}{12\delta t} = f(y_i) \tag{4.42}$$

leaving the asymmetric form $y'_4(5)$ at $i = N-1$,

$$\frac{-y_{N-4} + 6y_{N-3} - 18y_{N-2} + 10y_{N-1} + 3y_N}{12\delta t} = f(y_{N-1}) \tag{4.43}$$

and finally, the BDF form $y'_5(5)$ at the top end,

$$\frac{3y_{N-4} - 16y_{N-3} + 36y_{N-2} - 48y_{N-1} + 25y_N}{12\delta t} = f(y_N)\ . \tag{4.44}$$

Again replacing the function terms $f(.)$ in these equations with our specific example function (4.3), $f(y) = -y$, and rearranging, this becomes the system

$$\begin{bmatrix} (-10+12\delta t) & 18 & -6 & 1 & & & & \\ -8 & 12\delta t & 8 & -1 & & & & \\ 1 & -8 & 12\delta t & 8 & -1 & & & \\ & \ddots & & & & \ddots & & \\ & & & 1 & -8 & 12\delta t & 8 & -1 \\ & & & -1 & 6 & -18 & (10+12\delta t) & 3 \\ & & & 3 & -16 & 36 & -48 & (25+12\delta t) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{N-2} \\ y_{N-1} \\ y_N \end{bmatrix} = \begin{bmatrix} 3y_0 \\ -y_0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{bmatrix} . \tag{4.45}$$

This, as has been mentioned above, was not the way Kimble and White applied the method, but it works rather well and may well be more practical for *ode*s. In their application to *pde*s, the authors reduced the field to only a few points in both time and space, because otherwise the matrix to be solved, even when restricted to the pentadiagonal form, becomes very large.

### 4.10.1 Using KW as a Start for BDF

In the present authors' view, the KW method might find its most useful application in providing high-order starting values for the BDF method, both for *ode*s and *pde*s, where this idea has been dubbed a "hyperimplicit" algorithm by Feldberg and Goldstein [14], presumably because it computes a number of time levels simultaneously. The idea was broadly explained above in Sect. 4.8.1 but can now be detailed. In order to start the BDF steps, the first few points need to be known; using $k$-point BDF, this means, strictly speaking, the initial value $y_0$ plus the points up to $y_{k-2}$, in order to compute the next point $y_{k-1}$. When analysing this, however, it becomes clear that it is better to generate new points up to $y_{k-1}$, in order to get as high an error order as the BDF method itself. For example, to start a 5-point BDF simulation, we compute points $y_1 \dots y_4$. To get them, we can use KW. We restrict the discussion to the 5-point example. It has four unknowns, for which we must write four 5-point equations. It turns out that there are more possibilities than we require, and we can choose which four we use; all but the central one at $y_2$ are asymmetrical, and so we choose a pure BDF form at $y_4$, and choose not to refer to the derivative at $y_0$ (although we could do so). The equations are then, for the indicated $i$'s, using again the 5-point forms in Table A.1 (and moving the divisor $12\delta t$ to the right-hand side)

$$\begin{aligned} -3y_0 - 10y_1 + 18y_2 - 6y_3 + y_4 &= 12\delta t f(y_1) \\ y_0 - 8y_1 + 8y_3 - y_4 &= 12\delta t f(y_2) \\ -y_0 + 6y_1 - 18y_2 + 10y_3 + 3y_4 &= 12\delta t f(y_3) \\ 3y_0 - 16y_1 + 36y_2 - 48y_3 + 25y_4 &= 12\delta t f(y_4) \end{aligned} \tag{4.46}$$

which, applied again to the example *ode* $y' = -y$, produces the system

$$\begin{bmatrix} (-10 + 12\delta t) & 18 & -6 & 1 \\ -8 & 12\delta t & 8 & -1 \\ 6 & -18 & (10 + 12\delta t) & 3 \\ -16 & 36 & -48 & (25 + 12\delta t) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 3y_0 \\ -y_0 \\ y_0 \\ -3y_0 \end{bmatrix}. \tag{4.47}$$

This works rather well with *ode*s, as also seen in Fig. 4.7, where the 4-point BDF form was used. For use with *pde*s, however, it may be considered too much trouble to program, especially as there are easier options, for example, extrapolation, which produce results that are just as good. Also, if BDF is nonetheless chosen, it was found in Sect. 4.8.1 and proved mathematically in Appendix C that the simple start with a simple time correction produces rather good results for much less effort.

**Table 4.1** Errors at $t = 1$ in the simulation of $y' = -y$ using 10 steps of length 0.1, comparing all four BDF starts for BDF $k = 3, 4$ and 5

| Start | k = 3 | k = 4 | k = 5 |
|---|---|---|---|
| Simple | 0.0203(1.01) | 0.0185(1.00) | 0.0186(1.00) |
| Simp+ | 0.0014(2.04) | 0.00039(2.16) | 0.00027(2.00) |
| Rational | 0.0017(2.01) | 0.0021(1.99) | 0.0023(1.99) |
| KW | 0.0013(2.01) | 0.000090(3.00) | −0.0000041(3.94) |

Error orders are given in brackets (measured by taking 100 steps)

Nevertheless, we compared the four starting methods simple, simple with $\delta t/2$ correction ("simp+"), rational and KW for the same simulation, and Table 4.1 shows the errors at $t = 1$ for each case. It is seen that for 3-point BDF, all but the simple start result in much the same error, but for higher BDF forms, the KW start outstrips the others impressively.

The 3-point BDF KW start is actually very simple to implement, requiring only a $2 \times 2$ system whose solution (for $y_1$ and $y_2$) is easily expressed, and so it could be feasible for use in *pde*s. However, the table shows that it results in no better errors than simp+ or the rational start, so it does not recommend itself. It is interesting to note, regarding the error orders, that both simp+ and rational show an order close to 2, regardless of the BDF order, meaning that with these starts, BDF using more than three points is no improvement over three-point BDF. The only start that enables the full accuracy of higher BDF orders is the KW start, which follows the BDF order.

## 4.11 Systems of *ode*s

All the techniques described above can also be applied to the numerical solution of systems of *ode*s, and here we are getting closer to what happens when we solve *pde*s, because in effect, one reduces them to *ode* systems when discretising them.

Instead of a single variable $y$, there are now a number $n$ of variables, $y_1, y_2, \ldots, y_n$, represented by the vector $\mathbf{y}$. Each of these variables has its own differential equation, involving some function, on the function side, of the whole vector:

$$\begin{aligned} y_1' &= f_1(\mathbf{y}) \\ &\vdots \\ y_i' &= f_i(\mathbf{y}) \qquad (4.48) \\ &\vdots \\ y_n' &= f_n(\mathbf{y}) \end{aligned}$$

(4.49)

where each $f_i$ is some linear combination of the elements of the vector $\mathbf{y}$. We are concerned here only with linear systems; in those cases where a *pde* gives rise, upon discretisation, to a nonlinear system of *ode*s, tricks are normally used to avoid them, as will be seen in later chapters. The system is then conveniently written in vector–matrix form,

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}) \tag{4.50}$$

or, since this is linear, as

$$\mathbf{y}' = \mathbf{A}\mathbf{y} \, , \tag{4.51}$$

where $\mathbf{A}$ is the matrix of coefficients in the $n$ functions $f_i$. The system requires a set of boundary conditions (for example, initial values) for its solution, or

$$\mathbf{y}'(0) = \mathbf{y}_0 \, . \tag{4.52}$$

In principle, all the methods described above for single *ode*s can be used for the solution of such a system, when extended suitably. In the case of explicit methods such as Euler or RK, this is very simple to implement, whereas with implicit methods such as BI or the trapezium method, there are some choices to be made.

For brevity, the Euler method will be treated as a special case of RK, considered as RK1. The method is then to start by calculating a vector of $k_1$ values, one for each $y$ element. Discretising directly from (4.51), this is

$$\mathbf{k}_1 = \delta t \mathbf{A}\mathbf{y} \tag{4.53}$$

followed by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \mathbf{k}_1 \tag{4.54}$$

for the Euler method, where $\mathbf{y}_{n+1}$ is the next value of the whole vector $\mathbf{y}$ in the iteration. The extension to RK is obvious. Note that the vector $\mathbf{k_1}$ (and, for higher RK$n$, the other $\mathbf{k_i}$ vectors) can be computed one element after the other, from known elements of $\mathbf{y}$, this being an explicit method.

Of implicit methods, two will be mentioned here, the first being BI. As outlined in Sect. 4.6, this involves equating the same time derivative used in all the methods, with the function on the right-hand side, referred to the next time interval. For the system, then,

$$\frac{\mathbf{y}_{n+1} - \mathbf{y}_n}{\delta t} = \mathbf{A}\mathbf{y}_{n+1} \tag{4.55}$$

or

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \delta t \mathbf{A}\mathbf{y}_{n+1} \tag{4.56}$$

an implicit equation. The solution can be expressed as

$$\mathbf{y}_{n+1} = (\mathbf{I} - \delta t \mathbf{A})^{-1} \mathbf{y}_n \tag{4.57}$$

in which **I** stands for the identity matrix. Similarly, the trapezium method starts with the discretisation

$$\frac{\mathbf{y}_{n+1} - \mathbf{y}_n}{\delta t} = \left(\mathbf{A}\mathbf{y}_{n+1} + \mathbf{A}\mathbf{y}_n\right)/2 \tag{4.58}$$

leading finally to

$$\mathbf{y}_{n+1} = \left(\mathbf{I} - \tfrac{1}{2}\delta t \mathbf{A}\right)^{-1} \left(\mathbf{I} + \tfrac{1}{2}\delta t \mathbf{A}\right) \mathbf{y}_n \; . \tag{4.59}$$

These solutions are rather formal statements, and are rarely used as such, because the matrices involved are almost always either tridiagonal or pentadiagonal, making such direct solutions wasteful. It has been done in some cases [31, 32], without any attempt at optimisation. It is possible to use solution methods that recognise the sparse nature of these systems and many professional program packages are available. One of these will be mentioned below. For methods for *pde*s corresponding to BI, trapezium and BDF, there are more efficient procedures for the solution, to be described in a later chapter.

In some cases, for example, electrochemical *pde*s with derivative boundary conditions, the discretisation process for both the *pde* and the boundary conditions leads to a mix of a differential equation system and one or more plain algebraic equations. They might be, for example, equations of the form

$$\mathbf{f}(\mathbf{y}) = 0 \; . \tag{4.60}$$

The resulting system is called a set of **differential algebraic equations** (**DAEs**) and their solution is now a specialised field with its own texts [6, 33] and there is a package program, **DASSL** [34], for their solution. This can be of use in the present context, for example, with the MOL, which indeed often results in a **DAE** system. This is gone into in some detail in Chap. 9, in the context of Rosenbrock methods.

With most of the implicit methods to be described, however, the solution is found by specialised techniques that make the process efficient, and these will be described in their proper place.

## 4.12 Rosenbrock Methods

This section pertains to systems of *ode*s (or **DAE**s), although Rosenbrock methods are a kind of Runge–Kutta method (Sect. 4.5). In RK, a number of trial changes $k_i$ are explicitly computed, and a weighted sum of them is applied to the variable (or vector). As noted in that section, such explicit RK methods can be highly accurate, but are not stable for all step sizes, a limiting factor when applying them to diffusion problems. A better way is to use implicit Runge–Kutta formulae. With these, assuming, say, $s$ trial $k_i$, the $s$ equations contain expressions in all $k_i$ (some of them perhaps left out, that is, with zero coefficients). This gives rise to a system of $s$ equations in the vector **k**, which can be troublesome. The advantage is that these implicit methods can lead to highly accurate, and stable, responses.

An alternative, called "semi-implicit methods" in such texts as [2], avoids the problems, and some of the variants are L-stable (see Chap. 15 for an explanation of this term), a desirable property. This was devised by Rosenbrock in 1962 [35]. There are two strong points about this set of formulae. One is that the constants in the implicit set of equations for the $k$'s are chosen such that each $k_i$ can be evaluated explicitly by easy rearrangement of each equation. The other is that the method lends itself ideally to nonlinear functions, not requiring iteration (as with the Newton method), because it is, in a sense, already built-in. This is explained below.

Consider the problem of a nonlinear *ode*

$$y' = f(t, y)\ . \tag{4.61}$$

We might wish to solve it using an implicit method, for example, BI (Sect. 4.6). Discretising (4.61) then gives

$$y_{n+1} - y_n = \delta t f(t + \delta t, y_{n+1}) \tag{4.62}$$

and the function on the right-hand side might not be known, nor might we be able to isolate the unknown, $y_{n+1}$, as was possible with the simple *ode*, $y' = -y$. The essence of Rosenbrock is now to take a single Taylor step, expanding $f(t+\delta t, y_{n+1})$ around $f(t, y_n)$ (and given that $y_{n+1} = y_n + k_1$),

$$f(t + \delta t, y_{n+1}) = f(t, y_n) + k_1 f_y(t, y_n) + \delta t f_t(t, y_n) \tag{4.63}$$

where the $f_y$ denotes differentiation by $y$, and $f_t$ that by $t$. We need to know these differentials, but this is always easy. Now (4.62) can be written in the more amenable form,

$$y_{n+1} - y_n = \delta t\big(f(t, y_n) + k_1 f_y(t, y_n) + \delta t f_t(t, y_n)\big)\ . \tag{4.64}$$

The right-hand side is in fact the expression for $k_1$, and contains it; hence, the equation for $k_1$,

$$k_1 = \delta t\big(f(t, y_n) + k_1 f_y(t, y_n) + \delta t f_t(t, y_n)\big) \ , \tag{4.65}$$

is indeed implicit, but note that now $k_1$ can be isolated on the left-hand side, leaving only terms on the right-hand side that can be evaluated. We get

$$(1 - \delta t f_y(t, y_n))k_1 = \delta t f(t, y_n) + \delta t^2 f_t(t, y_n) \ , \tag{4.66}$$

easily solved for $k_1$. The formula might be called a one-stage Rosenbrock variant, and can in fact be used, although it is not highly accurate. Applied to our *ode*, $y' = -y$, for which $f_y = -1$ and $f_t = 0$, it results in the formula

$$y_{n+1} = \frac{y_n}{1 + \delta t} \tag{4.67}$$

which is seen to be identical with BI (in this case). Better than BI, however, (4.65) can be used for nonlinear equations, and also (including $f_t$) to *ode*s where $t$ plays a role, the so-called nonautonomous *ode*s. This is an important point later, when applying Rosenbrock to diffusion problems, where time sometimes enters the equations through boundary conditions (for example, LSV).

Before moving on to real Rosenbrock methods, consider again (4.66). The left-hand side contains a term in $f_y$; if we are dealing with a system of *ode*s, this is called the Jacobian of the system. It is often constant, evaluable in advance. It will be seen in Chap. 9 that unless the diffusion problem has nonlinear concentration terms (for example, from higher-order homogeneous reactions), the Jacobian is constant. If not, it must be evaluated at every step.

There are several Rosenbrock variants, and a profusion of symbols used. Rosenbrock originally described a second-order variant, that is, with errors of $O(\delta t^2)$. It was for an autonomous *ode*, not involving $t$ in the function on the right-hand side of (4.61), and the formula can be readily extended to the nonautonomous case (involving $t$) by a procedure described in [2, 6], among others. Briefly, the procedure consists in adding the time variable to the vector $y$, and taking into account that $t' = 1$, and then expanding the formula appropriately. The formula given by Rosenbrock (and in [2]) then expands to that given in Appendix A, Sect. 9.4. In this book, the profusion of symbols is reduced to one consistent set, as used in Hairer and Wanner [6], and this set is used exclusively in the table of constants in the Appendix. It is also the set adopted by Lang in his publications [36–38]. Lang, as will be noted, is the source of two new variants, at least one of them L-stable, and a modification of an existing one, as well as updated tables of coefficients, correct to 16 decimals in an Appendix in [38]. They are reproduced in the present Appendix A.

In general, a Rosenbrock method consists of a number $s$ of stages. At each stage, a Runge–Kutta-type $k_i$ value is calculated, from explicit rearrangement of implicit equations for these. At stage $i$, the equation is

$$\begin{aligned} k'_i = \; & \delta t\, f\big(t + \alpha_i \delta t, y + \sum_{j=1}^{i-1} \alpha_{ij} k'_j\big) \\ & + \delta t f_y(t,y) \sum_{j=1}^{i} \gamma_{ij} k'_j + \gamma_i \delta t^2 f_t(t,y) \;, \end{aligned} \tag{4.68}$$

the $\alpha$ and $\gamma$ being constants, tabulated in Appendix A. The reason for writing $k'$ is that this equation, applied to *ode* systems, has a small problem, in that the middle of the three terms on the right-hand side contains, in the sum, products of the form $f_y(t,y)\gamma_{ij}k'_j$, which for *ode* systems would mean multiplication of a Jacobian (the equivalent of $f_y(t,y)$) with the vectors $\mathbf{k}'_j$. Hairer and Wanner [6] show that by a transformation of the $k'$ values into new $k$ values,

$$k_i = \sum_{j=1}^{i} \gamma_{ij} k'_j \;, \tag{4.69}$$

a new equation using the $k_i$ can be developed, avoiding this problem. Application of this transformation to (4.68) and rearranging, then lead to a new equation. We now assume that this concerns a system of *ode*s with vectors $\mathbf{y}$ and $\mathbf{k}$. Also, all $\gamma_{ii}$ are conveniently chosen equal and are now simply called $\gamma$, and new constants $\gamma_i$, $a_{ij}$ and $c_{ij}$ appear. The final result is the explicit form,

$$(\mathbf{I} - \delta t \gamma \mathbf{f}_y)\mathbf{k}_i = \gamma\Big(\delta t \mathbf{f}\big(t + \alpha_i \delta t, \mathbf{y} + \sum_{j=1}^{i-1} a_{ij}\mathbf{k}_j\big) + \gamma_i \delta t^2 \mathbf{f}_t + \sum_{j=1}^{i-1} c_{ij}\mathbf{k}_j\Big) \tag{4.70}$$

(we write $\mathbf{f}_y$ and $\mathbf{f}_t$ without their arguments for brevity). Note that in texts such as [6, 38], the equation presented is divided on both sides by $\gamma\delta t$. There are practical reasons for not doing this in the present context. The equation must be applied $s$ times, according to the variant employed. Most variants seek to make the calculation convenient, by allowing some of the constants to be zero. A useful (and L-stable) second-order formula was described by Lang [38], called **ROS2**. A favourite third-order variant is **ROWDA3**, described by Roche [39] and later developed by Lang [37], making it more efficient. This is the variant favoured by Bieniasz, who introduced Rosenbrock methods to electrochemical digital simulation [40, 41] (using different symbols).

Having calculated the $s$ $\mathbf{k}_i$ values (vectors), the solution is

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \sum_{i=1}^{s} m_i \mathbf{k}_i \;, \tag{4.71}$$

where the $m_i$ are weighting factors, included in the tables of constants for each method.

### 4.12.1 Application to a Simple Example ODE

A simple example serves to illustrate the use of Rosenbrock, using the *ode*

$$y' = t + y \; ; \qquad y(0) = 1 \tag{4.72}$$

which has the analytical solution [7]

$$y(t) = 2\exp(t) - t - 1 \; . \tag{4.73}$$

This is of interest because it contains $t$, so that we must use both $f_y$ and $f_t$, both equal to unity. Applying the ROS2 variant to this (see Appendix A for the coefficient values), (4.70) (now for a single variable $y$) translates to the two equations

$$\begin{aligned} k_1 &= \frac{\gamma\left(\delta t f(t,y) + \gamma_1 \delta t^2 f_t\right)}{1 - \gamma \delta t f_y} \\ &= \frac{\gamma\left(\delta t(t + y_n) + \gamma_1 \delta t^2\right)}{1 - \gamma \delta t} \end{aligned} \tag{4.74}$$

and

$$\begin{aligned} k_2 &= \frac{\gamma\left(\delta t f(t + \alpha_2 \delta t, y + a_{21} k_1) + \gamma_2 \delta t^2 + c_{21} k_1\right)}{(1 - \gamma \delta t f_y)} \\ &= \frac{\gamma\left(\delta t(t + \alpha_2 \delta t + y + a_{21} k_1) + \gamma_2 \delta t^2 + c_{21} k_1\right)}{(1 - \gamma \delta t)} \end{aligned} \tag{4.75}$$

containing the (now) known $k_1$. Then applying

$$y_{n+1} = y_n + m_1 k_1 + m_2 k_2 \tag{4.76}$$

yields the solution. This works out rather well, and tests show that errors are $O(\delta t^2)$.

### 4.12.2 Error Estimates

In publications providing Rosenbrock coefficients such as [38, 40], there appear alternative coefficients, “hatted”, such as $\hat{m}_1$, etc. These always provide another variant with an order lower by one than the one used. The purpose of this is that the difference between the two forms provides an (over)estimate of the error. The practice is not followed in this book, as we are generally mainly interested in the error order. So these alternative, lower-order coefficients are not included in Appendix A.

## 4.13 Padé Approximants

The solution for the single variable (4.3) with boundary condition (4.4) is the exponential expression (4.5), and an equivalent solution for the multivariable system of *ode*s (4.50) with its boundary condition (4.52) is [42]

$$\mathbf{y}(t) = \mathrm{e}^{\mathbf{A}t}\mathbf{y}_0 \tag{4.77}$$

where $\mathbf{y}_0$ is the vector of initial values. The exponential function is defined by the power series

$$\mathrm{e}^{\mathbf{A}t} = \mathbf{I} + t\mathbf{A} + \frac{t^2\mathbf{A}^2}{2!} + \dots \tag{4.78}$$

($\mathbf{I}$ being the unit matrix) which leads to several interesting possibilities and throws some light on some of the methods described in this chapter. The problem is how to evaluate the matrix exponential series. One procedure is to use the series as it is, with a sufficient number of terms; for any given simulation problem, for stability reasons, the matrix norm must be less than unity, so powers of $\mathbf{A}$ will approach either zero or a constant value (see Chap. 15), so that the series will converge. This is the basis for the Lawson method [43] and the eigenvalue, eigenvector method, both to be described in Chap. 9. Here the focus is on Padé approximations of the matrix exponential. They belong to a more general class of rational fractions of polynomials for approximating series, described by Frobenius [44], as pointed out by Gragg [45], and extended by Padé [46]. There is a history of these approximations [47]. Smith [42] shows how Padé approximations to the exponential function lead to some well known finite difference schemes.

Rather than expressing the solution of the system of *ode*s at a time $t$ as in (4.77), we can take a single step of length $\delta t$ from a known solution $\mathbf{y}_n$ to the next at $n + 1$,

$$\mathbf{y}_{n+1} = \exp{(\delta t\mathbf{A})}\,\mathbf{y}_n\ . \tag{4.79}$$

Smith [42, p. 117] presents a table of Padé approximants to the exponential function. The simplest of these is the first two terms of the series expansion of the exponential, $1 + \delta t\mathbf{A}$. This produces the approximation

$$\mathbf{y}_{n+1} = (\mathbf{I} + \delta t\mathbf{A})\mathbf{y}_n \tag{4.80}$$

which is seen to be the explicit method. Another approximation is $(\mathbf{I}-\delta t\mathbf{A})^{-1}$, which leads to the form

$$\mathbf{y}_{n+1} = (\mathbf{I} - \delta t\mathbf{A})^{-1}\mathbf{y}_n \tag{4.81}$$

and after multiplication of both sides by $(\mathbf{I} - \delta t\mathbf{A})$ we obtain

$$(\mathbf{I} - \delta t\mathbf{A})\mathbf{y}_{n+1} = \mathbf{y}_n \tag{4.82}$$

which expresses the BI method, backward Euler. The trapezium method (Sect. 4.7) can be easily derived using the (1,1) Padé approximation $(\mathbf{I} + \delta t\mathbf{A}/2)\,(\mathbf{I} - \delta t\mathbf{A}/2)^{-1}$. For an explanation of the $(n,m)$ notation, see Smith [42] or Gragg [45]. There are higher-order Padé approximants to be explored and this has largely not been done, except for a series of works by Malvandi and Ganji [48] and the Rajendran group [49–53], all of whom however used Padé approximants to analytical series solutions.

## References

1. Verwer JG, Sanz-Serna JM (1984) Convergence of method of lines approximations to partial differential equations. Computing 33:297–313
2. Lapidus L, Seinfeld JH (1971) Numerical solution of ordinary differential equations. Academic Press, New York
3. Gear CW (1971) Numerical initial value problems in ordinary differential equations. Prentice-Hall, Englewood Cliffs, NJ
4. Jain MK (1984) Numerical solution of differential equations, 2nd edn. Wiley Eastern, New Delhi
5. Hairer E, Nørsett SP, Wanner G (1987) Solving ordinary differential equations I. Nonstiff problems. Springer, Berlin
6. Hairer E, Wanner G (1991) Solving ordinary differential equations II. Stiff and differential-algebraic problems. Springer, Berlin
7. Gerald CF (1978) Applied numerical analysis, 2nd edn. Addison–Wesley, Reading, MA
8. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical recipes in Fortran. The art of scientific computing, 2nd edn. Cambridge University Press, Cambridge
9. Curtiss CF, Hirschfelder JO (1952) Integration of stiff equations. Proc Natl Acad Sci USA 38:235–243
10. Bickley WG (1941) Formulae for numerical differentiation. Math Gaz 25:19–27
11. Britz D (1997) Stability of the backward differentiation formula (FIRM) applied to electrochemical digital simulation. Comput Chem 21:97–108. See Erratum in ibid. 22:267 (1997)
12. Johannsen K, Britz D (1999) Matrix stability of the backward differentiation formula in electrochemical digital simulation. Comput Chem 23:33–41
13. Mocak J, Feldberg SW (1994) The Richtmyer modification of the fully implicit finite difference algorithm for simulations of electrochemical problems. J Electroanal Chem 378:31–37
14. Feldberg SW, Goldstein CI (1995) Examination of the behavior of the fully implicit finite-difference algorithm with the Richtmyer modification: behavior with an exponentially expanding time grid. J Electroanal Chem 397:1–10
15. Brenan K (1986) Numerical simulation of trajectory prescribed path control problems by the backward differentiation formulas. IEEE Trans Autom Control AC-31:266–269

16. Henrici P (1962) Discrete variable methods in ordinary differential equations. Wiley, New York
17. Kimble MC, White RE (1990) A five-point finite difference method for solving parabolic differential equations. Comput Chem Eng 14:921–924
18. Britz D (1998) Time shift artifacts and start-up protocols with the BDF method in electrochemical digital simulation. Comput Chem 22:237–243
19. Britz D (2001) Consistency proof of Feldberg's simple BDF start in electrochemical digital simulation. J Electroanal Chem 515:1–7
20. Britz TJ, Britz D (2003) Mathematical proof of the consistency of Feldberg's simple BDF start in electrochemical digital simulations. J Electroanal Chem 546:123–125
21. Britz D, Strutwolf J, Thøgersen L (2001) Investigation of some starting protocols for BDF (FIRM) in electrochemical digital simulation. J Electroanal Chem 512:119–123
22. Lambert JD (1972) Computational methods in ordinary differential equations. Wiley, New York
23. Richardson LF (1927) The deferred approach to the limit. Part I. Single lattice. Philos Trans R Soc Lond Ser A 226:299–349
24. Lawson JD, Morris JL (1978) The extrapolation of first order methods for parabolic partial differential equations. I. SIAM J Numer Anal 15:1212–1224
25. Gourlay AR, Morris JL (1980) The extrapolation of first order methods for parabolic partial differential equations. II. SIAM J Numer Anal 17:641–655
26. Nguyen TV, White R (1987) A finite difference procedure for solving coupled, nonlinear elliptic partial differential equations. Comput Chem Eng 11:543–546
27. Britz D (1999) An interesting global stabilisation of a locally short-range unstable high-order scheme for the digital simulation of the diffusion equation. Comput Chem Eng 23:297–300
28. Richardson LF (1911) The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. Philos Trans R Soc Lond Ser A 210:307–357
29. Potter D (1973) Computational physics. Wiley, London
30. O'Brien GG, Hyman MA, Kaplan S (1950) A study of the numerical solution of partial differential equations. J Math Phys 29:223–251
31. Balslev H, Britz D (1992) Direct digital simulation of the steady-state limiting current at a rotating disk electrode for a complex mechanism. Acta Chem Scand 46:949–955
32. Britz D (1996) Brute force digital simulation. J Electroanal Chem 406:15–21
33. Brenan KE, Campbell SL, Petzold LR (1996) Numerical solution of initial-value problems in differential-algebraic equations. SIAM, Philadelphia
34. Petzold L (1983) A description of DASSL - a differential/algebraic system solver. In: Stepleman RS, Carver M, Peskin R, Ames WF, Vichnevetsky R (eds) Scientific Computing, volume 1, IMACS Trans. Sci. Comp., 10th IMACS World congress on systems simulation and scientific computation, Montreal, Canada, August 1982. North Holland, Amsterdam, pp 65–68
35. Rosenbrock H (1962/3) Some general implicit processes for the numerical solution of differential equations. Comput J 5:329–300
36. Lang J (1995) Two-dimensional fully adaptive solutions of reaction-diffusion equations. Appl Numer Math 18:223–240
37. Lang J (1996) High-resolution self-adaptive computations on chemical reaction-diffusion problems with internal boundaries. Chem Eng Sci 51:1055–1070
38. Lang J (2001) Adaptive multilevel solution of nonlinear parabolic PDE systems. Springer, Berlin
39. Roche M (1988) Rosenbrock methods for differential algebraic equations. Numer Math 52:45–63
40. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
41. Bieniasz LK, Britz D (2001) Chronopotentiometry at a microband electrode: simulation study using a Rosenbrock time integration scheme for differential-algebraic equations and a direct sparse solver. J Electroanal Chem 503:141–152

42. Smith GD (1985) Numerical solution of partial differential equations, 3rd edn. Oxford University Press, Oxford
43. Lawson JD (1967) Generalized Runge-Kutta processes for stable systems with large Lipschitz constants. SIAM J Numer Anal 4:372–380
44. Frobenius G (1881) Ueber Relationen zwischen den Näherungsbrüchen von Potenzreihen. J Reine Angew Math 90:1–17
45. Gragg WB (1972) The Padé table and its relation to certain algorithms of numerical analysis. SIAM Rev 14:1–62
46. Padé H (1892) Sur la représentation approchée d'une fonction par des fractions rationelles. Ph.D. thesis, École Nor. (3), Paris. Supplement
47. Brezinski C (1991) History of continued fractions and Padé approximants. Springer, Berlin
48. Malvandi A, Ganji DD (2013) A general mathematical expression of amperometric enzyme kinetics using He's variational iteration method with Padé approximation. J Electroanal Chem 711:32–37
49. Rajendran L, Sangaranarayanan MV (1995) A two-point Padé approximation for the non-steady state chronoamperometric current at ultramicrodisc electrodes. J Electroanal Chem 392:75–78
50. Rajendran L (2000) Padé approximation of ECE and DISP processes at channel electrodes. Electrochem Commun 2:186–189
51. Rajendran L (2000) Padé approximation of EC' processes at channel electrodes. J Electroanal Chem 487:72–74
52. Rajendran L (2006) Two-point Padé approximation of mass transfer rate at microdisk electrodes in a channel flow for all Péclet numbers. Electrochim Acta 51:5407–5411
53. Senthamarai R, Rajendran L (2008) Analytical expression for transient chronoamperometric current at ultramicroband electrode. Russ J Electrochem 44:1156–1161

# Chapter 5
# The Explicit Method

The simplest method of simulating *pde*s, and in particular *ode*s, is the Euler method, in the present context usually called the "explicit method", or **EX** hereafter. It has many drawbacks (to be outlined) but it does have the advantage of simplicity of programming and if you are willing to let your computer do the hard work, it can yield adequate results in many cases. There are recent examples of the use of the method even for rather complex systems [1, 2] or to simulate processes at a disk electrode [3], and a textbook on cyclic voltammetry [4] advocates the method (and provides a program in Pascal). One might thus choose the method as such, or choose to use it as a learning tool. The present authors prefer the latter. Having learned how to use **EX** and becoming aware of its drawbacks, one might be ready to learn something more advanced.

## 5.1 The Discretisation

The discussion will be restricted to the point method and to the one-dimensional case. We will now work in normalised variables, see Sect. 2.3. We then have concentration points $C_0, C_1, \ldots C_N, C_{N+1}$, at the locations $X = 0, H, \ldots NH, (N+1)H$, $H$ being the interval in $X$, see Fig. 5.1. The end points at $X = 0$ and $X = (N+1)H$ are boundary points with concentrations $C_0$ and $C_{N+1}$, respectively. It is the concentrations between these, that are subject to diffusional changes, as follows.

At any point with index $i$, that is at $X = iH$, the diffusion equation (1.1) is discretised on the left-hand side in the Euler manner (Sect. 4.4, or in other words the forward difference formula (3.1)) and on the right-hand side with the central three-point approximation (3.41), giving for the iteration going from time $T$ to the

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8_5

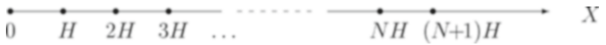


**Fig. 5.1** Discrete sample point sequence

next time $T + \delta T$,

$$\frac{C_i(T + \delta T) - C_i(T)}{\delta T} = \frac{1}{H^2}\left(C_{i-1}(T) - 2C_i(T) + C_{i+1}(T)\right) \ . \tag{5.1}$$

The notation is now simplified by always assuming that we are at time $T$ and are going to time $T + \delta T$ and writing $C(T + \delta T)$ as $C'$. The equation then rearranges to

$$C'_i = C_i + \lambda \left(C_{i-1} - 2C_i + C_{i+1}\right) \tag{5.2}$$

which is explicit for $C'_i$, the new concentration (noting that we have combined $\delta T$ and $H$ into $\lambda = \delta T/H^2$). This equation can be simply applied at all points. At the first and last points to be computed (at $i = 1$ and $N$), the expression on the right contains a boundary term. For $i = 1$ we thus have

$$C'_1 = C_1 + \lambda \left(C_0 - 2C_1 + C_2\right) \tag{5.3}$$

and for $i = N$,

$$C'_N = C_N + \lambda \left(C_{N-1} - 2C_N + C_{N+1}\right) \ . \tag{5.4}$$

The outer point $C_{N+1}$ is normally equal to the bulk initial value, and thus equal to unity, since concentrations have been normalised by the bulk value (in cases involving more than one diffusing species, their respective initial bulk values, normalised by that of the chosen main species). The value of $C_0$ is a little more complicated to set. It depends on the experiment to be simulated, and for simplicity at this point the discussion will be postponed to a later section in this chapter. Assume that we know the value $C_0$. Then we need only go through all concentrations $C_1 \dots C_N$, applying formula (5.2), and obtain the new row of values $C'_1 \dots C'_N$.

## 5.2 Practicalities

In a given computer program, the section in which the above formula (5.2) appears will normally be the shortest part of the program. Other parts of the program will read in the required parameters or print out the current where needed. This raises the question of which parameters to choose. Among these are the length of time assumed for the experiment to be simulated or, in our dimensionless terms, the number of units of the time used for normalisation, $T_{max}$. If the experiment is a step technique, this will mostly be unity. In the case of linear sweep voltammetry (LSV) (where, due to the normalisation, time and potential have the same magnitude), it will be the number of potential units to be swept through (see Sect. 2.3). In either case, we must decide how many intervals ($N_T$) there are to be per time unit or (conversely) the interval length $\delta T$. Other parameters to be decided are the interval $H$ and the number $N$ of points along $X$. This is best done in terms of the largest $X$-value, which in turn is set, via (2.40) and (2.43), such that there are no diffusional changes beyond this point. In view of the properties of the error function and (2.43), a value of $X_{max}$ given by

$$X_{max} = 6\sqrt{T_{max}} \tag{5.5}$$

is recommended. The number 6 appears for the first time in the early work of Feldberg [5], without explanation, and has been tacitly assumed in the electrochemical simulation literature since then. Given the analytical solution of the Cottrell system, Eq. (2.35) on page 16, it is at a distance such that there can be no changes in concentration there greater than $10^{-4}$ of the bulk concentration, which is usually deemed sufficient.

Next, we need the interval $H$ or conversely, the number of points $N$ along the $X$-axis. The recommendation here is to set this indirectly by means of the value of $\lambda$, since this value is known to affect the stability of the simulation critically. For the above formula, the largest usable value for it is 0.5. Having set its value, as well as that of either $N_T$ (preferred) or $\delta T$, $\lambda$ then sets $N$ and $H$.

One also needs to think about at which points in time to output the current, being almost always the aim of the simulation. If this is to be plotted, it might be output at every new time into a plotting file. If one is testing a method, one might only output the current at a selected number of intervals. The present authors find it most convenient to output it at expanding time intervals, as this always gives a compact output list. The current itself can be calculated from the concentration profile by a number of approximations, depending upon how many points one takes. In Chap. 3 (page 45), the function $\mathcal{G}$ was defined to evaluate the gradient $G$. The simplest is the two-point forward difference formula (3.1) which is $\mathcal{G}(C, 2, H)$ or

$$G \approx \frac{C_1 - C_0}{H} \tag{5.6}$$

preferred by many ($G$ here is the dimensionless current or gradient). It is, however, as described in Chap. 3, first-order with respect to $H$ and, with very little effort, one can do much better than that. Since the three-point approximation to the second derivative, the right-hand side of the diffusion equation as shown above in (5.2), is second-order with respect to $H$, there are good arguments [6] for using a second-order formula for $G$, the form $y'_1(3)$ in Table A.1, or

$$G \approx \frac{-3C_0 + 4C_1 - C_2}{2H} \, . \tag{5.7}$$

The present authors have used an even larger number of points, having established a function subroutine `G0FUNC` (now supplemented by `G0FORN`) (see the collection discussed in Appendix E), which allows up to seven points ($n = 7$) (`G0FORN` can handle more than this but even seven might be regarded as more than enough). One can then be sure that there are no significant error terms from the approximation to $G$, no matter which method is used. Furthermore, although the calculation does take a little longer than the simple two-point one, this is never a lot compared with the simulation iteration itself, so it does not matter.

Eyres et al. [7] used a three-point flux approximation for heat flow simulations; the earliest electrochemist to use simulation was Randles [8] and he also used a three-point current. Amatore and Savéant [9] used a six-point approximation, as did Bellamy et al. [10]. The latter authors also inverted the six-point formula to calculate $C_0$, in the manner of `C0FUNC` discussed in Appendix E.

## 5.3 Chronoamperometry and -Potentiometry

The program `COTTEX` (see Appendix E) is an example of a simulation of the Cottrell experiment, using method EX. In this simplest of experiments, the boundary concentration $C_0$ is held at zero from $T = 0$ onwards (see Sect. 2.4.1.1). Boundary conditions will be dealt with in detail in the next chapter, but here it is mentioned only that this condition is the Dirichlet boundary condition, in which, in general, the value of $C_0$ is given. The other extreme condition is the Neumann condition, in which the gradient $G$ is given (a derivative boundary condition). An example of this is given by chronopotentiometry, where a constant current is imposed on the electrode. There is an important point to make here in this regard. When stepping forward in time, one starts with a known array of concentration values $C$ and uses them explicitly (in this case of method EX) to compute the new row, $C'$. Thus, since the old boundary values, $C_0$ and $C_{N+1}$ also go into Eqs. [(5.3) and (5.4)], they must be available. Therefore, in chronopotentiometry, when a given $C$-row has been computed, that value of $C_0$ must be computed which yields the correct gradient $G$, the boundary condition. This can be done readily by inverting the approximation for $G$. For the two-point approximation (5.6) this becomes

$$C_0 = C_1 - GH \tag{5.8}$$

while for the three-point $G$ as in (5.7), it is

$$C_0 = \frac{1}{3}\left(4C_1 - C_2 - 2GH\right) \ . \tag{5.9}$$

In general, for *n* points, it becomes

$$C_0 = \frac{1}{\beta_0}\left(GH - \sum_{i=1}^{n-1} \beta_i C_i\right) . \tag{5.10}$$

This last general equation is implemented in the function `C0FUNC`, described in Appendix E. The formula is applied in the seven-point form in the example program `CHRONOEX` described in Appendix E, simulating chronopotentiometry. It must be applied before every new iteration, in order for the $C_0$ value to be in accord with the other $C$ values. In this program, the current is constant and it is the value of $C_0$ which is displayed and this should go to zero at $T = 1$ (Sect. 2.4.2). A more appropriate display might be the electrode potential, which is always the measured quantity, but this will be dealt with together with the more detailed discussion of boundary conditions in Chap. 6.

A final practical point is the following. When computing the new row $C'$, it must be computed from the old row. Therefore in a program, we cannot replace each $C_i$ with the new value in the array immediately, for each i, because the neighbouring $C_{i-1}$ has just been changed to $C'_{i-1}$. One can either declare two separate arrays, which might be considered slightly wasteful of space, or use the small trick seen in the example programs (Appendix E), where both `COTTEX` and `CHRONOEX` use a trio of scalar points that have the current three values $C_{i-1}$, $C_i$, $C_{i+1}$ that are needed. In this way, the new value $C'_i$ can replace the old one in the array, which is preserved in the scalar point trio. This must of course be shifted along at the bottom of the loop, and the point $C_{i+1}$ picked up at every new loop restart. This device is possibly less important these days, since computers have more memory and an extra array can be easily accommodated.

## 5.4 Homogeneous Chemical Reactions (*hcr*)

One of the simplest examples of a homogeneous chemical reaction (*hcr*) is the Reinert–Berg system [11], in which an electroactive species is generated, for example by means of a light flash, and then reduced as a Cottrell system, while the species decays chemically with a first-order reaction. The reactions are then

$$\begin{aligned} \mathrm{A} + e^- &\rightarrow \mathrm{B} \\ \mathrm{A} &\rightarrow \mathit{prod} \end{aligned} \tag{5.11}$$

with *k* the rate constant of the second reaction, the *hcr*. This gives rise to the single governing equation, for the concentration *C* of substance A

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} - KC \tag{5.12}$$

where everything, including the rate constant ($K = k\tau$) as described in Sect. 2.4, Eq. (2.63), has been normalised. The analytical solution of this equation with Cottrell boundary conditions is given in Chap. 2, Sect. 2.4, Eqs. (2.66) and (2.67). This system is very simple to discretise in the EX manner; the chemical term is added to the equations above, (5.2)–(5.4). For the $i$th concentration [Eq. (5.2)] this becomes

$$C'_i = C_i + \lambda\,(C_{i-1} - 2C_i + C_{i+1}) - K\delta T C_i\ . \tag{5.13}$$

If several species are involved (in this case there is the product *prod*, but we are not interested in it), the equations are extended in an obvious manner, apart from some tricks to be seen in a later chapter in connection with implicit methods. This is one of the attractive aspects of method EX. If the *hcr* is second-order, there will be a term in $C_i^2$ in the discrete equation, and it will present no problem in the discretisation step [12].

There are, however, several problems here. The first is that in writing the governing equation in discrete form as above, we are in effect uncoupling diffusional changes from chemical changes. Numerically, they appear to take place independently of each other, whereas in fact, they interact during the time interval. The last term in (5.13) leads to inaccuracies, and Nielsen et al. [13] proposed using the (explicit) Runge–Kutta method (RK) to overcome this problem, if the chemical changes during a single time interval amount to a few percent of the concentration itself or, in mathematical terms, if the quantity $K\delta T$ in (5.13) exceeds 0.01 or so. In [13] the method adopted was to use RK for the chemical reaction only, following earlier suggestions in the literature [14], whereas in [15] the method is applied to the whole equation, to be described in Chap. 9. This turns out, in either case, to give only a modest improvement in efficiency, which can however be improved a little, see the 5-point method, Chap. 9.

The above manner of computing the two components, that is, diffusional and chemical changes separately, is sometimes called the parallel method. Another way to improve efficiency slightly is to use instead what has been called [13] the sequential method. It was intuitively applied at first, without any real justification other than that it gave better results. The method consists of calculating the diffusional change first, augmenting the concentrations by these amounts, and then to apply the chemical reaction to these augmented values. As pointed out in [13], this gives rather good results, but it was not clear at that time why this should be so. Feldberg in fact used this method and describes it in his seminal chapter [5]. It turns out that, by coincidence (of which we have several in digital simulation), the sequential method does have a mathematical justification and is consistent with the model equations that the discrete expressions approximate. A mathematical proof of this consistency was given in 1991 [16] and is reproduced in Appendix C. The improvement is not great, however, and other methods were sought.

If one is computing the two changes separately (the parallel method), and given that the chemical reaction itself is usually tractable analytically, this component need not be simulated. For a first-order reaction as seen in (5.11) and (5.12), the last part in (5.12) has the general solution for a first-order reaction,

$$C(T) = C(0)\exp(-KT) \tag{5.14}$$

which becomes, over the interval $\delta T$, for the concentration $C_i$

$$C_i(T + \delta T) = C_i(T)\exp(-K\delta T) \tag{5.15}$$

and if $K\delta T$ is not large, this converges to

$$C_i(T + \delta T) = C_i(T)(1 - K\delta T) \tag{5.16}$$

which is indeed the chemical component in (5.13). Feldberg and Auerbach used the analytical method [17] in 1964, as did Flanagan and Marcoux [18] in 1973 and Amatore and Savéant in 1979. It has since then given way to other, better methods, due to the recognition that it is not justified to assume that diffusional and chemical changes are separate. One of these better methods is the explicit Runge–Kutta method (RK) applied to the whole discrete equation set (5.12), and it will be described in Chap. 9.

### 5.4.1 The Reaction Layer

All the above methods, when *hcr*s are present, have one very serious drawback: many *hcr*s give rise to a compact reaction layer, as described in Chap. 2. The above Reinert–Berg reaction does not, but the **EC** reaction,

$$\begin{aligned} \mathrm{A} + e^- &\rightarrow \mathrm{B} \\ \mathrm{B} &\rightarrow \mathrm{C} \end{aligned} \tag{5.17}$$

does. It turns out that the concentration profile for B extends less far into the solution than that of A, which follows the normal rules. Figure 5.2 illustrates this for the above mechanism, where concentration profiles for all three species are shown for the Cottrell experiment run on this system, at (dimensionless) unity time, and a dimensionless rate constant $K = 10$. The profile for species A is unchanged, that is, it is the same as if there were no following reaction, and shows the normal Nernst diffusion layer thickness $\delta$. Species B, however, is confined to a narrower region and its interface concentration is smaller than it would be without the following reaction; species C contains the deficit in B. The problem here is that, if the rate constant is large, the reaction layer is very thin and in order to be able to approximate such a concentration profile, close spacing of points is required, increasing computer time. There are ways to overcome this, but the EX method with equally spaced intervals is not one of them.

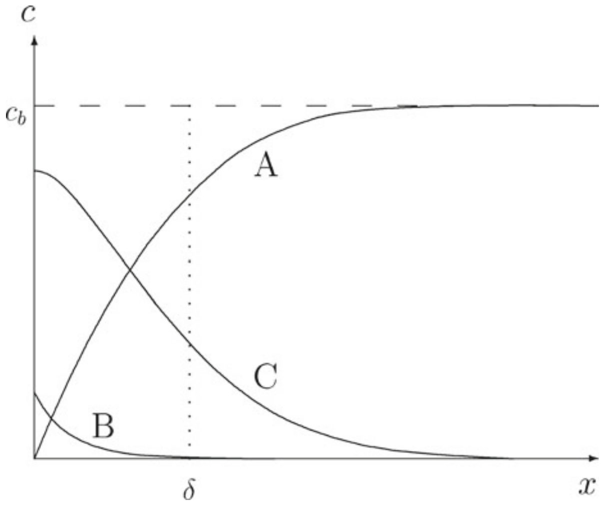


**Fig. 5.2** Concentration profiles with reaction layers

## 5.5 Linear Sweep Voltammetry

One of the main uses of digital simulation—for some workers, the only application—is for LSV or cyclic voltammetry (CV). This is more demanding than simulation of step methods, for which the simulation usually spans one observation time unit, whereas in LSV or CV, the characteristic time $\tau$ used to normalise time with is the time taken to sweep through one dimensionless potential unit (see Sect. 2.4.3) and typically, a sweep traverses around 24 of these units and a cyclic voltammogram twice that many. Thus, the explicit method is not very suitable, requiring rather many steps per unit, but will serve as a simple introduction. Also, the groundwork for the handling of boundary conditions for multispecies simulations is laid here.

The sequence of steps in a CV simulation program is as follows. A simple two-species reaction

$$\mathrm{A} + e^{-} \rightleftharpoons \mathrm{B} \tag{5.18}$$

is our example, assumed to be quasireversible with dimensionless heterogeneous standard rate constant $K_0$:

1. Read in starting potential $p_{start}$ and reversal potential $p_{rev}$ (both in dimensionless potential units), $n_{Tper}$, the number of time intervals per potential unit swept, and $\lambda$, the simulation parameter, and $K_0$, the dimensionless heterogeneous rate constant.

2. Calculate some numbers derived from these inputs, such as $N$, the number of points in space (see below) and $H$, the interval along $X$ and the total number of time steps $n_T$, as well as the time interval $\delta t$. Initialise arrays etc.
3. Open the required output files.
4. Set the potential step $\delta p$ to $-\delta t$, and the current potential $p$ to $p_{start}$; that is, the sweep starts in the negative direction.
5. Enter the loop, each time increasing the potential by $\delta p$, and computing the new concentrations. When these have been calculated, apply the boundary conditions for the present potential, to get the boundary values at $X = 0$; write out the potential and current into the file (perhaps only if there has been a change in current greater than some set value, to reduce the volume of output). If half the total number of time steps $n_T$ has been done, flip the sign of $\delta p$, so that the next half will go in the reverse direction, for CV.
6. During the loop, monitor the current to detect when it passes through a peak (negative sweep) or a trough (return sweep), and keep these values and the potential where they occurred.
7. Finish up by writing out the required numbers such as maximum and minimum currents and potentials.

Some remarks on the above are in order. In the example program `CV_EX` mentioned in Appendix E, a quasiversible reaction is indeed assumed, but if $K_0$ exceeds the value 1000, the boundary conditions are taken to be those for a reversible reaction. How these two different boundary conditions are applied to calculate the concentrations $C_{A,0}$ and $C_{B,0}$ is described below. Note that before new concentrations are to be computed, all old concentrations, including the boundary values, must be known. When a new potential is stepped to, it comes into effect only after the concentrations are renewed, after which $C_0$ is calculated. This might be thought of as less than satisfactory, but it is consistent with the explicit method. In Chaps. 8 and 9, more satisfactory methods will be presented.

Regarding the number $N$ of points in space, the rule shown in Eq. (5.5) is used; the total time $T_{max}$ here is equal to the total number of potential units swept through. If we, for example, set $p_{start}$ equal to 12 and $p_{rev}$ equal to $-12$, then $T_{max} = 48$. One could in principle save a little computing time by recognising the fact that after a given number $n_s$ of steps taken, only that many concentrations can have changed, due to the way changes propagate through the concentration profiles in the explicit method, so while $n_s < N$, one need only recompute $n_s$ points; but this is a small saving in computing time and is not worth the effort, and increases the risk of introducing a program error in the process.

Monitoring for peak and trough currents is seen in the example program `EX_CV`, by the device that a trio of current values $G_1, G_2, G_3$ is always kept ($G_3$ being the most recently computed value), and a check is made whether $G_2$ is maximum or minimium. If it is, the true peak or trough is computed, using the routine `MINMAX` described in Appendix E, which uses a parabolic fit to detect the values, as well as the position. This is converted to potential units in the program. There is a small device to prevent spurious peak/trough detections (which arise in regions where the current is close to zero), by means of the restriction that only those taking place within the potential range $-2 < p < 2$ are accepted.

The number of steps for a complete CV simulation will generally be quite large, especially for the explicit method, where one must set the $\delta p$ values down around at most 0.01, so that if currents are output for later plotting at each iteration, the file becomes unnecessarily large. For this reason, EX_CV checks that there has been a minimum change in current since the last, before writing the current out. Setting this value to 0.001 gave a reasonably reduced number of outputs in some test runs (for example, 1435 out of a total of 48,000, with $\delta p$ set to 0.001).

### 5.5.1 Boundary Condition Handling

In the example program EX_CV, as mentioned, two kinds of boundary conditions are accommodated: those for a quasireversible reaction, and for a fully reversible reaction. The choice is made on the basis of the dimensionless heterogeneous rate constant $K_0$; if it exceeds 1000, the reaction is considered reversible.

For the quasireversible case, the procedure is as follows. At a given stage in the simulation, assume that the two concentration profiles, $C_{A,i}$ and $C_{B,i}$, with $i = 1, 2, \dots, N$, have been calculated and that the potential is $p$. The dimensionless form of the Butler–Volmer equation applies (2.30) and provides the concentration gradient $G_A$, proportional to the current:

$$G_A = K_f C_{A,0} - K_b C_{B,0} \tag{5.19}$$

where the two constants, the forward and backward rate constants, are as given previously (2.31), functions of the potential. The left-hand side of this equation can be discretised as the $n$-point current approximation, leading to

$$\frac{1}{H}\sum_{i=0}^{n-1} \beta_i C_{A,i} = \mathcal{G}(C_A, n, H) = K_f C_{A,0} - K_b C_{B,0} \tag{5.20}$$

which can be dissected as

$$\frac{1}{H}\left(\beta_0 C_{A,0} + \sum_{i=1}^{n-1} \beta_i C_{A,i}\right) = K_f C_{A,0} - K_b C_{B,0}\ . \tag{5.21}$$

There are two unknowns, so we need one more equation. This comes from the fact that the flux of substance A at the electrode must be equal and opposite to that of substance B. If we assume equal diffusion coefficients for the moment, this means

$$G_A + G_B = 0 \tag{5.22}$$

and employing the current approximations for both species and dissecting as above, this and the former equation (5.21) rearranged, gives us the two-unknowns system

$$\begin{bmatrix} (K_f H - \beta_0) & -K_b H \\ \beta_0 & \beta_0 \end{bmatrix} \begin{bmatrix} C_{A,0} \\ C_{B,0} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n-1} \beta_i C_{A,i} \\ -\sum_{i=1}^{n-1} \beta_i C_{A,i} - \sum_{i=1}^{n-1} \beta_i C_{B,i} \end{bmatrix} \tag{5.23}$$

which is readily solved for the two boundary values. The sums can be obtained using the function `G0FUNC` discussed in Appendix E and a small trick. The function requires a number of $C_i$ values, including the (unknown) $C_0$. However, since we are calculating it and do not need the old value, we can afford to set it to zero, so that the function represents the sum for $i = 1, 2, \dots, n-1$, leaving out the zeroth element, as required in the above equations. This is made use of in the example program.

For the reversible case, the Nernst equation applies instead of the Butler–Volmer equation, that is, in dimensionless terms as in (2.32), rewritten as

$$C_{A,0} - e^p C_{B,0} = 0 \tag{5.24}$$

paired again with the flux equality condition (5.22). This gives the system in two unknowns

$$\begin{bmatrix} 1 & -e^p \\ \beta_0 & \beta_0 \end{bmatrix} \begin{bmatrix} C_{A,0} \\ C_{B,0} \end{bmatrix} = \begin{bmatrix} 0 \\ -\sum_{i=1}^{n-1} \beta_i C_{A,i} - \sum_{i=1}^{n-1} \beta_i C_{B,i} \end{bmatrix} \tag{5.25}$$

which is even easier to render as an explicit expression for $C_{A,0}$ and thereby for $C_{B,0}$ from (5.24).

More will be said about boundary conditions in Chap. 6.

## References

1. Jurczakowski R, Orlik M (1999) On the source of oscillatory instabilities in the electroreduction of thiocyanate complexes of nickel (II) at mercury electrodes. J Electroanal Chem 478:118–127
2. Lantelme F, Groult HH, Kumagai N (2000) Study of the concentration-dependent diffusion in lithium batteries. Electrochim Acta 45:3171–3180
3. Myland JC, Oldham KB (2005) Modelling diffusion to a disk electrode by fully explicit simulation. J Electroanal Chem 576:353–362
4. Gosser Jr DK (1993) Cyclic voltammetry. VCH, New York, Weinheim, Germany

5. Feldberg SW (1969) Digital simulation: a general method for solving electrochemical diffusion-kinetic problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 3. Marcel Dekker, New York, pp 199–296
6. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
7. Eyres NR, Hartree DR, Ingham J, Jackson R, Sarjant RJ, Wagstaff JB (1946) The calculation of variable heat flow in solids. Philos Trans R Soc Lond A 240:1–57
8. Randles JEB (1948) A cathode-ray polarograph. Part II - the current-voltage curves. Trans Faraday Soc 44:327–338
9. Amatore C, Savéant RG (1979) ECE and disproportionation. Part VI. General resolution. Application to potential step chronoamperometry. J Electroanal Chem 102:21–40
10. Bellamy AJ, Howat G, MacKirdy I (1978) The use of linear sweep voltammetry to study the addition of electrogenerated $^{-}CH_2CN$ to aromatic carbonyl compounds. J Chem Soc Perkin 11:786–793
11. Reinert KE, Berg H (1962) Theorie der polarographischen Verfolgung schneller chemischer Reaktionen in Lösung mittels reaktionsbedingter Diffusions-Zeit-Kurven. Monatsber Deut Akad Wiss Berlin 4:26–32
12. Britz D, Kastening B (1974) On the electrochemical observation of a second-order decay of radicals generated by flash photolysis or pulse radiolysis. J Electroanal Chem 56:73–90
13. Nielsen MF, Almdal K, Hammerich O, Parker VD (1987) The application of Runge-Kutta integration in digital simulation of electroanalytical experiments. An accurate treatment of the homogeneous kinetics. Acta Chem Scand A 41:423–440
14. Flanagan JB, Marcoux L (1974) Digital simulation of tubular electrode response in stationary and flowing solution. J Phys Chem 78:718–723
15. Britz D (1988) Electrochemical digital simulation by Runge-Kutta integration. J Electroanal Chem 240:17–26
16. Ružić I, Britz D (1991) Consistency proof of the sequential algorithm for the digital simulation of systems involving first-order homogeneous kinetics. Acta Chem Scand 45:1087–1089
17. Feldberg SW, Auerbach C (1964) Model for current reversal chronopotentiometry with second-order kinetic complications. Anal Chem 36:505–509
18. Flanagan JB, Marcoux L (1973) Digital simulation of edge effects at planar disc electrodes. J Phys Chem 77:1051–1055

# Chapter 6
# Boundary Conditions

In this chapter, boundary conditions and how to handle them in simulations are described. Of necessity, some material here overlaps with that in other chapters, especially Chaps. 8 and 9; but this cannot be avoided.

Adsorption kinetics has its own boundary conditions and is treated entirely separately in Chap. 10.

## 6.1 Classification of Boundary Conditions

In the world of numerical analysis, one distinguishes formally between three kinds of boundary conditions [1, 2]: the Dirichlet, Neumann (derivative) and Robin (mixed) conditions; they are also sometimes called [1, 3] the first, second and third kind, respectively. In electrochemistry, we normally have to do with derivative boundary conditions, except in the case of the Cottrell experiment, that is, a jump to a potential where the concentration is forced to zero at the electrode (or, formally, to a constant value different from the initial bulk value). This is Dirichlet only for a single species simulation. If the simulation involves two species (e.g. the reduced and oxidised form) and the surface kinetics obeys the Butler–Volmer equation, flux conditions must apply, i.e. derivatives are involved, see Sect. 5.5.1. If species do not undergo electrode reactions, zero-flux conditions prevail at the location of the electrode surface, involving also derivatives. In what follows below, we briefly treat the single species case, which includes the Cottrell (Dirichlet) condition as well as derivative conditions, and then the two-species case. In a later section in this chapter, a mathematical formalism is described that includes all possible boundary conditions for a single species and can be useful in some more fundamental investigations.

In this chapter, the current approximation function $\mathcal{G}$, defined in Chap. 3, Eq. (3.25), will be used extensively. Note also that since this function is a linear

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8_6

combination of the array argument (for example, $C$ as in $\mathcal{G}(C, n, H)$), the function of a weighted sum of two arrays, such as the arrays $\mathbf{u}$ and $\mathbf{v}$ (to be met later), the following holds ($a$ being some scalar factor):

$$\mathcal{G}(\mathbf{u} + a\mathbf{v}, n, H) = \mathcal{G}(\mathbf{u}, n, H) + a\mathcal{G}(\mathbf{v}, n, H) \ . \tag{6.1}$$

This will prove useful in connection with the "**u-v**" device, see below.

## 6.2 Single Species: The u-v Device

If the simulation only involves a single substance (species), the situation is relatively simple, and this is the starting point. Some of this has already been described in Chap. 5 but will be repeated here, more generally, and with reference to implicit methods, not yet described. Recall the convention that a concentration denoted as $C_i$ is a "present" value at the (spatial) index $i$, that is, a known concentration at time $T$, whereas $C_i'$ denotes a value, yet to be calculated, at time $T + \delta T$.

### *6.2.1 Dirichlet Condition*

Here the value of the boundary concentration is specified. A familiar example in the present context is the outer boundary, beyond the diffusion space, where the concentration usually remains at the initial bulk value during the whole period over which the simulation is carried out. This also applies to the case of the Reinert–Berg mechanism (page 22), in which the bulk concentration itself changes with time, but we know the bulk value at any time, because chemical reaction kinetics, uncomplicated by transport effects, is well understood. In such cases, we can set a given bulk concentration, albeit time-varying. Another familiar example arises from the Cottrell experiment, in which the concentration at the electrode, $C_0$, is set to zero. This is a particular case of that concentration being set to a definite value, not necessarily zero.

### *6.2.2 Derivative Boundary Conditions*

For a single species, there are only two cases of interest, arising from the two kinds of experiments in which either the current is controlled or the potential is controlled, and the reaction is irreversible (if it is not irreversible, two species must be considered). These two cases can serve as a kind of tutorial for the more complex two-species systems.

The chronopotentiometry (controlled current) case has already been described for the **EX** method, where one simply finds a $C_0$ value that fits the known gradient $G$ and the concentration points already established, as shown in (5.10). The situation is not quite so simple for implicit methods, and we introduce here both a preview of these, and the **u-v** device, which will be used extensively.

As will be seen in Chap. 8, implicit methods all lead (for the normal 3-point approximation of the term $\partial^2 C/\partial X^2$) to a system of $N$ equations, each with three unknowns. Generally, this can be written in the form, for the $i$th equation out of the $N$,

$$C'_{i-1} + a_1(i)C'_i + a_2(i)C'_{i+1} = b_i \tag{6.2}$$

where the coefficients $a_1(i)$ and $a_2(i)$ depend on the particular implicit method employed and the point spacing, and the $b_i$ term is some weighted sum of known concentrations, again depending on the method. Because of the fact that the last equation, where $i = N$, includes the bulk value $C'_{N+1}$ which is known, it is possible to reduce the set of equations recursively to one in which each equation has two unknowns, to be outlined in Chap. 8, going backwards from the outer value, and ending in the new equation system:

$$\begin{aligned} C'_0 + a'_1 C'_1 &= b'_1 \\ C'_1 + a'_2 C'_2 &= b'_2 \; . \\ C'_2 + a'_3 C'_3 &= b'_3 \\ &\cdots \end{aligned} \tag{6.3}$$

The details of how to get from the system (6.2) to (6.3) are described in Chap. 8. We can use (6.3) to solve for all $C'_i$, if we know the boundary value $C'_0$. In the case of the Cottrell system, we do know it; it is zero, thus giving us $C'_1$ directly, and then $C'_2$, etc. This is essentially the Thomas algorithm (see Chap. 8).

If there is a derivative boundary condition, things are a little more complicated. There are two kinds of cases. The first of these arises with controlled current, where we know the gradient $G$, as already seen in Chap. 5. Here, however, we cannot simply calculate $C'_0$, because we do not yet know the other concentrations. One way to handle this is to add an expression for the boundary condition to a few equations out of (6.3) and to solve. A simple example is to use the 2-point $G$-approximation in the case, for example, of controlled current ($G$), and the first equation from (6.3)

$$\begin{aligned} C'_1 - C'_0 &= GH \\ C'_0 + a'_1 C'_1 &= b'_1 \end{aligned} \tag{6.4}$$

and to solve for $C'_0$ (and $C'_1$), giving

$$C'_0 = \frac{b'_1 - a'_1 GH}{1 + a'_1} \, . \tag{6.5}$$

This is convenient for the simple 2-point approximation but if $n > 2$, more equations out of (6.3) are needed, and the solution is less straightforward.

Another, more convenient way is the **u-v** device. We establish a relation between the concentrations $C'_i$ and $C'_0$, in order to obtain the extra information needed to solve for $C'_0$. Taking the first equation in (6.3), we rewrite it explicitly for $C'_1$:

$$C'_1 = b'_1/a'_1 - C'_0/a'_1 \tag{6.6}$$

or as a linear function of $C'_0$,

$$C'_1 = u_1 + v_1 C'_0 \tag{6.7}$$

where

$$u_1 = b'_1/a'_1 \, ; \qquad v_1 = -1/a'_1 \, . \tag{6.8}$$

The second equation of the system (6.3) is then reorganised explicitly for $C'_2$, giving

$$C'_2 = b'_2/a'_2 - C'_1/a'_2 \tag{6.9}$$

and we substitute for $C'_1$ from (6.7), getting

$$C'_2 = b'_2/a'_2 - (u_1 + v_1 C'_0)/a'_2 \tag{6.10}$$

which again can be expressed as a linear expression in $C'_0$,

$$C'_2 = u_2 + v_2 C'_0 \tag{6.11}$$

where now

$$u_2 = (b'_1 - u'_1)/a'_2 \, ; \qquad v_2 = -v_1/a'_2 \, . \tag{6.12}$$

This can be continued and we can recursively express all $C'_i$ as linear functions of $C'_0$. For the *i*th concentration,

$$C'_i = u_i + v_i C'_0 \tag{6.13}$$

and the coefficients are given by the recursive expressions

$$u_i = (b'_i - u_{i-1})/a'_i \, ; \qquad v_i = -v_{i-1}/a'_i \, . \tag{6.14}$$

This is easily programmed as a loop process. If we want to avoid a special expression for $u_1$ and $v_1$, there is a trick: we start formally with a tautological equation,

$$C'_0 = u_0 + v_0 C'_0 \tag{6.15}$$

in which, obviously, $u_0 = 0$ and $v_0 = 1$. Then the loop process, applying (6.14) for $i = 1, 2, \ldots$, can be set running. We do not need many iterations—in fact, $n - 1$ are sufficient, $n$ being the number of concentrations used in the approximation for $G$. This is

$$G = \mathcal{G}(\mathbf{C}', n, H) \tag{6.16}$$

and substituting for all $C'_i$ from (6.13), noting (6.1) and defining the vectors (arrays) $\mathbf{u} \equiv [u_0\ u_1\ \ldots u_{n-1}]^T$ and $\mathbf{v} \equiv [v_0\ v_1\ \ldots v_{n-1}]^T$, we get

$$G = \mathcal{G}(\mathbf{u}, n, H) + C'_0\, \mathcal{G}(\mathbf{v}, n, H) \tag{6.17}$$

or multiplying by $H$,

$$GH = \mathcal{G}(\mathbf{u}, n, 1) + C'_0\, \mathcal{G}(\mathbf{v}, n, 1) \tag{6.18}$$

which can now be rearranged to the explicit equation for $C'_0$,

$$C'_0 = \frac{GH - \mathcal{G}(\mathbf{u}, n, 1)}{\mathcal{G}(\mathbf{v}, n, 1)} \tag{6.19}$$

yielding $C'_0$. It is now also seen that the little trick (6.15) has another advantage, enabling the use of the sums for $i = 0, 1, \ldots, n-1$, which our function $\mathcal{G}$ and thus procedure `G0FUNC` evaluates. `G0FUNC(u, n, 1.0_dbl)` must be called as shown and `u` and `v` are arrays with bounds `(0:n-1)`. For unequal point spacing, the routine `G0FORN` is called instead.

Formally, the above process is equivalent to (6.4), extended for any $n$ and solving that system. The **u-v** device is a more efficient way of solving it than any linear equation solver that might otherwise have been used, as $n$ becomes larger. The **u-v** device will be extensively used in this book, even with implicit methods for coupled equation systems, where we must solve for a number of concentration profiles (see below). There are practitioners who believe that $n = 2$, that is the two-point $G$-approximation, is good enough. This is justified in cases where $H$ is very small, as it often is, at least near the electrode, when unequal intervals are used (see Chap. 9). In that case, one can simply use (6.5).

Another case, if we have just one species, besides controlled current, is the irreversible, controlled potential case. The gradient $G$ is then given by half of the Butler–Volmer equation and for the as yet unknown concentrations and the potential $p'$ at the new time, this is

$$G' = K_f C'_0 \tag{6.20}$$

(where we write $G'$, since the gradient is now time-dependent and we are referring to the new time level) with $K_f$ given by

$$K_f = K_0 \exp(-\alpha p') \ . \tag{6.21}$$

$G'$ itself must then be replaced by the right-hand side of (6.16) and the resulting equation,

$$\mathcal{G}(C', n, H) = K_f C'_0 \tag{6.22}$$

decomposed by applying (6.13) to the left-hand side. After some rearrangement, this yields the solution,

$$C'_0 = \frac{\mathcal{G}(\mathbf{u}, n, 1)}{HK_f - \mathcal{G}(\mathbf{v}, n, 1)} \ . \tag{6.23}$$

Again, if a two-point approximation is used, this simplifies to

$$C'_0 = \frac{b'_1}{1 + a'_1(1 + HK_f)} \ . \tag{6.24}$$

Lastly, in Chap. 9, Sect. 9.2.7, an improvement in the above is described, based on Hermitian schemes for a better gradient approximation.

## 6.3 Two Species

In Chap. 5, the two-species cases were described for the explicit method. Here we add those for the implicit case. Both Dirichlet and derivative boundary conditions are of interest, the latter both with controlled current or quasireversible and systems under controlled potential.

When two species are involved, they may have different diffusion coefficients. Here it will be assumed that the two species might be two out of more than two species in a given mechanism, and that normalisation is referred to some species other than these two. Therefore both their diffusion coefficients need to be normalised. Let the two species be called O and R, and the reference species be called A. Then the normalisations are

$$d_O = D_O/D_A, \qquad d_R = D_R/D_A \tag{6.25}$$

and, of course, the concentrations are normalised as usual by the initial bulk concentration of the reference species A.

Often, for convenience, diffusion coefficients of all species in a mechanism are assumed equal, however unrealistic this probably is. If the reader wants to assume this, the *d*'s in what follows can simply be set equal to unity. The paper [4] underlines the point, finding significant effects of unequal diffusion coefficients, and Pedersen et al. [5] in fact measured differences of about 20 % between the diffusion coefficients of some organic compounds and the radical anions formed by their reduction. Such differences may well be significant in a simulation.

We consider here two species connected by the reduction reaction

$$\mathrm{O} + e^- \rightarrow \mathrm{R}\,. \tag{6.26}$$

We begin with the simpler case of the two species not being coupled, that is, each of their discrete equations contains only terms from one of the species. The coupled case is given below, being rather more complicated. There is of course coupling of the two species by the boundary conditions.

As with the single-species case above, we anticipate the treatment given in Chap. 8 for implicit methods. At each point *i* in space, there is an equation like (6.2) for each species:

$$\begin{aligned} C'_{O,i-1} + a_{O,1}(i)C'_{O,i} + a_{O,2}(i)C'_{O,i+1} &= b_{O,i} \\ C'_{R,i-1} + a_{R,1}(i)C'_{R,i} + a_{R,2}(i)C'_{R,i+1} &= b_{R,i} \end{aligned} \tag{6.27}$$

where the coefficients may be different (often some of them at least are common to the two). The two equations are of the same form as (6.2), except that there are now twice as many. Again, the bulk concentrations will be known and can be used to reduce the whole set to a new set in which each equation has only two unknowns, and we write out the first few of these:

$$\begin{aligned} C'_{O,0} + a'_{O,1}C'_{O,1} &= b'_{O,1} \\ C'_{R,0} + a'_{R,1}C'_{R,1} &= b'_{R,1} \\ C'_{O,1} + a'_{O,2}C'_{O,2} &= b'_{O,2} \\ C'_{R,1} + a'_{R,2}C'_{R,2} &= b'_{R,2} \\ C'_{O,2} + a'_{O,3}C'_{O,3} &= b'_{O,3} \\ C'_{R,2} + a'_{R,3}C'_{R,3} &= b'_{R,3} \\ &\ldots \end{aligned} \tag{6.28}$$

It might have been clearer to write these two systems separately but it was decided to mix them in the above manner, as this serves a certain purpose later, with coupled systems.

The **u-v** device can now be applied as before, the only complication being that there will be two sets of **u**'s and **v**'s; the treatment is identical to the above one and results in the two equations

$$C'_{Z,i} = u_{Z,i} + v_{Z,i} C'_{Z,0} \tag{6.29}$$

in which $Z$ can be either O or R. The equations that generate the coefficients are the same as (6.14).

It is now possible to bring in the particular boundary conditions, starting with the Cottrell case, which is the simplest. For all mechanisms and boundary conditions, we require two equations involving the two unknown boundary concentrations. As with all cases, one of these is the flux condition,

$$f_O + f_R = 0 \tag{6.30}$$

mentioned in Chap. 5, Sect. 5.5.1. We must take the possibly different diffusion coefficients into account, since it is the fluxes, not the concentration gradients, that must be equal and opposite:

$$D_O G_O + D_R G_R = 0 \tag{6.31}$$

or, applying the normalisations of the diffusion coefficients,

$$d_O G_O + d_R G_R = 0\ . \tag{6.32}$$

This then becomes, for the usual $n$-point $G$-approximation,

$$d_O \mathcal{G}(\mathbf{C}'_O, n, H) + d_R \mathcal{G}(\mathbf{C}'_R, n, H) = 0 \tag{6.33}$$

(where $\mathbf{C}'_O$ stands for the vector of the values $\mathbf{C}'_{O,0} \ldots$ and similarly for $\mathbf{C}'_R$) and substituting (6.14) for $C'_{O,i}$ and $C'_{R,i}$, and rearranging, this gives the equation

$$d_O \mathcal{G}(\mathbf{v}_O, n, 1) C'_{O,0} + d_R \mathcal{G}(\mathbf{v}_R, n, 1) C'_{R,0} = -d_O \mathcal{G}(\mathbf{u}_O, n, 1) - d_R \mathcal{G}(\mathbf{u}_R, n, 1) \tag{6.34}$$

where the $H$ has been divided out, but this is a matter of preference. The equation will be written in a briefer form,

$$d_O \mathcal{G}(\mathbf{v}_O) C'_{O,0} + d_R \mathcal{G}(\mathbf{v}_R) C'_{R,0} = -d_O \mathcal{G}(\mathbf{u}_O) - d_R \mathcal{G}(\mathbf{u}_R) \tag{6.35}$$

where the missing arguments (and the substitution of $H$ with unity) are assumed. The other equation depends on the boundary conditions and these will now be gone through. They are controlled current, controlled potential with quasireversible and reversible reactions.

With controlled current, the value of $d_O\, G_O$ is controlled (*not* $G_O$ itself!). Let the (dimensionless) value of this current be $G$. This yields the second equation simply as

$$G = d_O \mathcal{G}(\mathbf{C}'_O, n, H) = d_O \left( \mathcal{G}(\mathbf{u}_O, n, H) + C'_{O,0} d_O \mathcal{G}(\mathbf{v}_O, n, H) \right) \tag{6.36}$$

or, multiplying both sides by $H$ and rearranging (using the briefer form)

$$d_O C'_{O,0} \mathcal{G}(\mathbf{v}_O) = GH - d_O \mathcal{G}(\mathbf{u}_O) \tag{6.37}$$

which can be used directly to obtain $C'_O$ and thus, from (6.35), $C'_R$. Nevertheless for consistency with the other cases to follow, the equation system is given here:

$$\begin{bmatrix} d_O\mathcal{G}(\mathbf{v}_O) & d_R\mathcal{G}(\mathbf{v}_R) \\ d_O\mathcal{G}(\mathbf{v}_O) & 0 \end{bmatrix} \begin{bmatrix} C'_{O,0} \\ C'_{R,0} \end{bmatrix} = \begin{bmatrix} -d_O\mathcal{G}(\mathbf{u}_O) - d_O\mathcal{G}(\mathbf{u}_R) \\ GH - d_O\mathcal{G}(\mathbf{u}_O) \end{bmatrix} . \tag{6.38}$$

Controlled potential can be either a potential step or some potential program such as LSV/CV, staircase voltammetry or even ac voltammetry, see the standard texts [6, 7] for details. In all of these, we have (dimensionless) potential $p$ at time $T$ and the new potential $p'$ at the next time level; $p'$ might thus be a constant (as in potential step) or varying with time. One can then distinguish between the cases quasireversible (including irreversible) systems or fully reversible ones. Some simulation packages such as DigiSim [8] do not include the reversible case, arguing that it does not exist, and is in fact a quasireversible reaction with a large heterogeneous rate constant. This makes some sense but on the other hand, setting that rate at some arbitrarily very high value to ensure reversible behaviour is no more justifiable than assuming Nernstian equilibrium, that is, reversibility.

A quasireversible system is characterised by the Butler–Volmer equation, here in dimensionless form,

$$d_O G' = K_f C'_{O,0} - K_b C'_{R,0} \tag{6.39}$$

with

$$K_f = K_0 \exp(-\alpha p'), \qquad K_b = K_0 \exp([1-\alpha]p') . \tag{6.40}$$

(Note that in (6.39), $d_O G'$ is once again used, taking into account the diffusion coefficient of species O being different from that of the reference species). Equation (6.39) is expanded for $G'$ as before, the **u-v** substitutions made and rearranged, to give

$$\left(K_f - d_O\mathcal{G}(\mathbf{v}_O, n, H)\right) C'_{O,0} - K_b C'_{R,0} = d_O\mathcal{G}(\mathbf{u}_O, n, H) \tag{6.41}$$

which, multiplying by $H$ and adopting the previous shorthand for $\mathcal{G}$, and adding the flux condition (6.35), results in the system

$$\begin{bmatrix} -d_O\mathcal{G}(\mathbf{v}_O) & -d_R\mathcal{G}(\mathbf{v}_R) \\ HK_f - d_O\mathcal{G}(\mathbf{v}_O) & -HK_b \end{bmatrix} \begin{bmatrix} C'_{O,0} \\ C'_{R,0} \end{bmatrix} = \begin{bmatrix} d_O\left(\mathcal{G}(\mathbf{u}_O) + \mathcal{G}(\mathbf{u}_R)\right) \\ d_O\mathcal{G}(\mathbf{u}_O) \end{bmatrix} . \tag{6.42}$$

This can easily be made into the irreversible case by setting $K_b = 0$ and, in principle, into the reversible case by setting $K_0$ very large. However, another way to ensure reversibility is to specify it as such, by the Nernst equation as in Chap. 2, page 15,

$$C'_{O,0}/C'_{R,0} = \exp(p') \tag{6.43}$$

or

$$C'_{O,0} - \exp(p')C'_{R,0} = 0 \tag{6.44}$$

which, added to the flux condition produces the system

$$\begin{bmatrix} d_O\mathcal{G}(\mathbf{v}_O) & d_R\mathcal{G}(\mathbf{v}_R) \\ 1 & -\exp(p') \end{bmatrix} \begin{bmatrix} C'_{O,0} \\ C'_{R,0} \end{bmatrix} = \begin{bmatrix} -d_O\left(\mathcal{G}(\mathbf{u}_O) + \mathcal{G}(\mathbf{u}_R)\right) \\ 0 \end{bmatrix}. \tag{6.45}$$

### 6.3.1 Two-Point Derivative Cases

For those who prefer to keep the derivative approximation of $G$ down to the two-point form, the above can perhaps be simplified a little; the **u-v** device is not needed as such, as only the first substitution (6.6) is required.

The flux condition (6.32) is represented in two-point form as

$$\frac{d_O}{H}\left(C'_{O,1} - C'_{O,0}\right) + \frac{d_R}{H}\left(C'_{R,1} - C'_{R,0}\right) = 0 \tag{6.46}$$

and (6.6) applied to the first two equations of (6.28), the $C'$ at $X = H$ can be eliminated, to give, after some cleaning up, the flux condition equation

$$d_O\left(1 + \frac{1}{a'_{O,1}}\right)C'_{O,0} + d_R\left(1 + \frac{1}{a'_{R,1}}\right)C'_{R,0} = d_O\frac{b'_{O,1}}{a'_{O,1}} + d_R\frac{b'_{R,1}}{a'_{R,1}} \tag{6.47}$$

which is the one always needed out of the two. For controlled current, (6.36) becomes (again invoking (6.6))

$$GH = -\left(1 + \frac{1}{a'_{O,1}}\right)C'_{O,0} + \frac{b'_{O,1}}{a'_{O,1}} \tag{6.48}$$

or

$$\left(1 + \frac{1}{a'_{O,1}}\right)C'_{O,0} = -GH + \frac{b'_{O,1}}{a'_{O,1}} \tag{6.49}$$

and thus, together with the first equation (6.47), this results in the system

$$\begin{bmatrix} d_O\left(1+\frac{1}{a'_{O,1}}\right) & d_R\left(1+\frac{1}{a'_{R,1}}\right) \\ \left(1+\frac{1}{a'_{O,1}}\right) & 0 \end{bmatrix} \begin{bmatrix} C'_{O,0} \\ C'_{R,0} \end{bmatrix} = \begin{bmatrix} d_O\frac{b'_{O,1}}{a'_{O,1}} + d_R\frac{b'_{R,1}}{a'_{R,1}} \\ -GH + \frac{b'_{O,1}}{a'_{O,1}} \end{bmatrix} . \tag{6.50}$$

It is arguable whether this is in fact simpler than the form for general $n$, Eq. (6.38), but it does avoid calling a function.

The quasireversible case, analogous to Eqs. (6.39)–(6.41), becomes

$$\left[HK_f + d_O\left(1+\frac{1}{a'_{O,1}}\right)\right] C'_{O,0} - HK_b C'_{R,0} = d_O\frac{b'_{O,1}}{a'_{O,1}} \tag{6.51}$$

and the system to be solved,

$$\begin{bmatrix} d_O\left(1+\frac{1}{a'_{O,1}}\right) & d_R\left(1+\frac{1}{a'_{R,1}}\right) \\ HK_f + d_O\left(1+\frac{1}{a'_{O,1}}\right) & -HK_b \end{bmatrix} \begin{bmatrix} C'_{O,0} \\ C'_{R,0} \end{bmatrix} = \begin{bmatrix} d_O\frac{b'_{O,1}}{a'_{O,1}} + d_R\frac{b'_{R,1}}{a'_{R,1}} \\ d_O\frac{b'_{O,1}}{a'_{O,1}} \end{bmatrix} . \tag{6.52}$$

Again, the irreversible case is accommodated by setting $K_b$ to zero.

The reversible case gives rise to the same equation as for higher $n$ as in (6.44) and thus to the system

$$\begin{bmatrix} d_O\left(1+\frac{1}{a'_{O,1}}\right) & d_R\left(1+\frac{1}{a'_{R,1}}\right) \\ 1 & -\exp(p') \end{bmatrix} \begin{bmatrix} C'_{O,0} \\ C'_{R,0} \end{bmatrix} = \begin{bmatrix} d_O\frac{b'_{O,1}}{a'_{O,1}} + d_R\frac{b'_{R,1}}{a'_{R,1}} \\ 0 \end{bmatrix} . \tag{6.53}$$

## 6.4 Two Species with Coupled Reactions: U-V

Up to this point, the treatments have involved reactions for which the discrete form of the reaction-diffusion equations involves only terms in concentration of the species to which the discrete equation applies. That is, if there were two substances involved, O and R as above, then the discrete equation at a point $i$ had terms only in $C'_O$ for species O, and only $C'_R$ for species R. This made it straightforward to use the Thomas algorithm to reduce a system like (6.27) to (6.28), treating the two species' systems separately. They then get coupled through the boundary conditions.

When homogeneous reactions take place, it often happens that some of the discrete equations contain terms in concentration for more than the one species, and it is then not generally possible to use the simple Thomas algorithm to reduce the systems. These systems are said to be coupled. An example will illustrate this situation.

Consider the catalytic or EC′ reaction pair as described in Sect. 2.4, page 25, Eq. (2.75). The species designations A and B are now written as O and R, and the reaction pair then is

$$\begin{aligned} \mathrm{O} + e^- &\rightleftharpoons \mathrm{R} \\ \mathrm{R} &\rightarrow \mathrm{O}\,. \end{aligned} \tag{6.54}$$

The derivation of the discrete equations corresponding to this reaction pair will be given in Chap. 8 and it will suffice here to provide the general form they will take:

$$\begin{aligned} C'_{O,i-1} + a_{O,1}(i)C'_{O,i} + a_k(i)C'_{R,i} + a_{O,2}(i)C'_{O,i+1} &= b_{O,i} \\ C'_{R,i-1} \quad + (a_{R,1}(i) - a_k(i))C'_{R,i} + a_{R,2}(i)C'_{R,i+1} &= b_{R,i}\,. \end{aligned} \tag{6.55}$$

The coefficients $a_{.,1}(i)$ and $a_{.,2}(i)$ arise from the particular spatial approximation of the second derivative, while the $a_k(i)$ come from the homogeneous chemical reaction rate, as will be described in Chap. 8.

It will always be the case that the extra term, as seen here in the first equation for species O, lies at index $i$ only. As stated above, it is not generally possible to start at the outer limit for $X$ and reduce these two equations to fewer unknowns, as with uncoupled cases. In fact, in this particular case, this can be done using a slightly complicated trick but this will not be dealt with here. It is detailed in [9].

There are the usual boundary conditions depending on the experiment performed on this system. One possible way to handle all this is simply to write out the whole system as a large linear system, expand that to include the boundary conditions, and solve. This, "brute force" approach (see below), has in fact been used [10, 11] and can even be reasonably efficient if the number of equations is kept low, by use, for example, of unequal intervals, described in Chap. 7. If the equations in such a system are arranged in the order as above (6.55), it will be found that it is tightly banded, except for the first two rows for the boundary conditions, which may have a number of entries up to the number $n$ used for the current approximation.

A better alternative approach is the block-matrix solution method, made widely known to electrochemists by Rudolph [12]. It was in fact known before 1991 under various names, notably block-tridiagonal [13–17]—citing only electrochemical sources—(Newman using it tacitly in his BAND subroutine [15], repeated in an updated form in [16, p. 619]). Bieniasz [18] provides an extensive history of block matrix methods from the numerical literature, going back as far as 1952 [19]. If one lumps the large matrix into a matrix of smaller matrices and vectors, the result is a tridiagonal system that is amenable to more efficient methods of solution. In the present context, we define some vectors

$$\mathbf{C}'_x \equiv \begin{bmatrix} C'_{O,x} \\ C'_{R,x} \end{bmatrix} \tag{6.56}$$

where $x$ can be $i-1$, $i$ or $i+1$. The equation pair (6.55) can be partitioned into three vertical slices involving such vectors, and then rewritten in the matrix–vector form,

$$\mathbf{C}'_{i-1} + \mathbf{A}_i\mathbf{C}'_i + \mathbf{a}_2\mathbf{C}'_{i+1} = \mathbf{B}_i \tag{6.57}$$

in which we have another vector and two matrices for the coefficients:

$$\mathbf{B}_i \equiv \begin{bmatrix} b_{O,i} \\ b_{R,i} \end{bmatrix} , \tag{6.58}$$

$$\mathbf{A}_i \equiv \begin{bmatrix} a_{O,1}(i) & a_k(i) \\ 0 & a_{R,1}(i) - a_k(i) \end{bmatrix} , \tag{6.59}$$

and

$$\mathbf{a}_2(i) \equiv \begin{bmatrix} a_{O,2}(i) & 0 \\ 0 & a_{R,2}(i) \end{bmatrix} . \tag{6.60}$$

The point of this exercise is that (6.57) now is very like the previous single-species equation, (6.2), except that it involves concentration vectors and coefficient vectors and matrices, rather than all scalars as in (6.2). In Chap. 8, details will be given on how this equation can be reduced to two terms using a block-Thomas process; suffice it to say here that the process is analogous to that for a single species, making use of the $N$th equation containing the outer boundary vector $\mathbf{C}_{N+1}$. The result is that the system (6.57) is replaced by a new reduced system, the first few equations of which are

$$\begin{aligned} \mathbf{C}'_0 + \mathbf{A}'_1\mathbf{C}'_1 &= \mathbf{B}'_1 \\ \mathbf{C}'_1 + \mathbf{A}'_2\mathbf{C}'_2 &= \mathbf{B}'_2 \\ \mathbf{C}'_2 + \mathbf{A}'_3\mathbf{C}'_3 &= \mathbf{B}'_3 \ , \\ \dots \end{aligned} \tag{6.61}$$

very similar in form to the system (6.3). Not surprisingly, the **u-v** device used for the single-species case can be devised for the matrix–vector case, and will be called the **U-V** device. **U** becomes a vector and **V** a matrix. We start by defining starting values:

$$\mathbf{U}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{6.62}$$

and

$$\mathbf{V}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.63}$$

which clearly allows the tautological statement analogous to (6.15),

$$\mathbf{C'}_0 = \mathbf{U}_0 + \mathbf{V}_0\mathbf{C'}_0 \ . \tag{6.64}$$

Looking at (6.61), first line, we can, analogously to (6.6), write the vector $\mathbf{C'}_1$ as a linear function of $\mathbf{C'}_0$, that is,

$$\mathbf{C'}_1 = \mathbf{A'}^{-1}\mathbf{B'}_1 - \mathbf{A'}^{-1}\mathbf{C'}_0 \tag{6.65}$$

which is then rewritten as

$$\mathbf{C'}_1 = \mathbf{U}_1 + \mathbf{V}_1\mathbf{C'}_0 \tag{6.66}$$

with obvious definitions for the $\mathbf{U}_1$ and $\mathbf{V}_1$, in view of (6.65) and (6.64). This process can be repeated for indices $i$ equal to $2, 3, \ldots$, and the general recursive formulae for the $\mathbf{U}$ and $\mathbf{V}$ are as follows:

$$\mathbf{U}_i = \mathbf{A'}^{-1}\left(\mathbf{B'}_i - \mathbf{U}_{i-1}\right) \tag{6.67}$$

and

$$\mathbf{V}_i = -\mathbf{A'}^{-1}\mathbf{V}_{i-1} \ . \tag{6.68}$$

The process, as for the uncoupled case of the **u-v** device above, needs to be carried forward only to $i = n - 1$. It yields $n - 1$ equations

$$\mathbf{C'}_i = \mathbf{U}_i + \mathbf{V}_i\mathbf{C'}_0 \tag{6.69}$$

which can finally be used for the boundary conditions.

We are still dealing with two species as in the uncoupled case and the same boundary conditions apply; they are reformulated in the present matrix–vector form here. As noted above, there is a common condition for all experiments, the flux condition (6.30), generalised to include the normalised diffusion coefficients, to the gradient condition (6.32), and we now write out its discrete form fully, pairing the two species' terms for each spatial index:

$$d_O\beta_0C'_{O,0} + d_R\beta_0C'_{R,0} + \cdots + d_O\beta_{n-1}C'_{O,n-1} + d_R\beta_{n-1}C'_{R,n-1} = 0 \tag{6.70}$$

and invoking vector notation, writing $\mathbf{C}'_i$ for $[C'_{O,i}\ \ C'_{R,i}]^T$, this becomes

$$\beta_0[d_O\ \ d_R]\mathbf{C}'_0 + \beta_1[d_O\ \ d_R]\mathbf{C}'_1 + \cdots + \beta_{n-1}[d_O\ d_R]\mathbf{C}'_{n-1} = 0 \tag{6.71}$$

which will be one of the two equations needed for all cases. It will be combined with the particular equations for the cases Cottrell, quasi/irreversible and controlled current.

For the simple Cottrell case, we have

$$C'_{O,0} = 0 \tag{6.72}$$

and, for convenience in what follows, this is expanded to include the other species,

$$C'_{O,0} \quad + \left(0 \cdot C'_{R,0}\right) = 0 \tag{6.73}$$

or

$$\beta_0[1\ \ 0]\mathbf{C}'_0 = 0 \tag{6.74}$$

in vector form (multiplying by $\beta_0$ for convenience). At the risk of repetition but in order to make the next step clear, this equation is now paired with (6.71):

$$\begin{array}{llll} \beta_0[1\ \ 0]\mathbf{C}'_0 & & & = 0 \\ \beta_0[d_O\ \ d_R]\mathbf{C}'_0 & +\beta_1[d_O\ \ d_R]\mathbf{C}'_1 + \ldots & +\beta_{n-1}[d_O\ \ d_R]\mathbf{C}'_{n-1} & = 0 \end{array}$$

and these can be combined in one vector–matrix equation,

$$\beta_0 \begin{bmatrix} 1 & 0 \\ d_O & d_R \end{bmatrix} \mathbf{C}'_0 + \beta_1 \begin{bmatrix} 0 & 0 \\ d_O & d_R \end{bmatrix} \mathbf{C}'_1 + \cdots + \beta_{n-1} \begin{bmatrix} 0 & 0 \\ d_O & d_R \end{bmatrix} \mathbf{C}'_{n-1} = 0\,. \tag{6.75}$$

This is now written in the more general form,

$$\beta_0\mathbf{M}_0\mathbf{C}'_0 + \beta_1\mathbf{M}_1\mathbf{C}'_1 + \cdots + \beta_{n-1}\mathbf{M}_{n-1}\mathbf{C}'_{n-1} = 0 \tag{6.76}$$

with

$$\mathbf{M}_0 = \beta_0 \begin{bmatrix} 1 & 0 \\ d_O & d_R \end{bmatrix} \tag{6.77}$$

and

$$\mathbf{M}_i = \beta_i \begin{bmatrix} 0 & 0 \\ d_O & d_R \end{bmatrix} \tag{6.78}$$

for all $0 < i < n$. This equation contains the vectors $\mathbf{C}'_i$, and we can apply the **U-V** relations (6.69) to put it all in terms of the one unknown vector $\mathbf{C}'_0$,

$$\beta_0 \mathbf{M}_0 \left(\mathbf{U}_0 + \mathbf{V}_0 \mathbf{C}'_0\right) + \cdots + \beta_{n-1} \mathbf{M}_{n-1} \left(\mathbf{U}_{n-1} + \mathbf{V}_{n-1} \mathbf{C}'_0\right) = 0 \tag{6.79}$$

and, defining the matrices

$$\mathbf{P} \equiv \sum_{i=0}^{n-1} \beta_i \mathbf{M}_i \mathbf{V}_i \ , \tag{6.80}$$

$$\mathbf{Q} \equiv -\sum_{i=0}^{n-1} \beta_i \mathbf{M}_i \mathbf{U}_i \ , \tag{6.81}$$

Eq. (6.79) then becomes

$$\mathbf{P}\mathbf{C}'_0 = \mathbf{Q} \tag{6.82}$$

which can readily be solved for the boundary values.

It will be seen that the equation always takes this form except for constant current and in that case, only a slightly different one. The differences lie in the definitions of the **M** matrices.

For the reversible case, apart from the flux condition, there is the Nernst equation, previously shown to be

$$C'_{O,0} - \exp(p') C'_{R,0} = 0 \tag{6.83}$$

now written as

$$\begin{bmatrix} 1 & -\exp(p') \end{bmatrix} \mathbf{C}'_0 = 0 \tag{6.84}$$

which is combined with the flux equation (6.71). This time the two equations are not presented, because they follow the above pattern for the Cottrell case, ending with the same equation (6.82), with **P** and **Q** generated as sums as in (6.80) and (6.81), the difference being in the first **M**, here given by

$$\mathbf{M}_0 = \beta_0 \begin{bmatrix} 1 & \exp(p') \\ d_O & d_R \end{bmatrix} \tag{6.85}$$

and the other $\mathbf{M}_i$ exactly as in (6.78).

For the quasireversible case, the flux condition is combined with the Butler–Volmer equation, as given above in (6.39), (6.40) and (6.41), the latter now to be written in long-hand as

$$\beta_0 C'_{O,0} + \beta_1 C'_{O,1} + \cdots + \beta_{n-1} C'_{O,n-1} = \frac{K_f H}{d_0} C'_{O,0} - \frac{K_b H}{d_R} C'_{R,0} \tag{6.86}$$

and collecting terms, again pairing the two boundary values,

$$\beta_0 \left[ \left(1 - \frac{K_f H}{d_O \beta_0}\right) C'_{O,0} + \frac{K_b H}{d_0 \beta_0} C'_{R,0} \right] + \beta_1 C'_{O,1} + \cdots + \beta_{n-1} C'_{O,n-1} = 0 \tag{6.87}$$

which leads again to the same equations, with the only difference here lying in $\mathbf{M}_0$, now given by

$$\mathbf{M}_0 = \begin{bmatrix} \left(1 - \frac{K_f H}{d_O \beta_0}\right) & \frac{K_b H}{d_O \beta_0} \\ d_O & d_R \end{bmatrix} \tag{6.88}$$

and again, the other $\mathbf{M}_i$ as in (6.78).

The totally irreversible case is again obtained by setting $K_b$ to zero in the above equations.

This leaves the controlled current case. As noted above (Sect. 6.3), it is the current, not the gradient, that is controlled, so the equation is

$$G = \frac{d_0}{H} \sum_{i=0}^{n-1} \beta_i C'_{O,i} \tag{6.89}$$

where $G$ is the dimensionless current imposed. So, expanding the sum, working in the (zero-weighted) terms in $C'_{R,i}$ and going straight into the vector notation, this becomes

$$\beta_0 \,[1 \;\; 0]\, \mathbf{C}'_0 + \beta_1 \,[1 \;\; 0]\, \mathbf{C}'_1 \cdots + \beta_{n-1} \,[1 \;\; 0]\, \mathbf{C}'_{n-1} = GH/d_0 \tag{6.90}$$

and we note that this differs from all the equations up till now in this section, in that the right-hand side is not zero. Combined with the inevitable flux condition (6.71), this yields the slightly different matrix/vector equation

$$\mathbf{P}\mathbf{C}'_0 = \begin{bmatrix} \frac{GH}{d_0} \\ 0 \end{bmatrix} + \mathbf{Q} \tag{6.91}$$

again readily solved.

## 6.5 Brute Force

In the last two sections, we have applied increasingly tricky devices to solve what amounts to systems of equations, in order to make their solution efficient. Even the two-species uncoupled case involved the generation of the **u** and **v** vectors and the solution of small (2 × 2) systems to obtain the boundary values. In the case of coupled systems, the problems mount and matrix–vector equations had to be used.

One might ask, is all this necessary? It is not. In the next chapter, unequal intervals are described, and these make it possible to reduce the number of points $N$ in space to quite reasonably small values. Whereas with equal intervals we might need some hundreds of sample points along $X$, with unequal intervals we can make do with as few as 15 or so. This means that systems such as (6.2) have only as few equations as that or, with two species involved, twice that many (not counting the boundary equations for the moment). The full systems are strongly banded if the equations are ordered suitably, and this banding invites the use of more efficient methods of solution, but if computers can do it fast enough by simply solving the whole system without regard to the bandedness, tricks might not be needed. All we then need to do is to add the equations for the boundary conditions (that is, include the $C'_0$ variables in the system) and call a matrix solver for the solution. This was mentioned by Rudolph [20] but was at the time considered "prohibitively expensive" in computer time. Meanwhile, however, computers have become much faster and the idea has been investigated [11, 21] and found, in some cases, to be about as good as the tricky methods and in some cases even better. As well, there are some reasonably simple methods of varying complexity to make such whole-system solutions more efficient. These will be mentioned below. First, a more concrete description of what is meant is presented here.

A simple example would be the single-species case and Cottrell. Then, the system (6.2) would be augmented by adding, at the top, the Cottrell condition $C'_0 = 0$ and the matrix equation is

$$\begin{bmatrix} 1 & 0 & \dots & & & & \\ 1 & a_{1(1)} & a_{2(1)} & 0 & \dots & & \\ 0 & 1 & a_{1(2)} & a_{2(2)} & 0 & \dots & \\ & & \ddots & & \ddots & & \\ & & & 1 & a_{1(N-1)} & a_{2(N-1)} & \\ & & & & 1 & a_{1(N)} & \end{bmatrix} \begin{bmatrix} C'_0 \\ C'_1 \\ C'_2 \\ \vdots \\ C'_{N-1} \\ C'_N \end{bmatrix} = \begin{bmatrix} 0 \\ b_1 \\ b_2 \\ \vdots \\ b_{N-1} \\ b_N - a_{2(N)} C'_{N+1} \end{bmatrix} \tag{6.92}$$

where the known outer boundary term, $C'_{N+1}$ has been carried over to the vector of knowns. This is a trivial example but serves to explain the idea. It is in fact a tridiagonal system, amenable to the Thomas algorithm to be described in Chap. 8. Extension to the other more complicated cases is obvious, except where there are two species, when the optimal order of the unknowns vector $\mathbf{C}'$ is to pair the terms for the two species, that is,

$$\mathbf{C}' \equiv \left[ C'_{O,0}\ C'_{R,0}\ C'_{O,1}\ C'_{R,1} \dots\ C'_{O,N}\ C'_{R,N} \right]^T \tag{6.93}$$

which ensures tight banding.

Extension to the other less trivial cases appears straightforward. Most boundary conditions will put up to $2n$ elements into the first and second rows. For those cases involving coupled equations, the rows after the second will contain five elements. It can be seen from the first row of (6.55) that there needs to be a zero inserted after the $C'_{O,i-1}$, for the nonexistent $C'_{R,i-1}$, but not a similar insertion after the last element, or a total of five. Thus, if one eliminates the excess elements in the first two rows, one can then use a solver for pentadiagonal systems, which is also quite feasible.

## 6.6 A General Formalism

Sometimes, when trying out a new method when efficiency is not (initially) of highest priority, or when doing a stability study, it can be of advantage to have a general formula for all possible boundary conditions. An early use of such a formula is seen in [22], and the formula is also seen in some texts such as [2]. In the electrochemical context, it has been presented a few times in recent years [23–25]. The formula is given in the form of [23]

$$g + rc_0 - d\left(\frac{\partial c(0,t)}{\partial x}\right) = 0\,. \tag{6.94}$$

The constants $g$, $r$ and $d$ can take on various values to express any given boundary condition. Thus, if we set $d = 0$ and $r \neq 0$, we are left with the general form of the Dirichlet condition and specifically with $r = 1$ and $g = 0$ we have the Cottrell condition, while Eq. (6.94) expresses Robin conditions. The constant $r$ expresses the heterogeneous rate constant (this formula only considers a single species, so an irreversible reaction is implied).

The Cottrell case is simple, and needs no further comment. The other two cases can be usefully expressed in a different manner. The derivative is expressed as the $n$-point approximation, giving

$$g + rc_0 - d\sum_{i=0}^{n-1}\beta_i c_i = 0 \tag{6.95}$$

or, removing the $c_0$ element from the sum,

$$g + (r - d\beta_0)\,c_0 - d\sum_{i=1}^{n-1}\beta_i c_i = 0 \tag{6.96}$$

giving

$$c_0 = \frac{d\sum_{i=1}^{n-1} \beta_i c_i \quad - g}{r - d\beta_0} \tag{6.97}$$

and dividing by $-\beta_0$ and setting $d = 1$,

$$c_0 = \frac{-b}{\beta_0} \left\{ \sum_{i=1}^{n-1} \beta_i c_i \quad - g \right\} \tag{6.98}$$

with

$$b = \left( 1 - \frac{r}{\beta_0} \right)^{-1} . \tag{6.99}$$

The convenient thing here is that we now have the whole spectrum of conditions from a very fast reaction ($b = 0$, implying $r \rightarrow \infty$), through medium fast reactions (medium values of $b$ and thus $r$) to the controlled current case ($b = 1$). The first case also encompasses the Cottrell case.

This formula has been used in some stability studies [23, 26, 27].

## References

1. Großmann C, Roos HG (1994) Numerik partieller Differentialgleichungen. Teubner, Stuttgart
2. Strikwerda JC (1989) Finite difference schemes and partial differential equations. Wadsworth and Brooks/Cole, Pacific Grove, CA
3. Lapidus L, Pinder GF (1982) Numerical solution of partial differential equations in science and engineering. Wiley, New York
4. Serna C, López-Tenés M, González J (2001) Reversible multistep electrode processes. Consideration of the bulk presence of intermediate species and of the values of the diffusion coefficients in voltammetry. Electrochim Acta 46:2699–2709
5. Pedersen SU, Christensen TB, Thomasen T, Daasbjerg K (1998) New methods for the accurate determination of extinction and diffusion coefficients of aromatic and heteroaromatic radical anions in N,N-dimethyl formamide. J Electroanal Chem 454:123–143
6. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
7. Galus Z (1994) Fundamentals of electrochemical analysis, 2nd edn. Ellis Horwood, New York (trans: Chalmers RA, Bryce WAJ (eds))
8. Rudolph M, Reddy DP, Feldberg SW (1994) A simulator for cyclic voltammetry responses. Anal Chem 66:589A–600A
9. Britz D (2006) Modified Thomas algorithm for the digital simulation of the catalytic EC' mechanism under Cottrellian conditions. Int J Electrochem Sci 1:1–11
10. Balslev H, Britz D (1992) Direct digital simulation of the steady-state limiting current at a rotating disk electrode for a complex mechanism. Acta Chem Scand 46:949–955
11. Britz D (1996) Brute force digital simulation. J Electroanal Chem 406:15–21

12. Rudolph M (1991) A fast implicit finite difference algorithm for the digital simulation of electrochemcial processes. J Electroanal Chem 314:13–22
13. Aguilella VM, Garrido J, Mafé S, Pellicer J (1986) A finite-difference method for numerical solution of the steady-state Nernst-Planck equations with non-zero convection and electric current density. J Membr Sci 28:139–149
14. Fan D, White RE (1991) Modification of Newman's BAND(J) subroutine to multi-region systems containing interior boundaries: MBAND. J Electrochem Soc 138:1688–1691
15. Newman J (1973) Electrochemical systems. Prentice-Hall, Englewood Cliffs, NJ
16. Newman J, Thomas-Alyea KE (2004) Electrochemical systems, 3rd edn. Wiley, Hoboken, NJ
17. White RE (1978) On Newman's numerical technique for solving boundary value problems. Ind Eng Chem Fundam 17:367–369
18. Bieniasz LK (2001) Extension of the Thomas algorithm to a class of algebraic linear equation systems involving quasi-block-tridiagonal matrices with isolated block-pentadiagonal rows, assuming variable block dimensions. Computing 67:269–285. Erratum: ibid. 70:275 (2003)
19. Karlqvist O (1952) Numerical solution of elliptic difference equations by matrix methods. Tellus Q J Geophys 4:374–384
20. Rudolph M (1995) Digital simulations with the fast implicit finite difference algorithm: the development of a general simulator for electrochemical processes. In: Rubinstein I (ed) Physical Electrochemistry. Marcel Dekker, New York, pp 81–129
21. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376
22. Rizzo FJ, Shippy DJ (1970) A method of solution for certain problems of transient heat conduction. AIAA J 8:2004–2009
23. Bieniasz LK, Østerby O, Britz D (1995) Numerical stability of finite difference algorithms for electrochemical kinetic simulations: matrix stability analysis of the classic explicit, fully implicit and Crank-Nicolson methods and typical problems involving mixed boundary conditions. Comput Chem 19:121–136
24. Britz D, Strutwolf J (2000) Higher-order spatial discretisations in electrochemical digital simulation. 1. Combination with the BDF algorithm. Comput Chem 24:673–684
25. Strutwolf J (1995) Digitale Simulation elektrochemischer Systeme: Untersuchungen zeitabhängiger Phänomene an rotierenden Scheibenelektroden und Analyse von Cyclovoltammogrammen durch direkte Simulation. Ph.D. thesis, Universität Bielefeld, Germany
26. Britz D, Østerby O, Strutwolf J, Svennesen TK (2002) High-order spatial discretisations in electrochemical digital simulations. 3. Combination with the explicit Runge-Kutta algorithm. Comput Chem 26:97–103
27. Bieniasz LK, Østerby O, Britz D (1997) The effect of the discretization of the mixed boundary conditions on the numerical stability of the Crank-Nicolson algorithm of electrochemical kinetic simulations. Comput Chem 21:391–401

# Chapter 7
# Unequal Intervals

In the preceding chapters, a grid with equal intervals in both time and space was assumed (Fig. 1.1, page 3). There are several reasons for deviating from equal space intervals. Firstly, one wants both to minimise the number of points over the concentration profile and, at the same time, to have close spacing near the electrode. Secondly, in some simulations, there arise sharp concentration changes somewhere in the diffusion space, usually adjacent to the electrode. One then wants to have close spacing in such regions in order to be able to simulate concentration changes at all. This points to adaptive techniques; but first the simpler fixed unequal grid techniques will be dealt with.

In this chapter, only one-dimensional unequal intervals will be described. Mapping techniques for higher-dimensional simulations are left to Chap. 12.

Consider Fig. 2.4 on page 17, showing the concentration profile for a Cottrell simulation at different times. It is clear that at small $T$ the concentration change occurs only over a narrow range of $X$ close to the electrode, so that equal intervals in $X$ would be wasteful at larger $X$. An unequal spacing of the intervals could not only provide more detail near the electrode where it is needed, but also make do with fewer points by wider spacing far away from the electrode. So some kind of grid stretching is indicated on this account.

If there are homogeneous chemical reactions, they may give rise to reaction layers that can, for high reaction rates, be very thin. In order to get reasonable simulation results, at least a few points are needed within such a layer. If equal intervals in $X$ are used, this means using a very large value of $N$ and correspondingly long computation times. One needs to have an idea of the thickness, $\mu^*$, defined on page 25 (2.79), and set the position of the points accordingly, as described in the following sections.

There are several approaches to implementing grid stretching. The two competing approaches are (1) the direct application of a stretched grid, discretising directly on the unequal grid, and (2) the transformation of the equation to new coordinates and using equal intervals there. There is a wealth of literature on this

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8_7

subject. Noye [1], and Hunter and Jones [2] recommend transformation, as does an early study by Crowder and Dalton [3]. The much cited comparison by Kalnay de Rivas [4] reached the same conclusion. However, Rudolph [5] showed conclusively that, under the conditions of electrochemical simulations at least, the situation is the reverse. He showed that both the current approximation and the second, spatial, derivative as computed directly from an unequally spaced grid are more accurate than those computed from a transformed grid with equal intervals. The reason for the better performance of direct discretisation appears to be that concentration profiles tend to be close to linear near the electrode, so that the current approximation can be calculated quite well with only a few points (Rudolph always uses just two), whereas in the transformed space (see below, Sect. 7.1), the profile becomes curved near the electrode and more points are needed for a good approximation. Exactly why the second, spatial, derivative is also more accurate when calculated directly is not clear. Numerical experiments performed by the present authors show that, for several different (artificial) profile functions, including the realistic one of erf($x$), which often resembles real concentration profiles, the second derivative calculated on the transformed grid is poor, especially near the electrode, where accurate values are most needed. Direct calculation on the unequal grid yields roughly the same accuracy right across the profile.

As well, it will be seen that the formulation of the second spatial derivative on a general grid, spaced in some unspecified way, is rather flexible and permits easy replacement, in a given program, of the stretching function used including, if one desires, equal spacing, or even arbitrary placement of each point.

## 7.1 Transformation

Transformation for electrochemical work was proposed in the now classic paper by Joslin and Pletcher [6]. They described a transformation, say from $X$ to $Y$, such that equal intervals in $Y$ are a mapping of (correspond to) unequal intervals in $X$. The aim is to find a transformation function which produces in $Y$-space a concentration profile that resembles a straight line as much as possible.

The general treatment is as described by Joslin and Pletcher [6]. Assume an arbitrary transformation function, $Y = f(X)$ mapping points in $X$ onto the new axis $Y$ and its inverse, $X = g(Y)$. Then the right-hand diffusion term in the diffusion equation, $\frac{\partial^2 C}{\partial X^2}$ becomes, by the rules of elementary calculus,

$$\frac{\partial}{\partial X}\left(\frac{\partial C}{\partial X}\right) = \frac{1}{g'(Y)}\frac{\partial}{\partial Y}\left(\frac{1}{g'(Y)}\frac{\partial C}{\partial Y}\right) . \tag{7.1}$$

This can be expanded further to

$$\frac{\partial}{\partial X}\left(\frac{\partial C}{\partial X}\right) = \frac{1}{g'(Y)}\left(\frac{1}{g'(Y)}\frac{\partial^2 C}{\partial Y^2} + \left(\frac{\partial}{\partial Y}\left(\frac{1}{g'(Y)}\right)\right)\frac{\partial C}{\partial Y}\right) . \tag{7.2}$$

The work of Seeber and Stefani and Feldberg (both in 1981) [7, 8] indirectly provides a useful transformation function that has some convenient properties. The function is

$$f(X) = Y = \ln(1 + aX) \tag{7.3}$$

where $a$ is an adjustable parameter. Inserting this into (7.2), recognising that

$$g(Y) = (e^Y - 1)/a \qquad \text{and thus} \qquad g'(Y) = e^Y/a , \tag{7.4}$$

the new dimensionless diffusion equation in $Y$-space is then

$$\frac{\partial C}{\partial T} = a^2 e^{-2Y}\left(\frac{\partial^2 C}{\partial Y^2} - \frac{\partial C}{\partial Y}\right) . \tag{7.5}$$

Note that if the original equation to be solved contains homogeneous chemical terms, these do not change upon transforming the equation, since they give rise to additional terms not involving $X$ or $Y$.

The transformation function (7.3) is mathematically (approximately) equivalent to the stretching function (7.16), as is shown in Appendix C, where the relation between the respective adjustable parameters is given.

The gradient $G$ is conveniently calculated on the grid in $Y$, and it is easy to show that this is simply

$$G = \left.\frac{\partial C}{\partial X}\right|_{X=0} = a \left.\frac{\partial C}{\partial Y}\right|_{Y=0} \tag{7.6}$$

which seems very convenient, requiring only a call to the routine that evaluates the $\mathcal{G}$ function in $Y$-space and a multiplication by the parameter $a$. The problem, as Rudolph showed [5], is that this yields a poor $G$-value, unless a large $n$ is used (6 or even 7). This is not a bad thing in itself, as we have functions for $G$ that we can simply call. However, derivative boundary conditions involving many points become messy. Rudolph uses just two points, arguing that if the first point is sufficiently close to the electrode, as it is with severe stretching, two points are good enough, and this simplifies the discretisation of the boundary conditions a lot. There are some arguments for using $n = 3$; Bieniasz [9, 10] points out that if a second-order second spatial derivative is used for the simulation, then a matching second-order (3-point) $G$-approximation is best. On the other hand, the second spatial derivative directly discretised on using three points on an unequal grid is in fact a first-order approximation, arguing for Rudolph's two-point $G$. This will be a matter of individual choice.

### 7.1.1 Discretising the Transformed Equation

Transformation (7.3) leads to the new diffusion equation (7.5) in $Y$-space. Although it is fairly obvious how the new right-hand side is discretised, for completeness, this will be described here.

Instead of a number of sample points in $X$, we now have a number of equally spaced points along the new coordinate $Y$ with a spacing of $\delta Y$. Without considering which simulation algorithm is to be used, we discretise the new Eq. (7.5) at the point $Y_i$ as follows:

$$\delta C_i \approx \delta T a^2 exp(-2Y_i) \left( \frac{C_{i-1} - 2C_i + C_{i+1}}{\delta Y^2} - \frac{C_{i+1} - C_{i-1}}{2\delta Y} \right) \tag{7.7}$$

(with $\delta C_i = C'_i - C_i$) and given that $Y_i = i\ \delta Y$, this rearranges to

$$\delta C_i \approx \lambda_i \left( (1 + \tfrac{1}{2}\delta Y) C_{i-1} - 2C_i + (1 - \tfrac{1}{2}\delta Y) C_{i+1} \right) \tag{7.8}$$

with $\lambda_i$ defined as

$$\lambda_i = a^2 \exp(-2i\delta Y) \frac{\delta T}{\delta Y^2} \ . \tag{7.9}$$

The coefficients in the right-hand term in brackets in (7.8) can be precomputed, as can the row of $\lambda_i$ values. Further details of how all this is implemented are given in Chap. 8 for the respective simulation algorithms.

As mentioned above, Rudolph [5] pointed out that this discretisation yields very poor values and leads ultimately to poor simulation performance, compared to direct discretisation on an uneven grid, see below. Tests show that particularly at small $X$ values, near the electrode where the greatest changes occur, the second spatial derivatives as seen in (7.7) are approximated poorly. Rudolph [11, 12] and Bieniasz [13] showed that if what we might call the semi-transformed equation (7.1) is used, rather than the fully transformed equation, this problem is eliminated. Doing this in a consistent manner, and assuming general transformation functions $f(X)$ and $g(Y)$, we can write for the $i$th point the approximation

$$\frac{1}{g'(Y)} \frac{\partial}{\partial Y} \left( \frac{1}{g'(Y)} \frac{\partial C}{\partial Y} \right) \approx \frac{1}{g'(Y)} \frac{1}{\delta Y} \left( \frac{C_{i+1} - C_i}{g'(Y_{i+\frac{1}{2}}) \delta Y} - \frac{C_i - C_{i-1}}{g'(Y_{i-\frac{1}{2}}) \delta Y} \right) \ . \tag{7.10}$$

Using the transformation (7.3) and thus substituting for $g'(Y)$ as given in (7.4) at the indices given and rearranging a little, this becomes

$$\delta C_i \approx \lambda_i \left(\exp(-\tfrac{1}{2}\delta Y)(C_{i+1} - C_i) - \exp(\tfrac{1}{2}\delta Y)(C_i - C_{i-1})\right) \tag{7.11}$$

with $\lambda_i$ as defined above (7.9). Some tests indicate that this is a much better approximation, giving derivatives of roughly the same accuracy over the whole spatial range. The accuracy is comparable to that of direct discretisation on the uneven grid, described below. Incorporation into the whole diffusion equation, as was done for the completely transformed diffusion equation in (7.8), is obvious from here on.

### *7.1.2 Choice of Transformation Parameters*

We have seen from the above that, in some way or other, we choose the value of $H_1 = X_1$. We also have a maximum value $X_{max}$ along $X$, which depends on the experiment. Using (7.3), these two values provide the two equivalent values in $Y$-space. The $Y$-value corresponding to $X_1$ is also the interval in $Y$, as these are all equal. The equations are

$$\delta Y = \ln(1 + aX_1) \tag{7.12}$$

and

$$Y_N = \ln(1 + aX_{max}) \tag{7.13}$$

which set the number of intervals in $Y$,

$$N = Y_N/\delta Y \tag{7.14}$$

(rounded up, thus correcting $Y_N$ slightly). Knowing $X_1$, there are then two parameters to be determined, $a$ and $N$. These are dependent on each other, so the choice of one sets the other.

The easy alternative is to set $a$. One develops a feeling for what value might be a good one. Having set this value and knowing that of $X_{max}$, the above Eqs. (7.12)–(7.14) yield $N$.

Alternatively, one might want to set $X_1$ and $N$ and find an $a$ value that provides these. Dividing (7.13) by (7.12) and noting (7.14), we obtain

$$f(a) = N\ln(1 + aX_1) - \ln(1 + aX_{max}) \tag{7.15}$$

which can be solved numerically for that $a$ which gives $f(a) = 0$. There are two solutions. The trivial (and unwanted) solution is $a = 0$. What makes the calculation rather easy is the fact that we do not need a very accurate value for the $a$ parameter. So a rough binary search will very quickly find a suitable value (see such elementary texts on numerical computing as [14–16]). A binary search will be found better here than the generally more efficient Newton method, which can point in the wrong direction and converge to the trivial solution, or lead to numerical problems (negative arguments to the log function).

Lastly, it is possible also to set $a$, $X_{max}$ and $N$, and to use them to find $\delta Y$ and thereby $X_1$. If it is done on a calculator beforehand, one sees what value results, before committing the chosen parameters to a simulation run.

## 7.2 Direct Application of an Arbitrary Grid

A stretched stack of boxes was used by Seeber and Stefani and by Feldberg [7, 8] for the box-method, to be described in Chap. 9. Pao and Dougherty [17] developed the same idea (and stretching function) in 1969, in the context of fluid dynamic simulations. This is the simple placement of points at increasing intervals, in some suitable point distribution or stretching function, and discretisation of the second derivative of concentration along $X$ on that unequal grid.

There are various ways of specifying the stretched point placement. The current favourite appears to be the exponentially expanding sequence of intervals $H$ along $X$ [7, 8],

$$H_i = H_{i-1}\,\gamma \tag{7.16}$$

or

$$H_i = H_1\gamma^{i-1} \tag{7.17}$$

starting at some chosen $H_1$ and choosing the stretching parameter $\gamma$ suitably. In Feldberg's case, the points thus generated are in fact box walls, but one could equally well use them with the point method as concentration nodes. Also, Feldberg uses a slightly different notation, setting not the $\gamma$ used here, but the related parameter $\beta = \ln(\gamma)$. The value of $\gamma$ is chosen such that $H_1$ is rather small, but the number of points in the diffusion region is also rather small. While with equal intervals, some hundreds of points might be needed, a suitable choice of $\gamma$ (for example, 1.2–1.5) can reduce their number to 10–20 (Feldberg suggests a $\beta$ range $0\ldots0.5$).

As will be seen below, the way stretched intervals are used here is that a set of positions in $X$ are specified. We must therefore convert the intervals formula above (7.17) to one in terms of $X$. For any $N > 0$,

$$X_N = H_1\sum_{k=1}^{N}\gamma^{k-1} = H_1\sum_{k=0}^{N-1}\gamma^{k} \tag{7.18}$$

and this is readily summed to give the expression

$$X_N = H_1 \frac{\gamma^N - 1}{\gamma - 1} \tag{7.19}$$

which is also the expression seen in Feldberg [8], albeit in terms of $\beta$. In practice, we find that $\gamma$ should not exceed about 1.2 or so [18], corresponding to a $\beta$ value just under 0.2. However, a recent work by Martínez-Ortiz et al. [19] contradicts this, using expansion factors up to 2 and obtaining good results.

The drawback of this point sequence (and most others except a sequence of equal intervals) is that the three-point approximation to the second derivative with respect to $X$ is then a first-order approximation, as was mentioned in Chap. 3, Sect. 3.8. The use of more than three points is thus indicated, and such approximations are described in Chaps. 3 and 9, and some formulas are given in Appendix A.

There is one unequal sequence of points for which the second derivative, when applied directly to the points, retains the second-order nature of an even point spacing. This was found by Sundqvist and Veronis [20] in 1970. Their stretching function was

$$H_i = H_{i-1}(1 + \alpha H_{i-1}) \; . \tag{7.20}$$

In the original form, the $\alpha$ factor was effectively normalised by dividing by the total extension $X_{max}$ of the diffusion space. In the present context, a suitable normalisation might be division by $H_1$, giving

$$H_i = H_{i-1}(1 + \alpha H_{i-1}/H_1) \; . \tag{7.21}$$

Then, this function will yield sequences somewhat similar to exponentially expanding sequences, by taking $\alpha$ equal to something like $\frac{1}{2}(\gamma - 1)$. This sequence has not become popular (perhaps because it has escaped notice). It might, however, be a useful alternative. Interestingly, Saul'yev mentions [21, p. 149] a private communication from A.A. Samarskii, who found precisely the same relation (7.20) and that it permits second-order approximations to the three-point second derivative.

A comparison of the two functions discussed here is shown in Fig. 7.1, presenting the distribution out to about $X = 6$ for the exponential sequence (7.16) for $\gamma = 1.5$, and the S&V sequence (7.21) for $\alpha = 0.2$. Both were started with a first interval $H_1$ of 0.05. The exponential sequence gives 10 points, ending at $X = 5.67$ and the other sequence gives 9 points ending at $X = 6.36$. It is seen that the S&V sequence produces a more drastic expansion of the spacing. Preliminary numerical

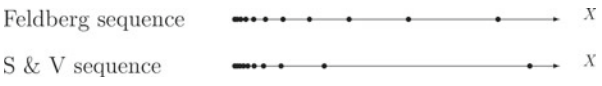


**Fig. 7.1** Points spaced unequally with the two functions

experiments by the present authors indicate that the second spatial derivatives on the S&V sequence are indeed mostly very accurate, but decline in accuracy at large $X$. In two comparison programs, one using the exponentially expanding and the other the S&V sequence, both starting with a base interval of 0.01 and using 50 points in the $X$ range 0–6 and discretising directly, a Cottrell simulation using 100 steps in time each of length 0.01, the exponentially expanding sequence showed an error in the final current at $T = 1$ of about $10^{-3}$ (relative), whereas the S&V sequence's error was $10^{-2}$. So, it appears that this point sequence might not be so good.

There is an inherently stretched grid implementation in the simulation technique called orthogonal collocation, to be discussed in Chap. 9. It will be seen that this can be extremely efficient but it suffers, as all fixed stretched grids do, from inflexibility, as is noted in general in Sect. 7.3.

An interesting special case, mentioned in Chap. 3, is that of the second derivative on four points, $u''_2(4)$. For arbitrarily (unequally) spaced points, this is a second-order accurate approximation and, as described in Chap. 9, it has some advantages. It allows the use of an efficient extended Thomas algorithm, rather than a pentadiagonal solver or a sparse solver required if more than four points are used for the approximation. There is one special case of this approximation, $\gamma = \sqrt{2}$, that is interesting in that it yields a third-order approximation, as found by Martínez-Ortiz et al. [22]. These authors also derived some conveniently compact specific approximation formulae for the exponentially expanding grid, for most cases of interest, obviating the need for a numerical computation of the approximation coefficients. The value $\sqrt{2}$ may appear a little large but if only a few points and a very small first interval are wanted, it might be useful. One would then have to find the first interval that satisfies the $\gamma$ value and the desired number of points in the space region, and this can be done by simple application of (7.19). As an example, if we want $N = 14$ and $X_{lim} = 6$, this makes $H_1 = 0.0195691$.

We have recently introduced another sequence of point positions, that we call damped exponentially expanding, or at times sigmoidal spacing [23, 24]. It becomes useful in cases where there are derivative boundary conditions at a far boundary. Then the large intervals resulting there from exponential expansion give rise to inaccurate derivative approximations, and smaller intervals are desirable. This can be realised by the usual exponentially expanding sequence going only halfway and then contracting again, but at the far boundary, the intervals need not be as small as are needed at the electrode, or at discontinuities such as electrode edges or corners met with in two-dimensional systems (see Chap. 12).

The sequence is similar to (7.16) but with the expansion factor now being damped

$$\gamma_i = 1 + (\gamma_1 - 1)\exp(-aX_i/X_{max}) \tag{7.22}$$

with $a$ a suitable factor, which we find about optimal at $a = 6$. Thus, the intervals expand at a decreasing rate, and towards the far boundary, they tend to a constant value, hence the term "sigmoidal". Figure 7.2 shows the result, and the sigmoidal nature of the intervals in the damped sequence.

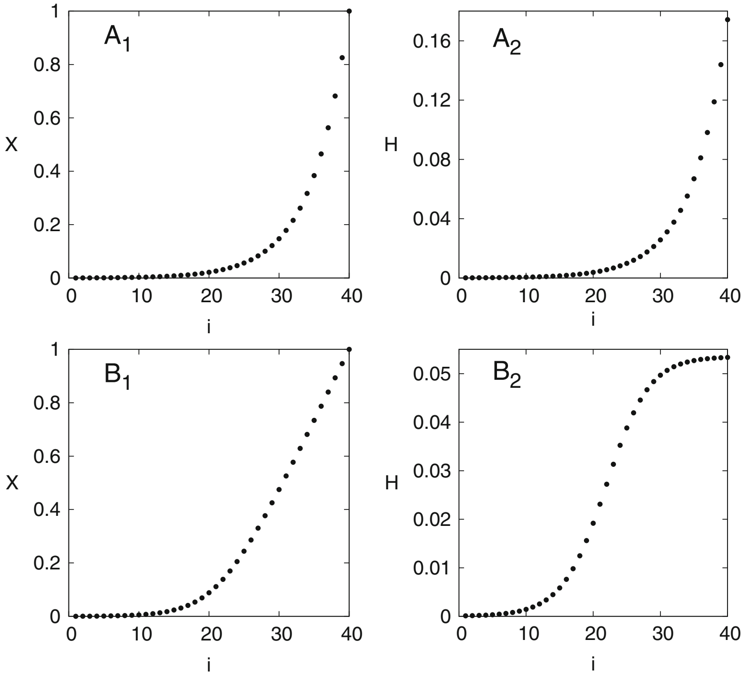


**Fig. 7.2** Comparison of an unconstrained and damped exponentially expanding point sequence, along the stretch $0 <= X <= 1$. The base $H_1$ was set at $10^{-4}$, giving, for 40 points, a fixed $\gamma$ of 1.21 for the unconstrained sequence and a base $\gamma_1$ of 1.35 for the damped sequence. *Top plots*, unconstrained exponentially expansion: ($A_1$), positions $X$ and ($A_2$), intervals $H_i$. *Bottom plots*, damped expansion: ($B_1$), positions $X$ and ($B_2$), intervals

The expression (7.22) does not lead simply to one in $X$ as (7.19), so a numerical search for the base $\gamma_1$ that satisfies the requirement that $X_N = X_{max}$ must be followed by the calculation of the $X$ positions as the running sum

$$X_i = \sum_{k=1}^{i} H_k \, . \tag{7.23}$$

All this is done in the subroutine `DAMPED_EXPANSION` in the Examples Appendix E.

## 7.2.1 Choice of Parameters

For any point sequence and a given stretching function, there are several parameters to choose. One is always $N$, the number of points along the profile. The others are the length of the first interval $H_1$ or (the same thing) the position of the first point next to the electrode $X_1$. This might then determine the function parameter. In most cases, setting $X_1$ is desirable; for example, in order to achieve a certain desired accuracy in the gradient $G$. If this is so, in both cases of the exponentially increasing intervals function (7.17) and (7.21), the stretching parameter then needs to be searched for by numerical means. Let the largest $X$-value be $X_{max}$, and the number of internal points be $N$. We have set the wanted $X_1$. For exponentially expanding intervals (7.17) we then apply (7.19) and seek a $\gamma$ value that satisfies it. A simple numerical (for example, binary) search finds $\gamma$. An example of such a search is shown in the function `EE_FAC` described in Appendix E. Less conveniently, one might choose $X_1$ and $\gamma$, and find out what $N$ then becomes, by a simple calculation. This is deemed less likely because one would usually want to have control over the $N$ value. The same goes for a damped exponentially expanding grid, where the starting $\gamma_1$ must be found.

### *7.2.2 Current and $C_0$ Approximations*

The formula (3.25) on page 45 can be used as is for approximating the current (or in fact the spatial gradient) on an arbitrary grid. The coefficients must be computed appropriately, and this can be done conveniently using the Fornberg algorithm [25], implemented in the routine `G0FORN` described in Appendix E.

For chronopotentiometry, where we have a known gradient and wish to compute the value of the concentration at the electrode that produces this value, the same formula as (5.10) on page 93 applies, again with appropriate coefficients. Again the Fornberg algorithm is most convenient and this is implemented in the Examples routine `C0FORN`.

## 7.3 Concluding Remarks on Unequal Spatial Intervals

The question arises of how low an $N$ value it is possible to work with and still get good results. The simulation package DigiSim due to Rudolph and Feldberg [26] routinely uses as few as 14 and is able to achieve sufficient accuracy in the current. This depends on one's definition of "sufficient". If 0.1 % accuracy is wanted, about 40 points in space might be optimal.

Clearly also, in order to choose a suitable set of parameters, one must know the requirements before the simulation. If some homogeneous rate constant changes during a series of program runs (for example, one in which such a rate constant is searched for), then the grid parameters should change. This makes adaptive grids more useful. These are described below.

As for the choice between direct discretisation on an arbitrarily spaced grid or the formulae for the semi-transformed or the transformed diffusion equation, the present authors now incline towards the first of these. Some formulae for the derivatives on arbitrarily spaced points are given in Chap. 3 and Appendix A, but in practice, it is convenient to use the Fornberg algorithm implemented in FORN, also in the same Appendix. The coefficients applying to each point can most conveniently be precomputed.

## 7.4 Unequal Time Intervals

Just as space can be divided into unequally spaced intervals, so might time also be unevenly divided. As with spatial intervals, there is the choice between discretising on an uneven time grid or using a transformation to a new time scale. Since, except for BDF methods, one usually differentiates with respect to time using only two time points (levels), transformation does not make sense here.

There can be several reasons for wanting unequal time intervals. One is that one may not want results at many equally spaced time intervals and only a few, possibly expanding, time intervals are wanted. This can of course be realised by a number of equal steps in time but only displaying results where wanted. More relevant are simulations of, for example, pulse voltammetry, where large changes occur just after the pulse onset but they slow down after that, so that expanding intervals are appropriate. Initial steps in this direction were taken by Flanagan et al. [27], Dillard et al. [28] and Nikolić [29], who used two different time intervals: largish intervals when the current does not change much, and finer intervals (1/100 to 1/9 as large) just after a pulse. Seeber and Stefani [7] used a rather complicated scheme, in which they used expanding intervals in space and direct discretisation on that grid; and in recognition of the fact that, far away from the electrode, the larger space intervals also made larger time intervals possible there, used that as well. Klymenko et al. [30] combined equally divided steps, followed by exponentially expanding time steps in a simulation of double potential step chronoamperometry.

Another reason for starting with a small interval and expanding from there is the use of simulation methods that have an oscillatory response, such as Crank–Nicolson, alternating directions implicit (ADI) and DuFort/Frankel (see Chaps 8 and 9). These methods are less oscillatory for small time intervals. There are several choices. For Crank–Nicolson, one might divide the first time step into subintervals, either equal (the Pearson method [31]), or expanding subintervals; both have been investigated [32, 33]. If choosing exponentially increasing time intervals over some period $\tau$, which may be the total simulation period, a pulse duration or (see below) a single whole time interval to be subdivided, one divides the period into $M$ intervals of length $\delta t_k, k = 1, \ldots, M$. Assume the recursive relation

$$\delta t_k = \gamma \; \delta t_{k-1} \tag{7.24}$$

and

$$\sum_{k=1}^{M} \delta t_k = \tau \tag{7.25}$$

(note that these equations are of the same form as (7.16) and (7.17)). Peaceman and Rachford [34] used the technique for the first time in 1955, in their classical paper describing the ADI method (Chap. 12) and this has been studied in an electrochemcial context [35–37]. As will be mentioned in Chap. 9, ADI, a much used method for two-dimensional simulations, can produce oscillations and unequal time intervals are routinely used [38–47], naming only publications of the last 10 years. None of these mentions oscillations. In [39] we see the remark "in order to maximise the efficiency.." on expanding time intervals, but in most works there is no explanation given. Some have used many small equal intervals, which can also damp oscillations, also without explanation [48–50].

There is no problem with varying time intervals with two-level simulation methods, but with a method like BDF, there is the problem that one needs multipoint time derivatives calculated from unequally spaced points in time. Feldberg and Goldstein [36] show how to do this and even show how to apply the Feldbergian correction of half a time interval in this case, that becomes necessary when using the simple start for BDF, described in Chap. 4 (see also the consistency proof for this procedure in Appendix C).

### 7.4.1 *Implementation of Exponentially Increasing Time Intervals*

A special case of exponentially increasing intervals, applied only to the subdivision of the first time interval with interval doubling, that is, the case $\gamma = 2$, was applied by Britz and Østerby [51]. Mocak et al. [52] suggested "gradually increasing $D^*$" (meaning time intervals). In [51], the sequence for $M$ such steps was the sequence of fractions $2^{-M+1}, 2^{-M+1}, 2^{-M+2}, 2^{-M+3}, \ldots, \frac{1}{2}$. Note that the smallest fraction is applied twice. The general formula using (7.24) and (7.25) is implemented in a different way. This was done in a recent paper [32], using subdivision of the first time interval, in order to damp the oscillations often produced by the Crank–Nicolson method (Chap. 8). The form of the equations for the required parameters ($M$, $\gamma$, size of the first subinterval) is exactly like that for exponentially increasing spatial intervals, Eqs. (7.16)–(7.19).

## 7.5 Adaptive Interval Changes

The most flexible strategy is to adapt intervals, in space or in time, according to need at any particular time during the simulation. Ablow and Schechter refer to *campylotropic* or *curvature-seeking* coordinates [53]. It was noted above that fixed unequal spatial intervals might not be suitable if, for example, a reaction layer becomes too thin even for the first few intervals to lie within it. Worse still, the method described above, in which points are most closely spaced near the electrode, cannot accommodate sharp changes in concentration changes that occur away from the electrode, as can indeed happen. Bieniasz [54] described, in 1994, a system involving a second-order homogeneous chemical reaction, in which a sharp concentration peak appears in the solution for one of the species. Only adaptive techniques can handle this situation. This was rediscovered in 2010 [55, 56], as pointed out in [57]. We distinguish between adaptation of spatial and temporal intervals. There is a vast numerical literature on this topic, and just some selected citations are given here. Bieniasz wrote a series of articles in which he introduced the idea to electrochemical simulations. In [9, 58], he described the use of a fixed number of grid points, moved about as required. He then [54] applied this to a concentration hump as mentioned above. Bieniasz later turned to a different technique [59], which he called "patch-adaptive", that starts with a coarse but evenly spaced grid, to which new points are added (and perhaps removed again later) midway between existing points, as required. He also applied time-step adaptation [60]. Nann and Heinze [61, 62] meanwhile developed a finite element method in which points (nodes) are added where needed, an idea carried forward, refined and applied to two-dimensional systems by Harriman et al. [63–67]. Ludwig and Speiser [68–70] applied an adaptive finite element algorithm to spatial integration, controlled by a hierarchical *a posteriori* error estimator. Time step adaptation, a standard in the literature on *ode*s (see such texts as [71, 72]), was first applied by Bieniasz [60] to electrochemical simulation (see below). The references cited here include citations of the important works within the larger numerical literature.

### *7.5.1 Spatial Interval Adaptation*

The single reference to Thompson's survey [73] must suffice to represent the numerical literature, and the references in the papers of Bieniasz [9, 54, 58, 59] provide further background.

Bieniasz began his series with an exploration of moving grids [9, 58], using a fixed number of points. As a given simulation develops, the program determines whether the spacing needs to be closer or wider across the concentration profile, in a preliminary forward step, and then adjusts the point positions. This is called regridding. The criterion for moving the points is a sensitive issue, on which there is some disagreement in the literature. The essence of all schemes is to produce a so-called monitor function [74] or a function based on it, that in some way resembles the simulated variable's profile, and then to slice this into equal vertical intervals, producing new points along $X$ which then, hopefully, place points where they are most needed. Dorfi et al. [75] suggest using a monitor function such as

$$M(i) = \sqrt{\alpha + (du/dx)^2} \tag{7.26}$$

at every point $i$, where $u$ is the variable to be computed. The value of $\alpha$ is given as unity in older papers such as that of Blom et al. [74], but Bieniasz found [9, 58] that a smaller value like 0.0005 is better in the present context. This monitor function is now integrated with respect to $x$ to produce the monitor profile, generally given the symbol $\xi$:

$$\xi(x) = \frac{\int_0^x M(x)\,\mathrm{d}x}{\int_0^{x_{max}} M(x)\,\mathrm{d}x} \tag{7.27}$$

(note that it is normalised by its value at the outer limit for $x$, $x_{max}$, so that it rises to unity). An algorithm is then applied to it to slice it into equal vertical intervals and to find the $x$-positions that correspond to them. The function (7.26) is also referred to elsewhere [73, 76], to name just a few references. Blom [74], on the other hand, recommends the use of the second derivative,

$$M(i) = \sqrt{\alpha + |\mathrm{d}^2u/\mathrm{d}x^2|)} \tag{7.28}$$

and Bieniasz follows this suggestion [9, 58]. The reason is (Bieniasz, 2001, private communication) that the first derivative is not itself of great significance, if the second derivative is small in the diffusion equation, so the second derivative indicates places in the profile where things are changing.

The procedure is then as follows. At a given time, a trial step is taken to the next time level. This produces a provisional new concentration profile. From this, the $\xi$-function (7.27) is generated and from it, a new set of positions for the points. Now the concentrations are interpolated at these points, between the present concentration points, and the step to the next time level repeated on the new set of points.

Let us provide an example. We take a uniform grid of just 20 points in the range $0 \le X \le 6$ and assume a Cottrell concentration profile at time $T = 0.5$ shown in Fig. 7.3. This is a little artificial, as one would never carry out such a drastic regridding, but it will illustrate the method better than what usually happens (small changes over a given time interval). It amounts to taking a huge step of 0.5 in $T$ and somehow having obtained a rather accurate new concentration profile. Against the advice of Blom [74] and Bieniasz [9, 58] (see below), we compute second derivatives of the profile at all the node points, using the usual central three-point formula, except at the electrode, where an asymmetric three-point formula is used (see Sect. 3.8, page 51 and Appendix A). These are all first-order accurate if the

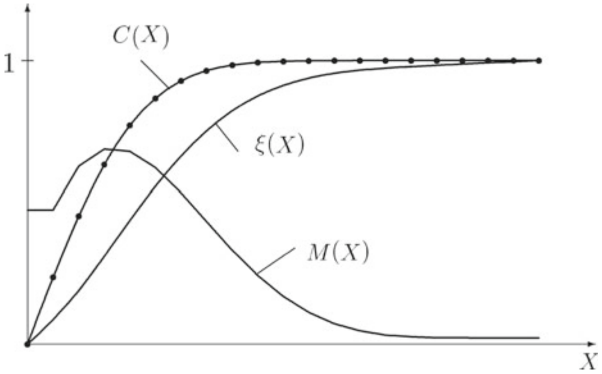


**Fig. 7.3** Some profiles derived from a Cottrell profile at $T = 0.5$

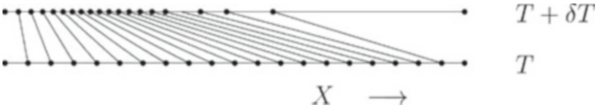


**Fig. 7.4** Regridding for the Cottrell profile in Fig. 7.3

intervals along $X$ are not equal and we obtain the function $M(X)$ in Fig. 7.3, plotted point to point. It is integrated to $\xi(X)$ using the trapezium method. Normalisation is to its final value, at $X_{max} = 6$. $\xi(X)$, ranging from zero to unity, is now inverted to $X(\xi)$ and 20 $X$-values found for it at equal intervals in $\xi$ of 0.05 by interpolation. Blom et al. [74] and Bieniasz [9, 58] cite a paper by de Boor [77] for this process, but it is in fact not very complicated to implement using a standard interpolation routine. Now, if one were to go on, as one normally would, a new concentration profile at the new set of positions along $X$ at $T$ is computed, again by interpolation. The shift in positions is indicated in Fig. 7.4 and it is seen that there is now a wide spacing at the far end, and a crowding of points, not near the electrode but some distance away from it, where the monitor function is maximum. That is also roughly where the greatest changes in concentration occur during the next time step; this supports the argument in favour of the second derivative in the monitor function. Note that the 20 points here are a rather small number, chosen to make the figure clearer. Bieniasz normally uses about 50, so that the possibly excessive spacing at the far end would not be so wide. Also, to some extent the wide spacing is a result of the large step in time taken in the example. A second regridding on the new grid shown in these figures does in fact lead to a smaller gap at large $X$ and this is what one would obtain if a number of smaller steps had been taken.

Some remarks are in order, starting with the purpose of the $\alpha$ term in (7.28). As mentioned, the numerical literature appears to prefer it to be close to unity. If one were to set it to zero, one would obtain an unacceptably wide spacing at parts of the profile where the second derivative is close to zero. In effect, a finite $\alpha$ value ensures a finite positive gradient of $\xi(X)$ at large $X$. If this is not done, the plateau obtained means excessively large intervals in this region upon regridding. As mentioned, in the work of Bieniasz, a value of 0.0005 was found optimal. Secondly, there is the question of how to compute the second derivatives over unevenly spaced points. Blom et al. [74], followed by Bieniasz [58], used a somewhat awkward method. The first point monitored (using their method) lies at the middle of the first interval, and the formula given by Blom et al. [74] is in fact incorrect. Presumably, the second derivative at $X = 0$ is assumed zero, which it need not be. In our example, it was computed as the one-sided three-point approximation at $X = 0$, which seems to make more sense. From there on to the far edge, Blom et al. use four-point expressions, centred on the middle of the mid-interval; however, their expression again is incorrect, rendering the use of four points useless. The object was to achieve a better approximation to the derivative, that might be second-order in the interval lengths in some sense. The expression was later corrected by Bieniasz [9]. Both teams then use the simple three-point approximations for the final calculation, presumably in order to avoid yet further interpolations. It appears that one might as well use three-point formulas in both phases, as was done in the example above. Alternatively, one might use higher-order formulas, especially for the diffusion step, on the unequal grid, using more than three points, for example five, centered on existing points. This has not been attempted to date. Such formulae are provided in Chap. 3; a few cases are also given in Appendix A, and a general procedure for them is described in Appendix E.

Despite the fact that adaptive gridding seems to work very well (with some refinements described in [9]), being for example, until recently, the only method capable of adapting to a narrow concentration hump away from the electrode [54], Bieniasz has recently concluded [59, 78–80] that another method is better. The problems he noted are, among others, the need to set $\alpha$ to some value, and the problems arising from the approximation to the second derivative on an unequally spaced grid. The new method is called patch-adaptive, and works with a continually varying number of points. It is based on older work in the numerical literature (see [59] for a large number of citations). One begins with a coarse evenly spaced grid, and does a calculation to the next time level. This is then repeated on a grid of twice as many points, the new points placed exactly midway between the first set. This ensures a locally equal spacing and thus second-order second derivatives. The two solutions are then used to provide an error estimate. The way this is done depends on the simulation algorithm. One way might be to use extrapolation, described in Chap. 9, which can provide an error estimate. At those places along $X$ where this error exceeds some set value in magnitude, new points are then placed midway between the existing points, and the calculation repeated. If there appear sharp gradients in the profile, more and more points will thus be inserted. All the time, however, one is working (locally) with equally spaced points and thus second-order second derivatives. The disadvantage is that one must keep track of a changing number of points along $X$, as points are added and perhaps removed later. This requires data structures that are not trivial to program. It seems to the present authors that this renders the method less interesting to the programming electrochemist. It might be of more interest to programmers of general simulation packages.

### *7.5.2 Time Interval Adaptation*

Just as sharp changes in the space direction point to changes in spatial intervals, so sharp changes with time demand time interval adaptation. This is in fact standard procedure in the *ode* world since the paper by Douglas [81], see, for example, such standard works as [71, 72]. In electrochemical simulations, there have been relatively few attempts to do this. The impetus for varying time intervals comes from two problems. One problem is that of pulse techniques, especially current reversal or potential double pulses. Clearly, there are sharp changes in concentration profiles at the onset of each pulse. Crude beginnings of this [27–29], using alternately larger and smaller time intervals before and after a pulse, have been mentioned above.

Once again, an adaptive technique might be the universal answer to these problems, especially since there might be unforeseen changes at various stages during a simulation, as can happen in linear sweep voltammetry. Such a scheme has been devised by Bieniasz [60]. Upon first considering this, one might assume, say, that current changes themselves could be the factor that decides the length of the next time interval to be used. However, as with adaptive spatial intervals, this is not as good as using a kind of second derivative, for similar reasons. If there were changes in concentrations linear in time, then no matter how large these changes are, large time intervals can be used; but if the changes are themselves changing (that is, there are significant second derivatives with time), the intervals must be reduced. The picture is complicated by the fact that this will mostly be used in conjunction with adaptive spatial grids, making the second derivatives less straightforward to express. Bieniasz suggests the use of the following quantity as a kind of monitor function. Assume that a tentative step of $\delta T$ has been taken on the present grid, and that a given point indexed $i$ along $X$ has just been moved by an amount $\delta H$; the estimate function $EST$ is then

$$EST = \frac{\delta T^2}{2}\frac{\partial^2 C}{\partial T^2} + \delta T \delta H \frac{\partial^2 C}{\partial X \partial T} + \frac{\delta H^2}{2}\frac{\partial^2 C}{\partial X^2} \tag{7.29}$$

where the second derivatives must be discretised by some suitable expression. The present authors regard this as more complicated than the average electrochemist is willing to program, and the method is left to programmers of general simulation packages and will not be detailed any further here. It did produce impressive results [60] with square wave simulations.

There are some simpler strategies that might do, and are easier to program. If an experiment such as double pulse or square wave voltammetry is simulated, the sharp changes occur at predictable times, and simple sequences of time intervals, such as exponentially expanding intervals, can be satisfactory, repeating the sequence at the onset of each pulse.

If there are unpredictable changes, the answer might be to use a professional package; that is, either a simulation package (see Chap. 17), or the method of lines (Chap. 9) and a professional routine for solving the resulting set of *ode*s, making use of the adaptive time intervals feature, which these routines normally offer.

## References

1. Noye J (1982) Finite difference methods for partial differential equations. In: Noye J (ed) Proceedings of the 1981 conference on the numerical solution of partial differential equations, Queen's College, Melbourne, Australia. North Holland, Amsterdam, pp 3–137
2. Hunter IC, Jones IP (1981) Numerical experiments on the effects of strong grid stretching in finite difference calculations. Technical Report AERE R-10301, United Kingdom Atomic Energy Authority, Harwell
3. Crowder HJ, Dalton C (1971) Errors in the use of nonuniform mesh systems. J Comput Phys 7:32–45
4. Kálnay de Rivas E (1972) On the use of nonuniform grids in finite-difference equations. J Comput Phys 10:202–210
5. Rudolph M (2002) Digital simulation on unequally spaced grids. Part 1. Critical remarks on using the point method by discretisation on a transformed grid. J Electroanal Chem 529:97–108
6. Joslin T, Pletcher D (1974) The digital simulation of electrode processes. Procedures for conserving computer time. J Electroanal Chem 49:171–186
7. Seeber R, Stefani S (1981) Explicit finite difference method in simulating electrode processes. Anal Chem 53:1011–1016
8. Feldberg SW (1981) Optimization of explicit finite-difference simulation of electrochemical phenomena utilizing an exponentially expanded space grid. Refinement of the Joslin-Pletcher algorithm. J Electroanal Chem 127:1–10
9. Bieniasz LK (1994) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 2. An improved finite-difference adaptive moving grid technique for fast homogeneous reaction-diffusion problems with reaction layers at the electrodes. J Electroanal Chem 374:1–22
10. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
11. Rudolph M (2003) Digital simulations on unequally spaced grids. Part 2. Using the box method by discretisation on a transformed equally spaced grid. J Electroanal Chem 543:23–39
12. Rudolph M (2003) Reply to L.K. Bieniasz's comments on my paper [J Electroanal Chem 529:97 (2002)]. J Electroanal Chem 558:171–176
13. Bieniasz LK (2003) Comments on the paper by M. Rudolph, entitled "Digital simulations on unequally spaced grids. Part 1. Critical remarks on using the point method by discretisation on a transformed grid" [J Electroanal Chem 529:97 (2002) ]. J Electroanal Chem 558:167–170
14. Cheney W, Kincaid D (1985) Numerical mathematics and computing. Brooks/Cole, Belmont, CA
15. Gerald CF (1978) Applied numerical analysis, 2nd edn. Addison–Wesley, Reading, MA
16. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical recipes in fortran. The art of scientific computing, 2nd edn. Cambridge University Press, Cambridge
17. Pao YH, Daugherty RJ (1969) Time-dependent viscous incompressible flow past a finite flat plate. Technical Report Rept. DI-82-0822, Boeing Sci. Res. Lab.
18. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376
19. Martínez-Ortiz F, Zoroa N, Laborda E, Molina A (2016) Brute force (or not so brute) digital simulation in electrochemistry revisited. Chem Phys Lett 643:71–76. Supplementary material in the form of C++ programs
20. Sundqvist H, Veronis G (1970) A simple finite-difference grid with non-constant intervals. Tellus 22:26–31
21. Saul'yev VK (1964) Integration of equations of parabolic type by the method of nets. Pergamon Press, New York
22. Martínez-Ortiz F, Zoroa N, Molina Á, Serna C, Laborda E (2009) Electrochemical digital simulations with an exponentially expanding grid: general expressions for higher order approximations to spatial derivatives. The special case of four-point formulas and their

application to multipulse techniques in planar and any size spherical electrodes. Electrochim Acta 54:1042–1055

23. Britz D, Strutwolf J (2014) Several ways to simulate time dependent liquid junction potentials by finite differences. Electrochim Acta 137:328–335
24. Britz D, Strutwolf J (2015) Digital simulation of chronoamperometry at a disk electrode under a flat polymer film containing an enzyme. Electrochim Acta 152:302–307
25. Fornberg B (1988) Generation of finite difference formulas on arbitrarily spaced grids. Math Comput 51:699–706
26. Rudolph M, Reddy DP, Feldberg SW (1994) A simulator for cyclic voltammetry responses. Anal Chem 66:589A–600A
27. Flanagan JB, Takahashi K, Anson FC (1977) Reactant adsorption in differential pulse polarography. Effects of adsorptive depletion of reactant, nonlinear adsorption isotherms and uncompensated resistance. J Electroanal Chem 81:261–273
28. Dillard JW, Turner JA, Osteryoung RA (1977) Digital simulation of differential pulse polarography with incremental time change. Anal Chem 49:1246–1250
29. Nikolić S (1983) Digitalna simulacija elektrodnih reakcija za pulsnu polarografiju i srodne tehnike. Master's thesis, Zagreb University
30. Klymenko OV, Evans RG, Hardacre C, Svir IB, Compton RG (2004) Double potential step chronoamperometry at microdisk electrodes: simulating the case of unequal diffusion coefficients. J Electroanal Chem 571:211–221
31. Pearson CE (1965) Impulsive end condition for diffusion equation. Math Comput 19:570–576
32. Britz D, Østerby O, Strutwolf J (2003) Damping of Crank-Nicolson error oscillations. Comput Biol Chem 27:253–263
33. Østerby O (2003) Five ways of reducing the Crank-Nicolson oscillations. BIT Numer Math 43:811–822
34. Peaceman DW, Rachford HH (1955) The numerical solution of parabolic and elliptic differential equations. J Soc Ind Appl Math 3:28–41
35. Britz D, Oldham KB, Østerby O (2009) Strategies for damping the oscillations of the alternating direction implicit method of simulation of diffusion-limited chronoamperometry at disk electrodes. Electrochimica Acta 54:4822–4828
36. Feldberg SW, Goldstein CI (1995) Examination of the behavior of the fully implicit finite-difference algorithm with the Richtmyer modification: behavior with an exponentially expanding time grid. J Electroanal Chem 397:1–10
37. Lavagnini I, Pastore P, Magno F, Amatore CA (1991) Performance of a numerical method based on the hopscotch algorithm and on an oblate spheroidal space coordinate- expanding time grid for simulation of voltammetric curves at an inlaid disk microelectrode. J Electroanal Chem 316:37–47
38. Amatore C, Oleinick A, Svir I (2005) Diffusion within nanometric and micrometric spherical-type domains by nanometric ring or pore active interfaces. Part 1: conformal mapping approach. J Electroanal Chem 575:103–123
39. Barnes AS, Streeter I, Compton RG (2008) On the use of digital staircase ramps for linear sweep voltammetry at microdisc electrodes: large step potentials significantly broaden and shift voltammetric peaks. J Electroanal Chem 623:129–133
40. Barnes EO, Lewis GEM, Dale SEC, Marken F, Compton RG (2013) Dual band electrodes in generator-collector mode: Simultaneous measurement of two species. J Electroanal Chem 703:38–44
41. Belding SR, Baron R, Dickinson EJF, Compton RG (2009) Modeling diffusion effects for a stepwise two-electron reduction process at a microelectrode: study of the reduction of para-quaterphenyl in tetrahydrofuran and inference of fast comproportionation of the dianion with the neutral parent molecule. J Phys Chem C 113:16042–16050
42. Eloul S, Compton RG (2014) Voltammetric sensitivity enhancement by using preconcentration adjacent to the electrode: simulation, critical evaluation, and insights. J Phys Chem C 118:24520–24532

43. Eloul S, Compton RG (2014) Shielding of a microdisc electrode surrounded by an adsorbing surface. Chem Electrochem 1:917–924
44. Klymenko OV, Oleinick AI, Amatore C, Svir I (2007) Reconstruction of hydrodynamic flow profiles in a rectangular channel using electrochemical methods of analysis. Electrochim Acta 53:1100–1106
45. Rogers EI, Huang X, Dickinson EJF, Hardacre C, Compton RG (2009) Investigating the mechanism and electrode kinetics of the oxygen/superoxide ($O_2|O_2^{\cdot -}$) couple in various room-temperature ionic liquids at gold and platinum electrodes in the temperature range 298–318 K. J Phys Chem C 113:17811–17823
46. Streeter I, Compton RG (2007) Linear sweep voltammetry at randomly distributed arrays of microband electrodes. J Phys Chem C 111:15053–15058
47. Svir I, Oleinick A, Yunus K, Fisher AC, Wadhawan JD, Davies TJ, Compton RG (2005) Theoretical and experimental study of the ECE mechanism at microring electrodes. J Electroanal Chem 578:289–299
48. Molina A, Gonzalez J, Barnes EO, Compton RG (2014) Simple analytical equations for the current-potential curves at microelectrodes: a universal approach. J Phys Chem C 118:346–356
49. Molina A, Olmos J, Laborda E (2015) Reverse pulse voltammetry at spherical and disc microelectrodes: characterization of homogeneous chemical equilibria and their impact on the species diffusivities. Electrochim Acta 169:300–309
50. Ngamchuea K, Eloul S, Tschulik K, Compton RG (2014) Planar diffusion to macro disc electrodes - what electrode size is required for the Cottrell and Randles-Sevcik equations to apply quantitatively? J Solid State Electrochem 18:3251–3257
51. Britz D, Østerby O (1994) Some numerical investigations of the stability of electrochemical digital simulation, particularly as affected by first-order homogeneous reactions. J Electroanal Chem 368:143–147
52. Mocak J, Feldberg SW (1994) The Richtmyer modification of the fully implicit finite difference algorithm for simulations of electrochemical problems. J Electroanal Chem 378:31–37
53. Ablow CM, Schechter S (1978) Campylotropic coordinates. J Comput Phys 27:351–362
54. Bieniasz LK (1994) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 4. The adaptive moving-grid solution of one-dimensional fast homogeneous reaction-diffusion problems with extremely thin reaction zones away from the electrodes. J Electroanal Chem 379:71–87
55. Amatore C, Klymenko O, Svir I (2010) A new strategy for simulation of electrochemical mechanisms involving acute reaction fronts in solution: application to model mechanisms. Electrochem Commun 12:1165–1169
56. Amatore C, Klymenko O, Svir I (2010) A new strategy for simulation of electrochemical mechanisms involving acute reaction fronts in solution: principle. Electrochem Commun 12:1170–1173
57. Britz D (2011) The true history of adaptive grids in electrochemical simulation. Electrochim Acta 56:4420–4421
58. Bieniasz LK (1993) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 1. Introductory exploration of the finite-difference adaptive moving grid solution of the one-dimensional fast homogeneous reaction-diffusion problem with a reaction layer. J Electroanal Chem 360:119–138
59. Bieniasz LK (2000) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 5. A finite-difference adaptive space/time strategy based on a patch-type local uniform grid refinement, for kinetic models in one-dimensional space geometry. J Electroanal Chem 481:115–133. Corrigendum: ibid. 565:131 (2004)
60. Bieniasz LK (1994) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 3. An adaptive moving grid-adaptive time step strategy for problems with discontinuous boundary conditions at the electrodes. J Electroanal Chem 374:23–35

61. Nann T (1997) Digitale Simulation in der Elektrochemie mit der Methode der Finiten Elementen. Ph.D. thesis, Albert-Ludwigs-Universität zu Freiburg im Breisgau. Publ. by Shaker Verlag, Aachen
62. Nann T, Heinze J (1999) Simulation in electrochemistry using the finite element method. Part 1. The algorithm. Electrochem Commun 1:289–294
63. Harriman K, Gavaghan DJ, Houston P, Kay D, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. *ECE* and $EC_2E$ mechanisms at channel microband electrodes. Electrochem Commun 2:576–585
64. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. An E reaction at a channel microband electrode. Electrochem Commun 2:567–575
65. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Application to a simple model problem. Electrochem Commun 2:150–156
66. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. First-order EC' mechanism at inlaid and recessed discs. Electrochem Commun 2:163–170
67. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Theory. Electrochem Commun 2:157–162
68. Ludwig K, Speiser B (2006) EChem++ - an object-oriented problem solving environment for electrochemistry: part 4. Adaptive multilevel finite elements applied to electrochemical models. Algorithm and benchmark calculations. J Electroanal Chem 588:74–87
69. Ludwig K, Speiser B (2007) EChem++ - An object-oriented problem solving environment for electrochemistry. Part 5. A differential-algebraic approach to the error control of adaptive algorithms. J Electroanal Chem 608:91–101
70. Ludwig K, Morales I, Speiser B (2007) EChem++ - An object-oriented problem solving environment for electrochemistry. Part 6. Adaptive finite element simulations of controlled-current electrochemical experiments. J Electroanal Chem 608:102–110
71. Brenan KE, Campbell SL, Petzold LR (1996) Numerical solution of initial-value problems in differential-algebraic equations. SIAM, Philadelphia
72. Hairer E, Nørsett SP, Wanner G (1987) Solving ordinary differential equations I. Nonstiff problems. Springer, Berlin
73. Thompson JF (1985) A survey of dynamically-adaptive grids in the numerical solution of partial differential equations. Appl Numer Math 1:3–27
74. Blom JG, Sanz-Serna JM, Verwer JG (1988) On simple moving grid methods for one-dimensional evolutionary partial differential equations. J Comput Phys 74:191–213
75. Dorfi EA, Drury LO (1987) Simple adaptive grids for 1-D initial value problems. J Comput Phys 69:175–195
76. Sanz-Serna JM, Christie I (1986) A simple adaptive technique for nonlinear wave problems. J Comput Phys 67:348–360
77. de Boor C (1974) Good approximation by splines with variable knots. II. In Watson GA (ed) Conference on the numerical solution of differential equations, Dundee, Scotland, 1973. Springer, Berlin, pp 12–20
78. Bieniasz LK (2000) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 6. Testing of the finite-difference patch-adaptive strategy on example models with solution difficulties at the electrodes, in one-dimensional space geometry. J Electroanal Chem 481:134–151. Corrigendum: ibid. 565:133 (2004)
79. Bieniasz LK (2001) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Patch-adaptive simulation of moving fronts in non-linear diffusion models of the switching of conductive polymers. Electrochem Commun 3:149–153

80. Bieniasz LK, Bureau C (2000) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 7. Testing of the finite-difference patch-adaptive strategy on example models with moving reaction fronts, in one-dimensional space geometry. J Electroanal Chem 481:152–167. Corrigendum: ibid. 565:135 (2004)
81. Douglas J Jr, Gallie TM Jr (1955) Variable time steps in the solution of the heat flow equation by a difference equation. Proc Am Math Soc 6:787–793

# Chapter 8
# The Commonly Used Implicit Methods

Essentially, only two implicit methods will be described here, but with extensions that make them more useful. They are derived from the implicit methods described for *ode*s in Chap. 4, BI and the trapezium method. These have different names in the *pde* context, as will be seen.

Implicit methods have the great advantage of being stable for any $\lambda = \delta T/H^2$, in contrast with the explicit method. It will be seen (and analysed in detail in Chap. 15) that the Laasonen method, a kind of BI, is very stable and responds to sharp transients with smoothly declining (but relatively large) errors, whereas Crank–Nicolson, also stable, responds with error oscillations of declining amplitude, but is more accurate. The drawbacks of both methods can be overcome, as will be described below.

First, the discretisation of the second, spatial derivative of concentration will be reiterated in a general form that can then be built into the methods to follow. For the three concentrations grouped around the point $X_i$, we can write the general linear expression,

$$\frac{\partial^2 C_i}{\partial X^2} \approx \alpha_1 C_{i-1} + \alpha_2 C_i + \alpha_3 C_{i+1} \tag{8.1}$$

in which the $\alpha$ coefficients are defined according to whether equal or unequal intervals are used. The three concentrations are situated at the three corresponding positions $X_{i-1}$, $X_i$ and $X_{i+1}$. For equal intervals $H$ in $X$, the coefficients are

$$\begin{aligned} \alpha_1 &= 1/H^2 \\ \alpha_2 &= -2/H^2 \\ \alpha_3 &= 1/H^2 \end{aligned} \tag{8.2}$$

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_8

as already given in Chap. 3, Eq. (3.41), and they are independent of the index $i$. If unequal intervals are used, the coefficients are

$$\begin{aligned}\alpha_1 &= \frac{2}{(X_i - X_{i-1})(X_{i+1} - X_{i-1})}\\ \alpha_2 &= -\frac{2}{(X_i - X_{i-1})(X_{i+1} - X_i)}\\ \alpha_3 &= \frac{2}{(X_{i+1} - X_i)(X_{i+1} - X_{i-1})}\end{aligned} \tag{8.3}$$

(as also seen, in a different but equivalent form, in Appendix A, Table A.4 on page 443) or they can be computed using the Fornberg subroutine `FORN` seen in Appendix E. These $\alpha$'s are dependent on the index $i$ but for brevity, this will not always be indicated in what follows below.

If transformation is to be used, by the function (7.3) on page 125, then the resulting right-hand-side of the diffusion equation (7.5) can be written as a transformation of the second spatial derivative,

$$\frac{\partial^2 C}{\partial X^2} = a^2 e^{-2Y}\left(\frac{\partial^2 C}{\partial Y^2} - \frac{\partial C}{\partial Y}\right) \tag{8.4}$$

and the (obvious) discretisation in terms of equal intervals $\delta Y$ then results again in a linear expression like (8.1), with the coefficients at the point $Y_i$ given by

$$\begin{aligned}\alpha_1 &= \left(1 + \frac{\delta Y}{2}\right)\, w_i\\ \alpha_2 &= -2\, w_i\\ \alpha_3 &= \left(1 - \frac{\delta Y}{2}\right)\, w_i\end{aligned} \tag{8.5}$$

the common $w_i$ being

$$w_i = \frac{a^2 \exp(-2i\delta Y)}{\delta Y^2}\,. \tag{8.6}$$

We thus have, for the diffusion equation

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2}\,, \tag{8.7}$$

a suitable discretisation for all three cases with their respective definitions of the $\alpha$ coefficients, in the form

$$\frac{\partial C}{\partial T} = \alpha_1 C_{i-1} + \alpha_2 C_i + \alpha_3 C_{i+1} \ . \tag{8.8}$$

More points might of course be chosen for the approximation, which extends the number of terms on the right-hand side, but for brevity we keep to the three-point approximation here. In Sect. 8.4 however, a model using four-point approximations is presented.

The diffusion equation, discretised on the right-hand side as in (8.8), is now a system of *ode*s in the concentration vector **C**, of the form

$$\frac{\partial \mathbf{C}}{\partial T} = f(\mathbf{C}) \tag{8.9}$$

and the two main implicit methods will be seen to be analogous to those used for *ode*s.

## 8.1 The Laasonen Method or BI

When applied to the solution of *ode*s, the BI method (Chap. 4) uses a backward difference for the derivative on the left-hand side of (8.9) and the argument of the function on the right-hand side is the future, unknown, concentration vector. In our notation, at the point $i$ along the row of concentrations, this is

$$\frac{C'_i - C_i}{\delta T} = \alpha_1 C'_{i-1} + \alpha_2 C'_i + \alpha_3 C'_{i+1} \ . \tag{8.10}$$

This was formulated by Laasonen [1] in 1949. The equation set becomes

$$\begin{aligned}
&C'_0 &&+ a_{1,1}C'_1 &&+ a_2 C'_2 &&= b_1 \\
&C'_1 &&+ a_{1,2}C'_2 &&+ a_2 C'_3 &&= b_2 \\
&\dots \\
&C'_{i-1} &&+ a_{1,i}C'_i &&+ a_2 C'_{i+1} &&= b_i \\
&\dots \\
&C'_{N-1} &&+ a_{1,N}C'_N &&+ a_2 C'_{N+1} &&= b_N
\end{aligned} \tag{8.11}$$

with the coefficients given by

$$\begin{aligned} a_{1,i} &= \frac{\alpha_2 - 1/\delta T}{\alpha_1} \\ a_2 &= \frac{\alpha_3}{\alpha_1} \\ b_i &= \frac{-1}{\delta T \alpha_1} C_i \end{aligned} \tag{8.12}$$

in which the coefficient $a_2$ has been written as independent of $i$. In most cases, $a_2$ will in fact be constant, and equal to unity for equal intervals in $X$. For exponentially expanding intervals as in (7.19), $\alpha_3/\alpha_2$ is equal to $1/\gamma$, that is, the inverse of the interval expansion factor. It is however good practice to assume all coefficients as $i$-dependent, making extension to more points for the approximations easier.

The solution of the above system of equations (8.11) will be described below, together with that for the CN method.

## 8.2 The Crank–Nicolson Method, CN

This method derives from the trapezium method in the *ode* field in which the time derivative in (8.9), expressed exactly as in (8.10), becomes a second-order central difference by virtue of the fact that the right-hand side now refers to a point in time midway in the time interval. This is achieved by taking the average of the second spatial derivative at the present time $T$ and that at $T + \delta T$:

$$\frac{C'_i - C_i}{\delta T} = \frac{1}{2}\left(\alpha_1 C'_{i-1} + \alpha_2 C'_i + \alpha_3 C'_{i+1} + \alpha_1 C_{i-1} + \alpha_2 C_i + \alpha_3 C_{i+1}\right) \ . \tag{8.13}$$

The result is a system exactly as (8.11) but with different definitions of the coefficients:

$$\begin{aligned} a_{1,i} &= \frac{\alpha_2 - 2/\delta T}{\alpha_1} \\ a_2 &= \frac{\alpha_3}{\alpha_1} \\ b_i &= -C_{i-1} - a_{3,i} C_i - a_2 C_{i+1} \end{aligned} \tag{8.14}$$

and the new coefficient

$$a_{3,i} = \frac{\alpha_2 + 2/\delta T}{\alpha_1} \ . \tag{8.15}$$

Again, $a_1$ and $a_3$ are dependent on the index $i$, by virtue of the fact that the $\alpha$'s are $i$-dependent.

Crank–Nicolson bears the name of its inventors [2]. It is interesting to note that in their paper, they cite Hartree and Womersley [3], who describe what amounts to its precursor.

## 8.3 Solving the Implicit System

The system (8.11) shown above, that is the result of discretising either according to the Laasonen or CN method, can be solved efficiently by the Thomas algorithm [4, 5]. This recognises that the system is tridiagonal. It can be reduced to a didiagonal system by working from either end, that is, from $C'_0$ or $C'_N$. The latter approach is better here. The last equation in (8.11) has a term in $C'_{N+1}$, which is the bulk value, not subject to diffusional changes, being a boundary value. Normally, it is constant, equal to the initial bulk value. In some cases, it can change with time, for example in the Reinert–Berg [6] or the Birk and Perone [7] systems, in which the reacting substance itself undergoes a homogeneous decay reaction. In such cases, the value of $C'_{N+1}$, while not constant, is still accurately predictable and thus known at any time. Thus, this known term can be moved to the right-hand side of the last equation of the system, thus giving the new last equation

$$C'_{N-1} + a_{1,N}C'_N = b_N - a_2C'_{N+1} \tag{8.16}$$

with only two unknowns. This is rewritten in the form

$$C'_{N-1} + a'_N C'_N = b'_N \tag{8.17}$$

with, clearly,

$$a'_N = a_{1,N} \tag{8.18}$$

and

$$b'_N = b_N - a_2C'_{N+1} \ . \tag{8.19}$$

Equation (8.17) is now used to express $C'_N$ in terms of $C'_{N-1}$:

$$C'_N = \frac{b'_N - C'_{N-1}}{a'_N} \tag{8.20}$$

and this is substituted into the second-last equation of the system (8.11),

$$C'_{N-2} + a_{1,N-1}C'_{N-1} + a_2C'_N = b_{N-1} \tag{8.21}$$

giving, after some tidying up, the next new equation

$$C'_{N-2} + a'_{N-1}C'_{N-1} = b'_{N-1} \tag{8.22}$$

with

$$a'_{N-1} = a_{1,N-1} - \frac{a_2}{a'_N} \tag{8.23}$$

and

$$b'_{N-1} = b_{N-1} - a_2\frac{b'_N}{a'_N} \; . \tag{8.24}$$

This process continues in the backward direction and the recursive expressions for the coefficients in the *i*th equation generated,

$$C'_{i-1} + a'_iC'_i = b'_i \tag{8.25}$$

are

$$a'_i = a_{1,i} - \frac{a_2}{a'_{i+1}} \tag{8.26}$$

and

$$b'_i = b_i - a_2\frac{b'_{i+1}}{a'_{i+1}} \tag{8.27}$$

[starting with (8.18) and (8.19)] until the first equation is reached,

$$C'_0 + a'_1C'_1 = b'_1 \; . \tag{8.28}$$

At this point, we have a new system of equations, each with two unknowns. The point of attack now is $C'_0$, the boundary value. How this is calculated, has been described in Chap. 6. When this is done, the process goes forward again, solving explicitly for all unknowns, starting with

$$C'_1 = \frac{b'_1 - C'_0}{a'_1} \tag{8.29}$$

or, for a general $C'_i$,

$$C'_i = \frac{b'_i - C'_{i-1}}{a'_i} \; . \tag{8.30}$$

In Appendix E, a few examples of the use of CN are described: for a Cottrell simulation (`COTT_CN`), chronopotentiometry (`CHRONO_CN`, `CHRONO_CN_HERM`) and LSV (`LSV_CN`).

## 8.4 Using Four-Point Spatial Second Derivatives

It was shown in Chap. 7 that the three-point second spatial derivative on an unequally spaced grid, leading to (8.1) with the coefficients defined in (8.2), can be improved with relatively small effort to an asymmetric four-point formula, spanning the indices $i-1, i, i+1, i+2$, with the second derivative referred to the point at index $i$. The diffusion equation is then semi-discretised to

$$\frac{dC}{dT} = \alpha_1 C_{i-1} + \alpha_2 C_i + \alpha_3 C_{i+1} + \alpha_4 C_{i+2} \tag{8.31}$$

analogous to the three-point form (8.8). The derivation of the coefficients are described in Chap. 3, and some formulae given in Appendix A, and procedures `FORN` and `FORNBERG` are described in Appendix E. The above $\alpha$ values are again $i$-dependent. Here, we describe only the implementation of the scheme to the Laasonen method, leaving out CN. The reason is that the Laasonen method best enables the use of extrapolation, of which the simple second-order variant nicely couples with the second-order four-point approximation. Thus, using Laasonen, (8.31) becomes

$$\frac{C'_i - C_i}{\delta T} = \alpha_1 C'_{i-1} + \alpha_2 C'_i + \alpha_3 C'_{i+1} + \alpha_4 C'_{i+2} \tag{8.32}$$

which leads to the system of equations

$$\begin{array}{lllll}
C'_0 & + a_{1,1}C'_1 & + a_{2,1}C'_2 & + a_{3,1}C'_3 & = b_1 \\
C'_1 & + a_{1,2}C'_2 & + a_{2,2}C'_3 & + a_{3,2}C'_4 & = b_2 \\
\dots & & & & \\
C'_{i-1} & + a_{1,i}C'_i & + a_{2,i}C'_{i+1} & + a_{3,i}C_{i+2} & = b_i \\
\dots & & & & \\
C'_{N-2} & + a_{1,N-1}C'_{N-1} & + a_{2,N-1}C'_N & + a_{3,N-1}C'_{N+1} & = b_{N-1} \\
C'_{N-1} & + a_{1,N}C'_N & + a_{2,N}C'_{N+1} & + a_{3,N}C'_{N+2} & = b_N
\end{array} \tag{8.33}$$

with the coefficients given by

$$\begin{aligned} a_{1,i} &= \frac{\alpha_2 - 1/\delta T}{\alpha_1} \\ a_{2,i} &= \frac{\alpha_3}{\alpha_1} \\ a_{3,i} &= \frac{\alpha_4}{\alpha_1} \\ b_i &= \frac{-1}{\delta T \alpha_1} C_i \quad . \end{aligned} \tag{8.34}$$

Note that this differs from the three-point system (8.11) in that every line now has four coefficients, and their definitions have been written as $i$-dependent, which is the case for unequally spaced $X$ positions. Note also that an extra point, index $N+2$, has been added to the row of $X$ values. Point $N$ is still the last one to undergo diffusional changes, so the fact that the point $N+2$ lies past $X_{max}$ (which is at index $N$) does not matter, both these extra points being either constant or set values, not subject to diffusional changes.

The extra two intervals $X_{N+1} - X_N$ and $X_{N+2} - X_{N+1}$ need not be expanding and are best set equal to $X_N - X_{N-1}$. Alternatively, one might choose asymmetric backward-pointing difference approximation $u_3''(4)$ for the point $X_{N-1}$ and make $X_N$ the bulk point, thus obviating the need for the extra two points. However, for methods where there is a Neumann (derivative) condition at the outer boundary, there is no such choice, and then $X_N$ is the furthest point, and backwards derivative approximations must be applied on the boundary. This happens in such cases as thin layer cells, the diffusion domain discussed in connection with arrays of electrodes in Chap. 12, and a case the present authors have worked on, a flat polymer film containing an enzyme over a disk electrode [8].

The above system, although leading to a quadradiagonal system of equations, can still be solved by a smallish extension of the Thomas algorithm [9]. Consider the last two equations of (8.33) and rewrite them, putting the bulk concentration terms on the right-hand side:

$$\begin{aligned} C'_{N-2} + a_{1,N-1}C'_{N-1} + a_{2,N-1}C'_N &= b_{N-1} - a_{3,N-1}C'_{N+1} \\ C'_{N-1} + a_{1,N}C'_N &= b_N - a_{2,N}C'_{N+1} - a_{3,N}C'_{N+2} \ . \end{aligned} \tag{8.35}$$

We rewrite the last equation, which is already down to two unknowns, in the form

$$C'_{N-1} + a'_N C'_N = b'_N \tag{8.36}$$

($a'_N$ and $b'_N$ being obvious, and defined below in (8.38)) which, as before, allows the substitution for $C'_N$ in the next-last equation, which then also reduces to two unknowns,

$$C'_{N-2} + a'_{N-1}C'_{N-1} = b'_{N-1}\ . \tag{8.37}$$

Thus far, this looks just like the Thomas algorithm for the tridiagonal system, as described above in Sect. 8.3. From here on, however, the processes diverge. We need to keep both substitutions for $C'_N$ and $C'_{N-1}$ and use them in the third-last equation, which contains both. This process is continued backwards, reducing all equations with four unknowns to new ones with just two unknowns. The expressions resulting from this are the following:

$$a'_N = a_{1,N}; \qquad b'_N = b_N - (a_{2,N} + a_{3,N})C_b\ , \tag{8.38}$$

$C_b$ being the bulk concentration, equal to $C'_{N+1}$ and $C'_{N+2}$ in system (8.33). Then,

$$a'_{N-1} = a_{1,N-1} - \frac{a_{2,N-1}}{a'_N}; \qquad b'_{N-1} = b_{N-1} - a_{2,N}\frac{b'_N}{a'_N} - a_{3,N-1}C_b\ . \tag{8.39}$$

These now serve as starting values for the recursive process; the $i$th equation of system (8.33) becomes

$$C'_{i-1} + a'_iC'_i = b'_i \tag{8.40}$$

with the two new coefficients recursively given by

$$a'_i = a_{1,i} - \frac{1}{a'_{i+1}}\left(a_{2,i} - \frac{a_{3,i}}{a'_{i+2}}\right) \tag{8.41}$$

$$b'_i = b_i - a_{2,i}\frac{b'_{i+1}}{a'_{i+1}} - a_{3,i}\left(\frac{b'_{i+2}}{a'_{i+2}} - \frac{b'_{i+1}}{a'_{i+1}a'_{i+2}}\right)\ . \tag{8.42}$$

This is not so difficult to program and leads to a new system just like that in Sect. 8.3, right down to (8.28). Boundary condition handling is the same, as is the forward scan that yields all the new unknowns, Eq. (8.30).

This has been programmed into example program `COTT_EXTRAP4` described in Appendix E. Compared with the three-point program, `COTT_EXTRAP`, it yields results, using the same parameters, with an accuracy about an order of magnitude better. Once programmed and debugged, the code can be easily transplanted into other programs and seems worthwhile implementing.

Finally, as mentioned earlier (Chap. 7, Sect. 7.2), Martínez-Ortiz [10] developed some rather simple formulae for derivative approximations for the special case of exponentially expanding grid spacings, and discovered that the four-point second-order derivative approximation $u''_2(4)$, for the expansion factor $\gamma = \sqrt{2}$ is third-order in accuracy, rather than second, as it is for other $\gamma$ values. This could be an easy and useful way to increase the accuracy, using the four-point formula. However, as has also been found [11], $\gamma$ is best held below about 1.2, so this may not be such a useful value in practice.

Martínez-Ortiz et al. [12] have recently extended the formulae to up to eight-point approximations, always using an asymmetric form referring to the second point in the window, and a very large expansion factor up to 2, with good results. They were thus able to reduce the number of points in space to as little as 15 and still get accurate values. The placement of the reference point and the addition of further points beyond the outer spatial limit allowed them to use an extension of our four-point Thomas algorithm. The use of $\gamma$ as high as 2 is in contrast with our finding that it should not be more than about 1.2, possibly due to the large number of points used in the approximations.

## 8.5 Improvements on CN and Laasonen

The Laasonen method, because of the forward difference in $T$, has errors of $O(\delta T, H^2)$, and the first-order behaviour with respect to $\delta T$ limits its accuracy to about the same as the explicit method described in Chap. 5. However, it has a smooth error response to disturbances such as an initial transient (Cottrell), and is stable for any value of $\delta T/H^2$, where $H$ is either the same as all intervals if equal intervals are used in $X$, or is the smallest (usually the first) interval if unequal intervals are used. This makes the method interesting, and it will be seen below that it can be improved. For simplicity, the symbol $\lambda$ will be used below, and denotes the largest value of that parameter, that is, the value from the smallest interval in space in a given system.

CN is formally as stable as Laasonen, and more accurate, with errors of $O(\delta T^2, H^2)$. However, it has one serious drawback. If the initial conditions are a sharp change in concentration (as in potential jump experiments), CN responds with errors oscillating about zero and for large $\lambda$ values these oscillations can persist over much of the simulation period. This has meant that simulators have tended to use other methods instead. The stability and the reason for the oscillatory response of CN are explained in Chap. 15, but here, a method of damping the oscillations will be described.

To appreciate the problems with both CN and Laasonen, consider Fig. 8.1. This shows three curves, and we ignore the stippled one for the moment (but see Sect. 8.5.1.2). The plot shows simulations of the Cottrell system, using only 20 steps in the range $0 < T \leq 1$ and a $\lambda$ value of 3 (equal spatial intervals). The vertical axis is the relative error $e$ in the computed current, defined as

$$e = \frac{G_{sim}}{G_{Cott}} - 1 \tag{8.43}$$

where $G_{sim}$ is the simulated current and $G_{Cott}$ the Cottrell current, as given in (2.44) on page 19. The smoothly falling solid line is that for the Laasonen method, while

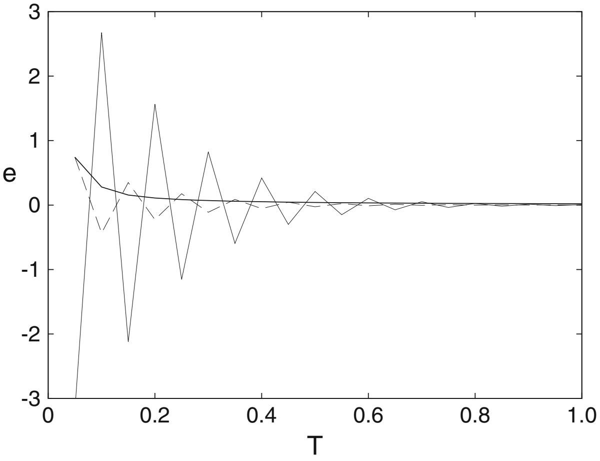


**Fig. 8.1** Relative error in computed current vs time

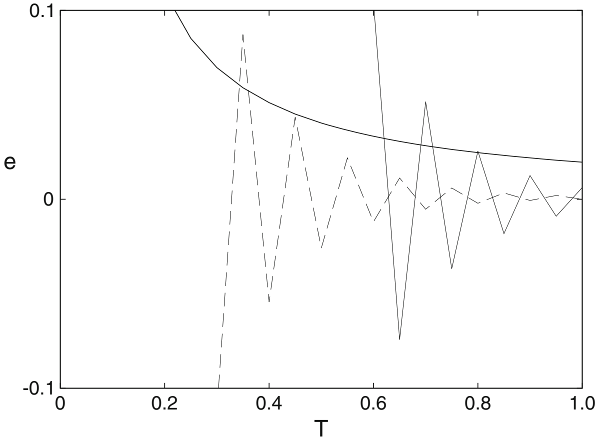


**Fig. 8.2** Relative error in computed current vs time, narrow scale

the widely oscillating solid line is from CN. The plot does not make clear that the error at $T = 1$ is in fact greater for Laasonen than for CN, but it does illustrate the problem of oscillation with CN. To show the difference in final errors, Fig. 8.2 shows the same results but with the vertical scale now narrowed down. At smaller times, the plot lies outside the range, but now it is clearly seen that although the CN curve is still oscillating, the Laasonen response has the greater error at $T = 1$.

### 8.5.1 Damping the CN Oscillations

Crank and Nicolson, in their original paper [2], recognised the oscillation problem with their method, writing “If $\gamma$ [which is their $\delta t/\delta x^2$ ] is very large an oscillatory error which only disappears very slowly may arise”. The problem is referred to in most texts describing the method. More detail is given in Chap. 15, but the essence of the problem is that CN will oscillate if $\lambda > 0.5$; in practice, a value of unity or even 2 will not cause serious oscillations. So one way to reduce oscillations is to lower $\lambda$, usually by reducing $\delta T$. This might cause unduly long execution times, but fortunately, once oscillations have been damped, they normally do not reappear, so if they can be damped within the first interval, the problem is solved. This leads to the most effective method, subdivision of the first time interval. Østerby has discussed the various methods that can be used to damp CN oscillations [13].

#### 8.5.1.1 First-Interval Subdivision

There are a few ways of subdividing the first time step: using a number $M$ of equal intervals, or a number of intervals expanding with time, usually exponentially. The two methods can be formally combined into one description. Let the full interval to be subdivided be of length $\delta T$, and let it be subdivided into $M$ smaller intervals $\tau_i, i = 1 \dots M$, such that

$$\sum_{i=1}^{M} \tau_i = \delta T \tag{8.44}$$

and

$$\tau_i = \tau_{i-1}\gamma = \tau_1\gamma^{i-1} \; . \tag{8.45}$$

Exponentially expanding subintervals are obtained if $\gamma > 1$ and equal intervals for $\gamma = 1$. The latter method is (here) called **Pearson**, after the author who first suggested it [14], in 1965. It has been studied more recently [13, 15–18]. Exponentially expanding subintervals will be called **ees** here. They were suggested [19] and later used [20–22], and studied in some detail recently [15, 18].

Whether Pearson (in the one form or the other) or ees is to be used is a matter of taste. Pearson is the simpler method, and it is simpler to determine the only parameter involved, $M$. Numerical experiments [15] show that a (sub)$\lambda$ value of about unity is sufficient to damp oscillations during the first $M$ substeps, so this sets $M$ simply to such a value as to satisfy the requirement. That is, if the main time interval $\delta T$ leads to $\lambda > 1$, then $\lambda$ subintervals are needed in order to bring the sub-$\lambda$ below unity, or $M = \lambda$, but rounded up to the nearest integer. Besides simplicity, this has the additional advantage of equal time intervals: in many simulations, the coefficients as in (8.14) depend on the time interval, and must be recalculated if that interval varies. They must thus be calculated once, prior to the $M$ substeps, and again once upon resumption of the full $\delta T$. With unequal time intervals, however, they must be recalculated before each substep. In some cases, this can have serious effects on computing time, even though in principle, fewer substeps can be used if they are expanding. The simple Pearson start is automatically used in the example program `COTT_CN` (Appendix E), and also, although as it turns out, not really necessary, in `CHRONO_CN` (chronopotentiometry does not cause oscillation problems with CN). If a large $\lambda$ value is used, Pearson can result in an excessive number of substeps, and ees will then be better.

For ees, there is no simple recipe for the choice of $M$ and $\gamma$. The reader is referred to the study [15], where several contour plots are provided that can help. A rough guide is that $\gamma = 1.5$ is a fairly universally useful value. It is the opinion of the present authors that Pearson is the best choice here.

If ees is considered desirable, there is the small matter of the determination of the parameters. Normally, one would choose $M$ first, and then either the size of the first interval $\tau_1$ (which sets the expansion parameter $\gamma$) or $\gamma$, which sets the first interval. In the former case, having chosen $M$ and $\tau_1$, the function `EE_FAC` (see Appendix E) can then be used to find the appropriate $\gamma$. In the latter case, Eq. (7.19) on page 129 can be inverted to give explicitly

$$\tau_1 = \delta T \frac{\gamma - 1}{\gamma^M - 1} \ . \tag{8.46}$$

Finally, there is another mode of operation for ees. If one considers the total simulation time as a single step, this can be subdivided into a number of exponentially expanding "subintervals" in the same manner as the above description of subdivision of the first interval. This was first suggested by Peaceman and Rachford in 1955 [23], in their famous paper describing the ADI method (see Chap. 12), and was used later [24, 25]. It is routinely used by Svir and coworkers [26, 27]. These workers tend to use strong expansion with $\gamma = 2$. The method requires a large number of recalculations of the coefficients and thus uses more computer time than equal intervals with a damping device applied to the first interval. This is however possibly compensated by a much smaller number of steps in time.

#### 8.5.1.2 Initial BI Step(s)

Rannacher and coworker [28–30] have experimented with the idea of starting a CN simulation with one or more BI steps. The rationale is that BI damps errors in a non-oscillatory manner, unlike CN, and in fact the larger $\lambda$ is, the more strongly an error is damped. This also applies to an initial transient or singularity, as encountered in a potential jump, for example. The disadvantage of BI, used for a whole simulation, is that it has a global first-order error with respect to the time interval. However, the local error for each individual step is second-order and it turns out that if one uses a fixed number of BI steps to start with and then continues with CN, the global error is still second-order. Rannacher and his coworker seem to prefer taking two or even four initial BI steps. In their second 1982 paper [28], they hint at a problem with the larger number of steps and that two might after all be preferable. However, by 1984, Rannacher again preferred four. The problem is that although the global error order does indeed remain $O(\delta t^2)$ for any fixed number $M$ of BI steps, the error itself increases with $M$. If the method is used to damp oscillations with CN, there must then be a compromise between the degree of damping and the acceptable error. The method has been studied again recently by Khaliq and Wade [31], who also advocate taking four BI steps, and more recently by others [15, 17, 18, 32], in which a single BI step was investigated, among other methods. Some tests convince the present authors, that a single step is probably the best or at most two, and if that does not remove the oscillations sufficiently, another method should be used, such as Pearson.

The method works well if $\lambda \gg 1$ or, in the case of unequal intervals or two-dimensional geometries, where there is some critical, largest effective $\lambda$ greatly exceeding unity. It was found [15] that the method works very well with a single BI step in the case of (2-D) microdisk simulations, where indeed large effective $\lambda$ values result at the disk edge and it is these that are responsible for the oscillations if CN is used.

The dashed line in Figs. 8.1 and 8.2 is the response for a single Laasonen step followed by CN from then on. For this simulation, $\lambda$ was set at 3, not a very large value. Nevertheless, the single Laasonen step has clearly reduced the oscillation amplitudes.

As a final note on this method, it is worth noting that Wood and Lewis [33] essentially used this method, possibly not realising it. In an investigation of how to damp CN's oscillations, one of the methods they tried was to average the initial values of the simulated quantities with the result of the first CN step. It can readily be shown that this is equivalent to taking a single BI step of half a time interval. They found some damping, but they must also have introduced an error in the time by half an interval, which would persist thereafter and degrade the accuracy, probably to first-order.

#### 8.5.1.3 Averaging and Extrapolation

Lindberg [34] investigated smoothing of the trapezoidal response in the solution of systems of *ode*s, and this is of course a related problem. The idea of averaging is that if a sequence of errors show alternate signs (which is what we mean by oscillation), then combining several in a sequence might eliminate or reduce the error. Lindberg used a three-point averaging formula, and combined it with extrapolation (see below, and Sect. 4.9). Extrapolation alone, used with CN, does not damp oscillations but does help, marginally, in conjunction with averaging. It seems, however, that Lindberg's results do not justify these techniques.

#### 8.5.1.4 Singularity Correction

Clearly, the largest errors in both simulated concentration and current values for any simulation involving an initial potential or current step occur just after onset of the perturbation, or singularity. BI shows large errors, smoothly damped in subsequent steps, while CN has damped oscillatory errors. As Bieniasz writes [35], this is because of "the lack of compatibility between partial differential equations (PDEs) and their initial and boundary conditions". He followed the work of Flyer and Fornberg [36], who studied this problem for general *pde*s. Bieniasz modified their method to obtain accurate responses for the one-dimensional problem involving a homogeneous chemical reaction. The essence of this is to partition the equations and boundary values into two, one of them being the pure diffusion case for which there is a solution (for example, the Cottrell equation) and which holds to high accuracy at very small times, including the first time interval, plus the one involving the chemical reaction. Upon applying the analytical solution, the other partition now has compatible initial conditions. The reader is referred to Bieniaz's work [35] for detail, but suffice it to say here that this produced highly accurate concentrations and currents without large initial errors.

#### 8.5.1.5 Recommendations

A choice needs to be made. The reader may or may not want to experiment with the various possibilities. It is possible to provide some guidance here. Clearly, if $\lambda$ is very large, then $M$ can become ridiculously large if program `COTT_CN` is used—$M$ will be equal to $\lambda$. It is in such cases, however, that the BI method works best. So a rough guide might be the following. For $3 \le \lambda \le 100$, use Pearson; for larger $\lambda$, the BI start method might be favourable, perhaps taking 2–4 initial BI steps despite the slight loss in accuracy.

### *8.5.2 Making Laasonen More Accurate*

In contrast to CN, Laasonen has a very acceptable error response, damping the error (and initial concentration transients) smoothly, especially at high $\lambda$; but it has the disadvantage of poor accuracy, being globally first-order with respect to $\delta T$. There are two popular ways of increasing the accuracy (raising the order) of the method, while preserving the smooth error response.

The two methods are BDF and extrapolation. Both methods are used for the numerical solution of *ode*s and are described in Chap. 4. The extension to the solution of *pde*s is most easily understood if the *pde* is semidiscretised; that is, if we only discretise the right-hand side of the diffusion equation, thus producing a set of *ode*s. This is the Method of Lines or **MOL**. Once we have such a set, as seen in (8.9), the methods for systems of *ode*s can be applied, after adding boundary conditions.

#### 8.5.2.1 BDF

The BDF method has been described in Chap. 4. One starts with the system such as (8.9), and goes on from there as described. This was first suggested for *pde*s by Richtmyer in 1957 [37], who suggested the three-point variant, and was first used in electrochemistry by Mocak and Feldberg [38] and later refined to variable time intervals by Feldberg and Goldstein [24]. These workers call it FIRM, an abbreviation of "**F**inite **I**mplicit **R**ichtmyer **M**odification". The modification referred to is that of the BI (or Laasonen) method to a multi-level backward differentiation method. These authors also use a very simple start-up strategy, described as the **simple start with correction** in Sect. 4.8.1 on page 69. As is shown in Appendix C, this method, by good luck, provides second-order accuracy at the corrected times. The second-order nature of it also implies, however, that there is little sense in going to more than three-point BDF. With three points in time, the global error is of $O(\delta T^2)$, and although an increase in the number of points included in the BDF algorithm raises that order (up to seven-point can be used, and Feldberg and coworkers now routinely use five-point or fourth-order), this is held down to second-order by the start-up method. However, a second-order method is very useful, being of the same order as CN, and three-point BDF has the smooth error response of Laasonen. One small drawback of the method is that additional concentration vectors must be kept in memory; in the case of three-point, one extra array is needed. This is not so bad, and the results might be considered worth the effort.

There have been attempts to improve the performance of BDF, which is normally limited by the second-order (in the spatial interval $H$) discretisation of the spatial derivative. Higher-order spatial second derivatives have been tried out in connection with BDF [39, 40]. They can only work as intended if a high-order start is used, such as the **KW** start as described in Sect. 4.8.1. This start was not found to be efficient in [40], but it may be that a technique other than the one used there, such as Numerov (see Chap. 9), which does not produce banded matrices, will make the use of KW efficient and thus interesting. For this reason, the KW start is described below.

First, it is worthwhile detailing the implementation of BDF itself, ignoring startup for the moment. We choose three-point BDF. Based on (4.28) given on page 68 for *ode*s, the diffusion equation (8.8) is discretised at index $i$ as

$$\frac{\frac{1}{2}\,{}'C_i - 2C_i + \frac{3}{2}C'_i}{\delta T} = \alpha_1 C'_{i-1} + \alpha_2 C'_i + \alpha_3 C'_{i+1} \tag{8.47}$$

where ${}'C_i$ indicates the concentration at point $i$ and $T-\delta T$, that is, the past (known) concentration point. When this system of discrete equations is rearranged, it is of the same form as (8.11), with the new coefficients

$$\begin{aligned} a_{1,i} &= \frac{\alpha_2 - \frac{3}{2}/\delta T}{\alpha_1} \\ a_2 &= \frac{\alpha_3}{\alpha_1} \\ b_i &= \frac{{}'C_i}{2\delta T \alpha_1} - \frac{2C_i}{\delta T \alpha_1} \end{aligned} \tag{8.48}$$

which is seen to be hardly more complicated than those for Laasonen. There is the added step at every iteration of moving the concentration rows down one level, that is, the rows for the present level $T$ and $T+\delta T$ (now computed) down to, respectively, $T-\delta T$ and $T$.

Now consider the KW start for BDF. The description in [40] will be followed here. First of all, (8.47) is rewritten in *ode* form for the whole system, replacing the left-hand side by the time derivative and the right-hand side by the general matrix form

$$\frac{\mathrm{d}\mathbf{c}_j}{\mathrm{d}t} = \frac{\lambda}{\delta t}\left(\mathbf{A}\mathbf{c}_j + \mathbf{s}\right) \tag{8.49}$$

where $\mathbf{c}_j$ refers to the whole $\mathbf{c}$ vector across the spatial dimension at time index $j$, matrix $\mathbf{A}$ arises from the coefficients such as those on the right-hand side of (8.47) (but may be those from any other discretisation, including multi-point or Numerov, see the next chapter), and $\mathbf{s}$ is the vector arising from boundary conditions. The left-hand side of this equation now leaves us free to choose the particular BDF form. As mentioned, the problem to solve here is providing the first few concentration rows. It might be thought that for $k$-point BDF, $k-2$ new $\mathbf{c}_j$ are needed, those for $j = 1\dots k-2$ (we already have the initial row for $j = 0$). Indeed this is true, but it turns out that the row $k-1$ is best also included in the calculation, as this produces values of the same accuracy order with respect to $t$ as the subsequent BDF steps. This was mentioned in the chapter on *ode*s, Sect. 4.10. There is then always one more equation to choose from than needed, and a choice has to be made which ones to use. This is more or less arbitrary. For example, for three-point BDF, we calculate rows for $j = 1$ and 2. For these, we can refer the time derivative to two out of levels 0, 1 and 2. If we choose levels 1 and 2, we have the two matrix equations

$$\frac{\mathbf{c}_2 - \mathbf{c}_0}{2\delta t} = \frac{\lambda}{\delta t}\left(\mathbf{A}\mathbf{c}_1 + \mathbf{s}\right) \tag{8.50}$$

referring to $j = 1$ (thus a central difference), and

$$\frac{\mathbf{c}_0 - 4\mathbf{c}_1 + 3\mathbf{c}_2}{2\delta t} = \frac{\lambda}{\delta t}\left(\mathbf{A}\mathbf{c}_2 + \mathbf{s}\right) \tag{8.51}$$

employing a BDF form referring to $j = 2$. These two equations combine into the single block matrix equation

$$\begin{bmatrix} 2\lambda\mathbf{A} & -\mathbf{I} \\ 4\mathbf{I} & (2\lambda\mathbf{A} - 3\mathbf{I}) \end{bmatrix} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix} = \begin{bmatrix} -\mathbf{c}_0 - 2\lambda\mathbf{s} \\ \mathbf{c}_0 - 2\lambda\mathbf{s} \end{bmatrix} . \tag{8.52}$$

Corresponding higher-order forms can now be constructed using the examples in Chap. 4, Sect. 4.10. In the present context, the matrix equations get rather large for larger $k$. If there are $N$ unknowns across the spatial dimension, then the matrix equation will be $(k-1)N \times (k-1)N$. So the method might be suitable only for smallish $N$.

Generally, it is sufficient in our experience to use a rational start for BDF, which is simply a single BI step for three-point BDF, and this has become our own practice.

#### 8.5.2.2 Extrapolation

Extrapolation is described in Chap. 4, Sect. 4.9. It can easily be adapted to *pde*s, as first suggested by Liebmann [41] in 1955, but ascribed to Lawson and Morris [42], followed by Gourlay and Morris [43]. Strutwolf et al. [44, 45] first described its use in electrochemistry, and a higher-order variant was described later [46], in an attempt to take advantage of the higher-order extrapolation schemes. Bieniasz [47] now gives extrapolation based on BI the name **LMGE-x** (with x the order, for example LMGE-2), meaning Lawson-Morris-Gourlay extrapolation. Here it will be simply called extrapolation. The “BI” might be redundant, as it is almost always BI that forms the basis for extrapolation, although other methods can in principle be enhanced by extrapolation. Thus, Hartree and Womersley [3] used it in connection with their method, which has the essential elements of CN. There is a review by Deuflhart [48] and Twizell also used extrapolation in conjunction with CN [49], as did Zhou [50] with ADI.

As with BDF, the simpler second-order scheme appears about optimal. This method also shows the same smooth and damped error response of Laasonen, with the accuracy of CN. The drawback is that for every step, several calculations must be performed—in the case of second-order extrapolation, three in all (see Sect. 4.9). This also implies an extra concentration array, for the final application of the formula, for example the vector equivalent of (4.31), requiring the result of the first, whole step, and then the result of the two half-steps. Discretisation for extrapolation is the same as for Laasonen [coefficients as in (8.12)], but using two different values of $\delta T$ and therefore two coefficient matrices. These can of course be precomputed if the coefficients are constant over the simulation period, so this is not a great problem. There are example programs using extrapolation (`COTT_EXTRAP` and `COTT_EXTRAP4`) referred to in Appendix E.

## 8.6 Homogeneous Chemical Reactions

Homogeneous chemical reactions (*hcr*s) have already been mentioned in Chap. 5, where a simple explicit treatment is given. Some of the problems are also mentioned there. For the explicit method, the main one is that if a term like $K\delta T$ in the discrete equation exceeds a few percent, the simulation is inaccurate [51]. For large rate constants, this means unacceptably small $\delta T$ values leading to very long computation times. Improvements were sought at the time, such as the use of Runge–Kutta integration either for just the *hcr* part [51] or for the whole simulation [52], and tricks such as the "sequential method" (described in Chap. 5). These did not work very well. In a previous work [53], one of the present authors (DB) classified *hcr*s into the three categories slow, medium and fast, and for each of them a different method was suggested. Slow *hcr*s could be handled by the explicit method, medium-rate ones by implicit methods or Runge–Kutta, and fast reactions could only be handled by mathematical tricks such as that of Ruzić [54, 55], the so-called heterogeneous equivalent, in which the *hcr* was combined with the heterogeneous electron transfer reaction into a single new one with a different (equivalent) heterogeneous rate constant.

Such tricks are no longer needed. Starting in the early 1990s, several advances were made in simulation that have solved all the problems, and we can now handle all *hcr*s with efficient implicit methods in a straight-forward manner. These solutions are described in what follows.

The main problems that have been solved are these:

- thin reaction layers
- nonlinear equations (and negative concentrations)
- coupled systems.

The problem of thin reaction layers is described sufficiently in Chap. 5. The solution is to use unequal intervals, that is, a few very small intervals near the electrode, so that there are sample points within the thin profile. This can be done up to a point by a fixed unequal grid such as the exponentially expanding grid described in Chap. 7. A more flexible approach is the use of adaptive grids also described in that chapter. This problem is thus solved and needs no further attention here.

In Appendix E, the program `CV_EC` is described, which simulates a CV for a simple EC reaction.

### *8.6.1 Nonlinear Equations*

If a given *hcr* is of higher than first order, nonlinear terms arise in the dynamic equation(s). With terms, for example, in squared concentrations (see below), there is the danger, due to computational errors, that a concentration becomes negative, after which it can never be corrected. The technique CN is especially prone to this, because of the oscillations it engenders as a response to sharp transients such as a potential jump. This is one reason some workers prefer the Laasonen method or its improved offshoots, which have a smooth error response without any oscillations. With a Pearson start, however, CN can be used safely, without the appearance of negative concentrations, as has been seen in the above.

Until fairly recently, the problem was regarded as too hard. For example, Fisher and Compton [56], in a study involving coupled equations with second-order terms, used explicit discretisation for the second-order terms. This degrades the accuracy of the simulation and forces very small time intervals.

With the usual nonlinear terms, which are either of the form of a squared term or the product of two species' concentrations, there are two approaches. One of them is to approximate the nonlinear terms by linearised terms. The approximations are different for CN and Laasonen.

#### 8.6.1.1 Linearising Squared Concentration Terms

This has been mentioned in several papers, but the first to describe such approximations were Mastragostino et al. in 1968 [57]. Let the squared term be $C^2$; for example, a term in $-KC^2$ in the dynamic equation. The change $\delta C$ is equal to $C'-C$.

For the Laasonen method, $C^2$ is expressed as the square of the next, unknown concentration, $C'^2$. We have

$$\begin{aligned} C'^2 &= (C + \delta C)^2 \\ &= C^2 + 2C\delta C + \delta C^2 \\ &\approx C^2 + 2C(C' - C) \\ &= 2CC' - C^2 \,. \end{aligned} \tag{8.53}$$

This is now a linear expression in the unknown, $C'$, since $C$ is known. The approximation is $O(\delta C^2)$, since a term of that order was dropped.

For CN, discretisation makes the squared term the mean of the old and new terms, so

$$\begin{aligned} \tfrac{1}{2}(C^2 + C'^2) &= \tfrac{1}{2}(C^2 + (C + \delta C)^2) \\ &\approx \tfrac{1}{2}(2C^2 + 2C\delta C) \\ &= C^2 + C(C' - C) \\ &= CC' \,. \end{aligned} \tag{8.54}$$

Again, this is $O(\delta C^2)$.

#### 8.6.1.2 Linearising the Product of Concentrations of Two Species

For convenience, the two species' concentrations are given the symbols $A$ and $B$ here, with $A'$ and $B'$ the unknowns. Lerke et al. mention this briefly [58], for the DuFort–Frankel method [59], for some time a method suggested by Feldberg [60].

For Laasonen, we then have

$$\begin{aligned} A'B' &= (A + \delta A)(B + \delta B) \\ &= AB + B\delta A + A\delta B + \delta A\,\delta B \\ &\approx AB + B(A' - A) + A(B' - B) \\ &= A'B + AB' - AB\ . \end{aligned} \tag{8.55}$$

For CN,

$$\begin{aligned} \tfrac{1}{2}(AB + A'B') &= \tfrac{1}{2}(AB + (A + \delta A)(B + \delta B)) \\ &\approx \tfrac{1}{2}(2AB + B\delta A + A\delta B) \\ &= \tfrac{1}{2}(2AB + B(A' - A) + A(B' - B)) \\ &= \tfrac{1}{2}(A'B + AB')\ . \end{aligned} \tag{8.56}$$

This covers all the cases.

#### 8.6.1.3 An Example Case: Linearising

To show how this is done both in the linearised and the nonlinear form, a simple example is chosen, having the advantage of being a single-species mechanism. It is that described by Birk and Perone [7]. The electroactive substance A is formed at a uniform (bulk) concentration in a cell by a flash of light. It begins immediately to decay in a second-order *hcr*, while being electrolysed in a Cottrell-like experiment. This system will be called **BP** here. The equations are

$$\begin{gathered} \mathrm{A} + e^- \rightarrow \mathrm{B} \\ 2\mathrm{A} \rightarrow prod \end{gathered} \tag{8.57}$$

with the chemical step irreversible and with (dimensionless) rate constant $K$. The normalised dynamic equation is then

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} - 2KC^2\ . \tag{8.58}$$

The number 2 seems to be controversial but seems logical because every time two molecules of A react, both are removed from solution. Birk and Perone presented a solution for the current, but this was incorrect and was later corrected and augmented by solutions for various electrode geometries [61]. So a solution exists that can be used to test a simulation.

The linearised version is discretised for CN, using (8.54), as

$$\begin{aligned}\frac{C'_i - C_i}{\delta T} &= \frac{1}{2}\left(\alpha_1 C'_{i-1} + \alpha_2 C'_i + \alpha_3 C'_{i+1} + \alpha_1 C_{i-1} + \alpha_2 C_i + \alpha_3 C_{i+1}\right) \\ &\quad - 2KC_iC'_i\,. \qquad (8.59)\end{aligned}$$

Equation (8.59) then becomes a system like (8.11), except that the middle term on the left-hand side is different:

$$C'_{i-1} + (a_{1,i} + a_{k,i}C_i)C'_i + a_2 C'_{i+1} = b_i \qquad (8.60)$$

where the new coefficient is given by

$$a_{k,i} = -4K/\alpha_1 \qquad (8.61)$$

and $b_i$ is exactly as already defined in (8.14); it does not contain a term arising from the *hcr*. The system contains, besides the constant coefficients, concentration terms that vary from step to step. In a given program, the *a*-coefficients can be precomputed but the multiplication with the (known) concentrations must be performed at every step. Reduction to the didiagonal form (the first step of the Thomas algorithm), described by (8.16)–(8.28) will be modified in that (8.18) becomes

$$a'_N = a_{1,N} + a_{k,N}C_N \qquad (8.62)$$

and the *i*th equation becomes

$$a'_i = a_{1,i} + a_{k,i}C_i - \frac{a_2}{a'_{i+1}}\,. \qquad (8.63)$$

The expressions for the $b'$ are unchanged, except for the outer value, where attention must be given to the fact that the bulk concentration itself changes (decreases) with time. Since a solution is available here (see any text on physical chemistry, for example [62]), it may as well be used. In dimensionless terms and taking the factor 2 into account, it is

$$C_{N+1}(T) = (1 + 2KT)^{-1}\,. \qquad (8.64)$$

Consistent with CN custom, then, the last equation in the system is

$$C'_{N-1} + (a_{1,N} + a_{k,N}C_N)\, C'_N = b_N - a_2C'_{N+1} \tag{8.65}$$

in which the term $b_N$ (from (8.13)) contains the bulk value at time $T$ (the old value), but the last term on the right-hand side contains the new value for $T + \delta T$. It is important to do this correctly, for accuracy.

The above is incorporated into the example program `BP_LIN` described in Appendix E.

#### 8.6.1.4 An Example Case: Nonlinear

We can also choose not to linearise the nonlinear term by an approximation, in which case we do not run the (minimal) risk of adding errors to the simulation by the linearising approximation. The same example as used above (8.57) and again choosing CN as the method, the dynamic equation (8.58) is discretised as

$$\begin{aligned}\frac{C'_i - C_i}{\delta T} = \frac{1}{2}\big(&\alpha_1C'_{i-1} + \alpha_2C'_i + \alpha_3C'_{i+1} + \alpha_1C_{i-1} + \alpha_2C_i + \alpha_3C_{i+1}\\ &- 2KC_i^2 - 2KC_i'^2\big)\end{aligned} \tag{8.66}$$

in which the term in $2KC_i'^2$ is left in its nonlinear form (along with the other nonlinear form $2KC_i^2$, but this one is a known quantity and causes no trouble). This engenders a new nonlinear system of equations

$$C'_{i-1} + a_{1,i}C'_i + a_{k,i}C_i'^2 + a_2C'_{i+1} = b_i \tag{8.67}$$

where $a_{k,i}$ is defined differently:

$$a_{k,i} = -2K/\alpha_1 \tag{8.68}$$

and also the right-hand side, $b_i$, is different, containing a chemical term involving $a_{k,i}$:

$$b_i = -C_{i-1} - a_{3,i}C_i - a_{k,i}C_i^2 - a_2C_{i+1}\ . \tag{8.69}$$

The above equations (8.67), of which there are $N$ for $i = 1 \ldots N$, form a new system, which we rewrite in a new form, using the symbol $C$ for what is to become $C'$ at the end of the Newton iterations (**b** is constant during the process). The equations are the system, for $i = 1 \ldots N$,

$$f_i(\mathbf{C}) = C_{i-1} + a_{1,i}C_i + a_{k,i}C_i^2 + a_2C_{i+1} - b_i\,. \tag{8.70}$$

At the start of an iteration, all $C$ are equal to the known $C$. We seek a solution such that all $f_i$ are equal to zero, which they are not at the start of the step. The Newton method, which will be used, takes a number of steps, correcting the $C$ values at each step. Before we can do this, we must pay attention to the first and last equations of the set. The first one ($i = 1$) contains the boundary value $C_0$. If the system is that for a Cottrell experiment, then that value is zero, so that the equation is

$$f_1(\mathbf{C}) = a_{1,1}C_1 + a_{k,1}C_1^2 + a_2C_2 - b_1 \tag{8.71}$$

which is very simple. If we have derivative boundary conditions and (sensibly, with unequal intervals) use a two-point approximation for $G$, then $C_0$ can be replaced by a linear form in $C_1$ according to the procedures described in Chap. 6. If a multi-point derivative approximation is desired, that expression will be a linear combination of several $C_i, i = 1 \ldots n - 1$, which is more complicated and is not recommended. For this example, we stay with Cottrell.

The other equation needing attention is that for $i = N$, being

$$f_N(\mathbf{C}) = C_{N-1} + a_{1,N}C_N + a_{k,N}C_N^2 + a_2C_{N+1} - b_N \tag{8.72}$$

which contains the bulk value $C_{N+1}$. This is already known, being in this case given by (8.64) for the time $T + \delta T$. It is not part of the unknown set. Note that the last term $b_N$ is the expression for (8.69) for $i = N$ and contains an old bulk value. It is important in the program to distinguish between these two different bulk values.

We are now ready to implement the Newton method. We wish to correct $\mathbf{C}$. For details of the Newton method used on a set of nonlinear equations, see a text like Press et al. [63]. More briefly here, Taylor expansion of the system (8.66) around the current $C$ to the corrected $\mathbf{C} + \boldsymbol{\delta C}$ where $\boldsymbol{\delta C}$ is the correction term row, produces the set of equations linear in $\boldsymbol{\delta C}$,

$$\begin{aligned}
f(C_1 + \delta C_1) &= f(C_1) + (a_{1,1} + 2a_{k,1}C_1)\delta C_1 + a_2\delta C_2 \\
f(C_2 + \delta C_2) &= f(C_2) + \delta C_1 + (a_{1,2} + 2a_{k,2}C_2)\delta C_2 + a_2\delta C_3 \\
&\ldots \qquad (8.73) \\
f(C_i + \delta C_i) &= f(C_i) + \delta C_{i-1} + (a_{1,i} + 2a_{k,i}C_i)\delta C_i + a_2\delta C_{i+1} \qquad (8.74) \\
&\ldots \\
f(C_N + \delta C_N) &= f(C_N) + \delta C_{N-1} + (a_{1,N} + 2a_{k,N}C_N)\delta C_N
\end{aligned}$$

where the $\delta C_i$ are the unknowns, the correction terms. In vector/matrix notation, defining $\mathbf{C} \equiv [C_1, C_2, \ldots C_N]^T$, $\boldsymbol{\delta C} \equiv [\delta C_1, \delta C_2, \ldots \delta C_N]^T$ and the Jacobian, defined as

$$\mathbf{J}(\mathbf{C}) = \begin{bmatrix} \frac{\partial f_1}{\partial C_1} & \frac{\partial f_1}{\partial C_2} & \cdots & \frac{\partial f_1}{\partial C_N} \\ \frac{\partial f_2}{\partial C_1} & \frac{\partial f_2}{\partial C_2} & \cdots & \frac{\partial f_2}{\partial C_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_N}{\partial C_1} & \frac{\partial f_N}{\partial C_2} & \cdots & \frac{\partial f_N}{\partial C_N} \end{bmatrix} \tag{8.75}$$

becomes

$$\mathbf{J} \equiv \begin{bmatrix} (a_{1,1} + 2a_{k,1}C_1) & a_2 & & & \\ 1 & (a_{1,2} + 2a_{k,2}C_2) & a_2 & & \\ & 1 & (a_{1,3} + 2a_{k,3}C_3) & a_2 & \\ & \ddots & & \ddots & \\ & & & 1 & (a_{1,N} + 2a_{k,N}C_N) \end{bmatrix} \tag{8.76}$$

(only nonzero elements are shown), this becomes

$$\mathbf{F}(\mathbf{C} + \boldsymbol{\delta C}) = \mathbf{F}(\mathbf{C}) + \mathbf{J} \cdot \boldsymbol{\delta C} \tag{8.77}$$

and expecting convergence and thus setting $\mathbf{F}(\mathbf{C} + \boldsymbol{\delta C})$ to zero, we get the new, now linear system,

$$\mathbf{J} \cdot \boldsymbol{\delta C} = -\mathbf{F}(\mathbf{C}) \tag{8.78}$$

which is a tridiagonal system that can be solved by the Thomas algorithm as usual, for $\boldsymbol{\delta C}$. One must then either check the residual (8.77); its norm should be below some value one sets, such as $10^{-6}$. Alternatively, one can check the correction vector $\boldsymbol{\delta C}$. If its norm is below that small value, then no further iterations need be carried out. The first method requires an extra calculation, while the second always requires a last extra iteration because even if the very first iteration yields the correct set of $\boldsymbol{\delta C}$ values, that set itself will not be zero, but the second set will be.

The above is implemented in the program `BP_NONLIN` (Appendix E). One finds that 2–3 iterations tend to be enough, and the results are very slightly better, for a given set of simulation parameters, than those from the linearised version, `BP_LIN`.

A further method to handle nonlinearities directly without iteration, the Rosenbrock method, is explained in Chap. 9.

### *8.6.2 Coupled Equations*

Coupled equations are those in which some or all of the dynamic equations have terms in more than one of the variables (concentrations). This leads, upon discretisation, to systems of discrete equations that cannot usually be solved using the plain Thomas algorithm because, no matter how one orders the concentration vectors, the systems correspond to matrix equations that are more than tridiagonal or banded. It is not very long ago that this too was regarded as too hard, and the explicit method was used, as this presents no special problems—except for accuracy and computer time. Two techniques solved the problem: the block matrix method, and direct matrix equation solving. The block matrix method, as will be seen below, is itself a method of solving the matrix equations but, by proper vector ordering and blocking, makes the matrix into a block-tridiagonal system, which can be solved by a kind of block-Thomas algorithm. It was reintroduced to electrochemistry by Rudolph in 1991 [64]. The technique was known outside electrochemistry since 1952 [65] and since [66–74], and within the field [64, 75–80]. Newman used the technique in 1968 [81] and Honeychurch mentions it in his book [82]. Other methods exist to deal with the problem of banded matrices. The most notable of these are the "strongly implicit procedure" (SIP) of Stone [83], used recently by Alden et al. [84–86], the Krylov method [87], used by the same team [86, 88, 89] and by Bard et al. [90] and Welford et al. [91]; and the multigrid method [92], also used by Alden et al. [86]. The Alden et al. team made use of ready-made commercial subroutines. These techniques are not trivial to apply, and only the block matrix method will be described here.

To illustrate the block matrix method, we take the relatively simple two-species catalytic or EC$'$ mechanism

$$\begin{aligned} \mathrm{O} + e^- &\Leftrightarrow \mathrm{R} \\ \mathrm{R} &\rightarrow \mathrm{O} \end{aligned} \tag{8.79}$$

as already mentioned in Chap. 6, page 112. As mentioned there, this leads, through the obvious dynamic equations, to the discretised pair of equations for a given point along $X$ with index $i$,

$$\begin{array}{llllll} C'_{O,i-1} & +a_{1,i}C'_{O,i} & & +a_{k,i}C'_{R,i} & +a_2C'_{O,i+1} & = b_{O,i} \\ C'_{R,i-1} & & +(a_{1,i} - a_{k,i})C'_{R,i} & & +a_2C'_{R,i+1} & = b_{R,i} \end{array} \tag{8.80}$$

(we assume here that the coefficients are the same for both species and, again, that $a_2$ is constant). The coefficients depend on the simulation method used (CN, Laasonen or its variants) and on the sample point distribution, as already described.

This pair of equations produces, when written for all $i$, a system of $2N$ equations. It is convenient to order the unknown concentrations in the sequence $C_{O,0}, C_{R,0}, C_{O,1}, C_{R,1}, \ldots, C_{O,N}, C_{R,N}$—that is, with $C_O$ and $C_R$ alternating. The system then becomes a pentadiagonal matrix equation, with perhaps some elements off the five diagonals, depending upon the way derivative boundary conditions are discretised. It is true that there are pentadiagonal solvers similar to the Thomas algorithm [93], but we stay with the block matrix method here because the example is a simple one, and the method can be extended to more than two species, which widen the band in the matrix to more than pentadiagonal. Incidentally, in this specific case, there is a method of using something like the Thomas algorithm, using a double recursive method without blocks, but it is specific to this example, has no general utility and will not be gone into here (unpublished work by DB).

The concentration vector is now lumped into pairs, rendering it into a vector of two-element vectors; let

$$\mathbf{C}_i \equiv \begin{bmatrix} C_{O,i} \\ C_{R,i} \end{bmatrix} \tag{8.81}$$

and this allows us to write (8.80) in the more compact form, already seen in Chap. 6, Eq. (6.57),

$$\mathbf{C}'_{i-1} + \mathbf{A}_i\mathbf{C}'_i + \mathbf{a}_2\mathbf{C}'_{i+1} = \mathbf{B}_i \tag{8.82}$$

with the definitions

$$\mathbf{A}_i \equiv \begin{bmatrix} a_{1,i} & a_{k,i} \\ 0 & (a_{1,i} - a_{k,i}) \end{bmatrix}, \tag{8.83}$$

$$\mathbf{B}_i \equiv \begin{bmatrix} b_{O,i} \\ b_{R,i} \end{bmatrix}. \tag{8.84}$$

The last equation of this new system, for $i = N$, can be used in the same manner as the last equation in the scalar system (8.11), as the last term on the left-hand side is known, $\mathbf{C}'_{i+1}$ being the bulk values. In the scalar system, the equations, each with three unknowns, were reduced to a new set, each with two unknowns, generating a new set of scalar coefficients $a'_i$ and $b'_i$. The same process can be used here, but working with vectors and matrices. Applying the same approach as the above, (and therefore not needing a lot of explanation), we write, analogous to (8.18),

$$\mathbf{A}'_N = \mathbf{A}_N \tag{8.85}$$

and as in (8.19)

$$\mathbf{B}'_N = \mathbf{B}_N - a_2\mathbf{C}'_{N+1}\,. \tag{8.86}$$

Then, analogous to the scalar process above, Eqs. (8.26) and (8.27), we have the recursive relations, going backwards from $N$,

$$\mathbf{A}'_i = \mathbf{A}_i - a_2(\mathbf{A}'_{i+1})^{-1} \tag{8.87}$$

and

$$\mathbf{B}'_i = \mathbf{B}_i - a_2(\mathbf{A}'_{i+1})^{-1}\mathbf{B}'_{i+1} \;. \tag{8.88}$$

This is continued down to $i = 1$, giving the system (6.61), described on page 113.

It turns out in practice that only the inverses of the new coefficient matrices $\mathbf{A}'$ are needed for the last step, so only these inverses need be stored. Since the matrices are small (in this case, just 2 × 2), the inversions can be efficiently computed.

Before the last sweep can be carried out, the boundary concentration vector $\mathbf{C}'_0$ is needed, and how this is calculated is fully described in Chap. 6, Sect. 6.4, starting on page 113. When this has been done, all the new concentrations can at last be computed, from the forward-sweeping recursive expressions

$$\mathbf{C}'_i = (\mathbf{A}'_i)^{-1}\left(\mathbf{B}'_i - \mathbf{C}'_{i-1}\right) \;, \tag{8.89}$$

seen to be analogous with (8.30). If the concentrations are stored separately for each species, then it might be most convenient to put each $\mathbf{C}_i$ vector away into its place in those arrays as soon as they are computed; it is matter of personal strategy, how to store the values.

The program CV_CAT (see Appendix E) is an example of a simulation of this system, for cyclic voltammetry.

## References

1. Laasonen P (1949) Über eine Methode zur Lösung der Wärmeleitungsgleichung. Acta Math 81:309–317
2. Crank J, Nicolson P (1947) A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. Proc Camb Philos Soc 43:50–67. Reprinted in Adv Comput Math 6:207–226 (1996), with some slight changes to the list of references
3. Hartree DR, Womersley JR (1937) A method for the numerical or mechanical solution of certain types of partial differential equations. Proc R Soc Lond A 161:353–367
4. Thomas LH (1949) Elliptic problems in linear difference equations over a network. Watson Scientific Computing Laboratory, Columbia University, New York
5. Thomas LH (1949). Elliptic problems in linear equations over a network. Special Collection Dept. of the North Carolina State Univ. Libraries, the Llewellyn Hilleth Thomas Papers 1921–1989, item 210.8–13.8
6. Reinert KE, Berg H (1962) Theorie der polarographischen Verfolgung schneller chemischer Reaktionen in Lösung mittels reaktionsbedingter Diffusions-Zeit-Kurven. Monatsber Deut Akad Wiss Berlin 4:26–32

7. Birk JR, Perone SP (1968) Electrochemical studies of rapid photolytic processes. A theoretical and experimental evaluation of potentiostatic analysis in flash photolysed solutions. Anal Chem 40:496–500
8. Britz D, Strutwolf J (2015) Digital simulation of chronoamperometry at a disk electrode under a flat polymer film containing an enzyme. Electrochim Acta 152:302–307
9. Britz D, Strutwolf J (2003) Higher-order spatial discretisations in electrochemical digital simulation. Part 4. Discretisation on an arbitrarily spaced grid. Comput Biol Chem 27:327—337
10. Martínez-Ortiz F, Zoroa N, Molina Á, Serna C, Laborda E (2009) Electrochemical digital simulations with an exponentially expanding grid: general expressions for higher order approximations to spatial derivatives. The special case of four-point formulas and their application to multipulse techniques in planar and any size spherical electrodes. Electrochim Acta 54:1042–1055
11. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376
12. Martínez-Ortiz F, Zoroa N, Laborda E, Molina A (2016) Brute force (or not so brute) digital simulation in electrochemistry revisited. Chem Phys Lett 643:71–76. Supplementary material in the form of C++ programs
13. Østerby O (2008) Numerical solution of parabolic equations. Technical Report Daimi FN-65, Dept. of Computer Science, University of Aarhus, Denmark. Accessible at http://www.daimi.au.dk/~oleby/notes/partial.pdf
14. Pearson CE (1965) Impulsive end condition for diffusion equation. Math Comput 19:570–576
15. Britz D, Østerby O, Strutwolf J (2003) Damping of Crank-Nicolson error oscillations. Comput Biol Chem 27:253–263
16. Fang H, Chen HY (1997) A strategy to improve the accuracy of digital simulation for electroanalytical chemistry. Chin J Chem 15:250–259
17. Martínez-Ortiz F, Molina A, Laborda E (2011) Electrochemical digital simulation with highly expanding grid four point discretization: can Crank-Nicolson uncouple diffusion and homogeneous chemical reactions? Electrochim Acta 56:5707–5716
18. Østerby O (2002) Five ways of reducing the Crank-Nicolson oscillations. Technical Report Daimi PB-558, Dept. of Computer Science, Aarhus University
19. Britz D, Østerby O (1994) Some numerical investigations of the stability of electrochemical digital simulation, particularly as affected by first-order homogeneous reactions. J Electroanal Chem 368:143–147
20. Britz D (1996) Brute force digital simulation. J Electroanal Chem 406:15–21
21. Georganopoulou DG, Caruana DJ, Strutwolf J, Williams DE (2000) Electron transfer mediated by glucose oxidase at the liquid/liquid interface. Faraday Discuss 116:109–118
22. Klymenko OV, Evans RG, Hardacre C, Svir IB, Compton RG (2004) Double potential step chronoamperometry at microdisk electrodes: simulating the case of unequal diffusion coefficients. J Electroanal Chem 571:211–221
23. Peaceman DW, Rachford HH (1955) The numerical solution of parabolic and elliptic differential equations. J Soc Ind Appl Math 3:28–41
24. Feldberg SW, Goldstein CI (1995) Examination of the behavior of the fully implicit finite-difference algorithm with the Richtmyer modification: behavior with an exponentially expanding time grid. J Electroanal Chem 397:1–10
25. Lavagnini I, Pastore P, Magno F, Amatore CA (1991) Performance of a numerical method based on the hopscotch algorithm and on an oblate spheroidal space coordinate- expanding time grid for simulation of voltammetric curves at an inlaid disk microelectrode. J Electroanal Chem 316:37–47
26. Svir IB, Golovenko VM (2001) Simulation of the microdisc problem in spherical coordinates. Application to electrogenerated chemiluminescence. Electrochem Commun 3:11–15
27. Svir IB, Oleinick AI (2001) The electrogenerated chemiluminescence kinetics at a microdisk electrode. J Electroanal Chem 499:30–38

28. Luskin M, Rannacher R (1982) On the smoothing properties of the Crank-Nicolson scheme. Appl Anal 14:117–135
29. Rannacher R (1982) Discretisation of the heat equation with singular initial data. Z Angew Math Mech 62:T346–T348
30. Rannacher R (1984) Finite element solution of diffusion problems with irregular data. Numer Math 43:309–327
31. Khaliq AQM, Wade BA (2001) On smoothing of the Crank-Nicolson scheme for nonhomogeneous parabolic problems. J Comput Methods Sci Eng 1:107–124
32. Østerby O (2003) Five ways of reducing the Crank-Nicolson oscillations. BIT Numer Math 43:811–822
33. Wood WL, Lewis RW (1975) A comparison of time marching schemes for the transient heat conduction equation. Int J Numer Methods Eng 9:679–689
34. Lindberg B (1971) On smoothing and extrapolation for the trapezoidal rule. BIT Numer Math 11:29–52
35. Bieniasz LK (2005) A singularity correction procedure for digital simulation of potential-step chronoamperometric transients in one-dimensional homogeneous reaction-diffusion systems. Electrochim Acta 50:3253–3261
36. Flyer N, Fornberg B (2003) Accurate numerical resolution of transients in initial-boundary value problems for the heat equation. J Comput Phys 184:526–539
37. Richtmyer RD (1957) Difference methods for initial-value problems. Interscience, New York
38. Mocak J, Feldberg SW (1994) The Richtmyer modification of the fully implicit finite difference algorithm for simulations of electrochemical problems. J Electroanal Chem 378:31–37
39. Britz D, Strutwolf J (2000) Higher-order spatial discretisations in electrochemical digital simulation. 1. Combination with the BDF algorithm. Comput Chem 24:673–684
40. Britz D, Strutwolf J, Thøgersen L (2001) Investigation of some starting protocols for BDF (FIRM) in electrochemical digital simulation. J Electroanal Chem 512:119–123
41. Liebmann G (1955) The solution of transient heat flow and heat transfer problems by relaxation. Brit J Appl Phys 6:129–135
42. Lawson JD, Morris JL (1978) The extrapolation of first order methods for parabolic partial differential equations. I. SIAM J Numer Anal 15:1212–1224
43. Gourlay AR, Morris JL (1981) Linear combinations of generalized Crank Nicolson schemes. IMA J Numer Anal 1:347–357
44. Strutwolf J, Schoeller WW (1997) Digital simulation of potential step experiments using the extrapolation method. Electroanalysis 9:1403–1408
45. Strutwolf J, Williams DE (1999) Digital simulation of two-dimensional mass transfer problems in electrochemistry using the extrapolation method. Electroanalysis 11:487–493
46. Strutwolf J, Britz D (2001) Use of high-order discretisations in digital simulation. 2. Combination with the extrapolation algorithm. Comput Chem 25:511–520
47. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
48. Deuflhard P (1985) Recent progress in extrapolation methods for ordinary differential equations. SIAM Rev 27:505–535
49. Twizell EH (1985) The extrapolation of implicit methods for the constant coefficient diffusion-convection equation. Commun Appl Numer Methods 1:129–135
50. Zhou H, Wu YJ, Tian W (2012) Extrapolation algorithm of compact ADI approximation for two-dimensional parabolic equation. Appl Math Comput 219:2875–2884
51. Nielsen MF, Almdal K, Hammerich O, Parker VD (1987) The application of Runge-Kutta integration in digital simulation of electroanalytical experiments. An accurate treatment of the homogeneous kinetics. Acta Chem Scand A 41:423–440
52. Britz D (1988) Electrochemical digital simulation by Runge-Kutta integration. J Electroanal Chem 240:17–26
53. Britz D (1988) Digital simulation in electrochemistry, 2nd edn. Springer, Berlin
54. Ružić I, Feldberg S (1974) The heterogeneous equivalent: a method for digital simulation of electrochemical systems with compact reaction layers. J Electroanal Chem 50:153–162

55. Ružić I (1983) Digital simulation of very rapid chemical reactions coupled by electrode reaction. An amended approach. J Electroanal Chem 144:433–436
56. Fisher AC, Compton RG (1992) The EC' mechanism: split waves at the channel electrode. Electroanal 4:311–315
57. Mastragostino M, Nadjo L, Saveant JM (1968) Disproportionation and ECE mechanisms. I. Theoretical analysis. Relationships for linear sweep voltammetry. Electrochim Acta 13:721–749
58. Lerke SA, Evans DH, Feldberg SW (1990) Digital simulation of the square scheme in cyclic voltammetry. A comparison of methods. J Electroanal Chem 296:299–315
59. DuFort EC, Frankel SP (1953) Stability conditions in the numerical treatment of parabolic differential equations. Math Tables Aids Comput 7:135–152
60. Feldberg SW (1990) A fast quasi-explicit finite difference method for simulating electrochemical phenomena. Part I. Application to cyclic voltammetric problems. J Electroanal Chem 290:49–65
61. Britz D, Kastening B (1974) On the electrochemical observation of a second-order decay of radicals generated by flash photolysis or pulse radiolysis. J Electroanal Chem 56:73–90
62. Atkins PW (1998) Physical chemistry, 6th edn. Oxford University Press, Oxford
63. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical recipes in Fortran. The art of scientific computing, 2nd edn. Cambridge University Press, Cambridge
64. Rudolph M (1991) A fast implicit finite difference algorithm for the digital simulation of electrochemical processes. J Electroanal Chem 314:13–22
65. Karlqvist O (1952) Numerical solution of elliptic difference equations by matrix methods. Tellus Q J Geophys 4:374–384
66. Byrne GD, Hindmarsh AC (1985) Experiments in numerical methods for a problem in combustion modeling. Appl Numer Math 1:29–57
67. Demmel JW (1995) Stability of block LU factorization. Numer Linear Algebra Appl 2:173–190
68. Golub GH, van Loan CF (1996) Matrix computations. John Hopkins University Press, Baltimore, London
69. Greenfield PF (1974) Recursive formulae for the solution of simultaneous parabolic differential equations. Simulation 22:152–154
70. Hindmarsh AC (1977) Solution of block-tridiagonal systems of linear algebraic equations. Technical Report Rept. UCID-30150, LLL
71. Isaacson E, Keller HB (1966) Analysis of numerical methods. Wiley, New York
72. Marsh F, Potter DE (1981) Recurrence solution of a block tridiagonal matrix equation with Neumann, Dirichlet, mixed or periodic boundary conditions. Comput Phys Commun 24:185–190
73. Varga RS (1962) Matrix iterative analysis. Prentice-Hall, Englewood Cliffs, NJ
74. Zhen H, Hongyuan F, Longjun S (1994) Pure alternating block explicit-implicit method for the diffusion equation in two space dimensions. Int J Comput Math 51:81–93
75. Aguilella VM, Garrido J, Mafé S, Pellicer J (1986) A finite-difference method for numerical solution of the steady-state Nernst-Planck equations with non-zero convection and electric current density. J Membr Sci 28:139–149
76. Bieniasz LK (2001) Extension of the Thomas algorithm to a class of algebraic linear equation systems involving quasi-block-tridiagonal matrices with isolated block-pentadiagonal rows, assuming variable block dimensions. Computing 67:269–285. Erratum: ibid. 70:275 (2003)
77. Fan D, White RE (1991) Modification of Newman's BAND(J) subroutine to multi-region systems containing interior boundaries: MBAND. J Electrochem Soc 138:1688–1691
78. Fletcher CAJ (1991) Computational techniques for fluid dynamics, vol I, 2nd edn. Springer, Berlin
79. Newman J (1966) Effect of ionic migration on limiting currents. Ind Eng Chem 5:525–529
80. Smyrl WH, Newman J (1968) Potentials of cells with liquid junctions. J Phys Chem 72:4660–4671

81. Newman J (1968) Numerical solution of coupled ordinary differential equations. Ind Eng Chem Fundam 7:514–517
82. Honeychurch MJ (2004–2006) Simulating electrochemical reactions with Mathematica. IBNH, St. Lucia, QLD
83. Stone HL (1968) Iterative solution of implicit approximations of multidimensional partial differential equations. SIAM J Numer Anal 5:530–558
84. Alden JA, Booth J, Compton RG, Dryfe RAW, Sanders GHW (1995) Diffusional mass transport to microband electrodes of practical geometries: a simulation study using the strongly implicit procedure. J Electroanal Chem 389:45–54
85. Alden JA, Compton RG (1996) Hydrodynamic voltammetry with channel microband electrodes: axial diffusion effects. J Electroanal Chem 404:27–35
86. Alden JA, Feldman MA, Hill E, Prieto F, Oyama M, Coles BA, Compton RG (1998) Channel microband electrode arrays for mechanistic electrochemistry. Two -dimensional voltammetry: transport-limited currents. Anal Chem 70:1707–1720
87. Krylov A (1931) On the numerical solution of equations by which the frequencies of small vibrations of material systems in technical problems are determined. Izv Akad Nauk SSSR VII:491–539. In Russian
88. Alden JA, Compton RG (1997) A general method for electrochemical simulations. 2. Application to the simulation of steady-state currents at microdisk electrodes: homogeneous and heterogeneous kinetics. J Phys Chem B 101:9606–9616
89. Alden JA, Hutchinson F, Compton RG (1997) Can cyclic voltammetry at microdisc electrodes be approximately described by one-dimensional diffusion? J Phys Chem B 101:949–958
90. Bard AJ, Denuault G, Friesner RA, Dornblaser BC, Tuckerman LS (1991) Scanning electrochemical microscopy: Theory and application of the transient (chronoamperometric) SECM response. Anal Chem 63:1282–1288
91. Welford PJ, Brookes BA, Climent V, Compton RG (2001) The hanging meniscus contact: geometry induced diffusional overpotential. The reduction of oxygen in dimethylsulphoxide at Au(111). J Electroanal Chem 513:8–15
92. Wesseling P (1992) An introduction to multigrid methods. Wiley, New York
93. Engeln-Müllges G, Niederdrenk K, Wodicka R (2011) Numerik-Algorithmen; Verfahren, Beispiele, Anwendungen, 10th edn. Springer, Heidelberg

# Chapter 9
# Other Methods

In previous chapters, those methods that are regarded as most advisable in some sense are presented in some detail. The explicit method cannot really be said to be advisable, but it does serve as an introduction to simulation, from which one can advance to the slightly harder implicit methods mentioned in Chap. 8. In the present chapter, a large number of alternative schemes that have been advocated in the last several decades are at least mentioned, some in more detail than others, according to the present authors' estimation of the feasibility of the methods' use, or the ability of the average electrochemist to program them. This is inevitably a subjective judgement and there will be some disagreement. References are provided for the reader who wants to delve more deeply.

## 9.1 The Box Method

The Feldberg approach to digital simulation [1] uses a somewhat different method of discretisation, and the method is alive and well [2]. It begins with Fick's first diffusion equation, using fluxes between boxes or finite volumes, rather than concentrations at points in the discretisation process (see below).

Rather than, as is done in this book, sampling concentration along the $x$-axis at a number of points, Feldberg thought in terms of boxes along the axis. Initially, the boxes were of equal length but both Seeber and Stefani [3] and Feldberg [4] proposed in 1981 that boxes of unequal length were better, and suggested exponentially expanding box-lengths. A few such boxes are shown in Fig. 9.1. We have selected three boxes, consecutively numbered, as shown. The middle box is indexed with $i$, and is bounded by the positions $x_{i-1}$ and $x_i$. Its length is $h_i$. The formula for the expansion can be expressed as follows. We start with a first box of length $h_1$, chosen suitably (with perhaps a homogeneous chemical reaction in

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_9

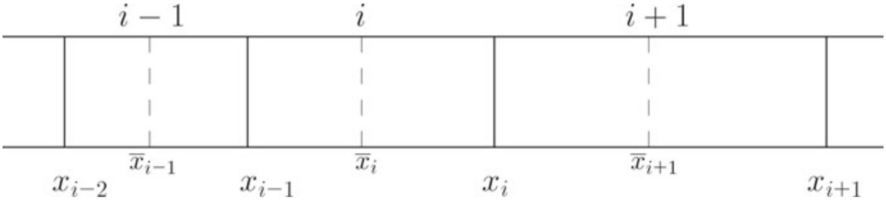


**Fig. 9.1** Discrete boxes along *x*

mind, so that $h_1$ lies well within the reaction layer). Then, each successive box has a length a fixed multiple $> 1$ of the length of the one before it. This is in fact precisely the same as is done in the exponentially spaced point positions, described in Chap. 7. There, the symbol $\gamma$ is given to the expansion factor. Tradition has it, in the box method approach, to use a different symbol and definition for the expansion factor; our $\gamma$ is equivalent to $\exp(\beta)$ in the box terminology. We can thus directly describe the outer box boundaries in the same terms as in (7.19), and arrive at Feldberg's equation,

$$x_i = h_1 \frac{\exp(i\beta) - 1}{\exp(\beta) - 1} \,. \tag{9.1}$$

This is the formula in [4], seen again in Rudolph's chapter in [5]. Also, the box lengths themselves are given by

$$h_i = h_1 \exp((i-1)\beta) \,. \tag{9.2}$$

This expanding box strategy is mathematically equivalent to the transformation from *X* into *Y* as described for point positions in Chap. 7, Eq. (7.3), as is shown in Appendix C. Its implementation in the discretisation process is however different.

The way this is used is as follows. Fick's first law is used, and fluxes into and out of box *i* are considered. For this, we need to assign distances between successive boxes, and here a small difficulty arises. For boxes of equal length, the distance is simply that length, stretching from box midpoint to the next box midpoint. With boxes of unequal lengths, this leads to inaccuracies. What is done instead is to (mentally) map the position *x* onto an index space, the *i*'s in Fig. 9.1. These have equal intervals of size unity. The assigned midpoint of a given box indexed with *i* is then at $i - \frac{1}{2}$, and this transforms to the midpoint positions marked in Fig. 9.1 as $\overline{x}_{i-1} \ldots \overline{x}_{i+1}$, at the dashed lines. These are not the midpoints of the boxes, that is, not $(x_i + x_{i+1})/2$, etc. Distances between boxes are then taken as the distances between these points. The points are given by

$$\overline{x}_i = h_1 \frac{\exp((i-\frac{1}{2})\beta) - 1}{\exp(\beta) - 1} \,. \tag{9.3}$$

The flux $f_1$ going into box $i$ from the left is

$$f_1 = -AD\frac{c_i - c_{i-1}}{\overline{x}_i - \overline{x}_{i-1}} \tag{9.4}$$

where $A$ is the cross-sectional area of the box and $D$ is the diffusion coefficient. The flux $f_2$ going out of the $i$th box to the right is

$$f_2 = -AD\frac{c_{i+1} - c_i}{\overline{x}_{i+1} - \overline{x}_i} \ . \tag{9.5}$$

The resultant flux into box $i$ is the difference between the two,

$$f = f_1 - f_2 \tag{9.6}$$

which has units of moles per second. We want concentration changes, so we must multiply by the time interval to get moles, and divide by the box volume to get $\delta c_i$. The time interval is $\delta t$ and the box volume $V_i$ is

$$V_i = Ah_i \ . \tag{9.7}$$

All this leads to the equation for the change in $c_i$,

$$\delta c_i = \frac{D\delta t}{h_i}\left(\frac{c_{i+1} - c_i}{\overline{x}_{i+1} - \overline{x}_i} - \frac{c_i - c_{i-1}}{\overline{x}_i - \overline{x}_{i-1}}\right) \tag{9.8}$$

which, using Feldberg's [4] and Rudolph's [5] notation, is now expressed in the form

$$\delta c_i = D^*_{2i}(c_{i+1} - c_i) - D^*_{1i}(c_i - c_{i-1}) \ . \tag{9.9}$$

The $D^*$ coefficients can be worked out from (9.8), substituting for the $\overline{x}$ terms using (9.3). The denominators in the two terms in brackets on the right-hand side of (9.8) can be simplified. As an example, consider the first of these. It is simplified in the following manner. From (9.3),

$$\overline{x}_{i+1} - \overline{x}_i = h_1\frac{\exp((i + \frac{1}{2})\beta) - \exp((i - \frac{1}{2})\beta)}{\exp(\beta) - 1} \tag{9.10}$$

and dividing top and bottom of this fraction by $\exp((i - \frac{1}{2})\beta)$, we are left with

$$\overline{x}_{i+1} - \overline{x}_i = h_1 \exp((i - \tfrac{1}{2})\beta) \ . \tag{9.11}$$

The second denominator term can be simplified in an analogous manner, dividing by $\exp((i-\frac{3}{2})\beta)$. This leads to the expressions, for $i > 1$,

$$\begin{aligned} D^*_{2i} &= D^* \exp(2\beta(\tfrac{3}{4} - i) \\ D^*_{1i} &= D^* \exp(2\beta(\tfrac{5}{4} - i) \end{aligned} \tag{9.12}$$

with

$$D^* = \frac{D\delta t}{h_1^2} \, . \tag{9.13}$$

For the very first box ($i = 1$) there is a small problem. There is no box at $i = 0$, and the first box extends to $x_1 = h_1$, and from (9.3) ($i = 1$),

$$\bar{x}_1 = h_1 \frac{\exp(\frac{1}{2}\beta) - 1}{\exp(\beta) - 1} \, , \tag{9.14}$$

which is the assigned distance of this box from the electrode. Inserting this appropriately into (9.8) results in the two coefficients

$$\begin{aligned} D^*_{21} &= D^* \exp(-\tfrac{1}{2}\beta) \\ D^*_{11} &= D^* \frac{\exp(\beta) - 1}{\exp(\frac{1}{2}\beta) - 1} \, . \end{aligned} \tag{9.15}$$

The above equations are all given in [4] and [5].

Consideration of (9.9) reveals that it is of the same form as that shown for the point method using arbitrarily spaced points, Eq. (8.8) in Chap. 8. We can proceed from here in the same way as in that chapter. That is, all methods described there (or even the explicit method) can be applied. By dividing (9.9) by $\delta t$, we can even go into an Method of Lines (MOL)-type method (see below). There is thus no need to describe the procedure further from here.

As a final word on the box method, it should be mentioned that in his publications on unequal intervals, Rudolph [6–8] makes a strong case for the box method. His initial aim in these papers was to show that discretisation in the point method, on equal intervals in the transformed space (as described in Chap. 7) is not as accurate as had been supposed. Rudolph devised an improved way of discretising the transformed diffusion equation[7, 9] (the same as the present (7.11) in Chap. 7, derived by Bieniasz [10]), and states that the box method with exponentially expanding intervals, as described above, is as accurate as when using this improved formula. It seems that the use of fluxes is the cause of the accuracy of the box method, even though computed concentration values might be less accurate. Rudolph refers [8] to exponential convergence of calculated flux values, using this method. This is supported by the literature on the control volume method [11]. Patankar [11] writes that “even coarse-grid solution exhibits *exact* integral balances”. The control volume method has been used in two other electrochemical works [12, 13]. There is as yet no agreement on this, and further study is needed. Certainly Rudolph’s 2004 paper [8], showing very rapid exponential convergence of the computed flux, makes a strong case for this method.

## 9.2 Improvements on Standard Methods

Both the explicit and implicit methods already described have been improved to greater accuracy and, hopefully, greater efficiency.

### *9.2.1 The Kimble and White Method*

Kimble and White [14] developed a scheme which, as described and intended, was somewhat awkward to use and limited the possible number of points in time and space. The method is mentioned in other chapters for its use as a high-order start for BDF (for which it did indeed work, but not with great efficiency). It is perhaps best described in two stages. Consider Fig. 9.2, a modest-sized grid on which the KW method is to be used, representing positions in time (indices *j*) and space (indices *i*). The thicker bottom line represents initial conditions; the dotted line at the left is that

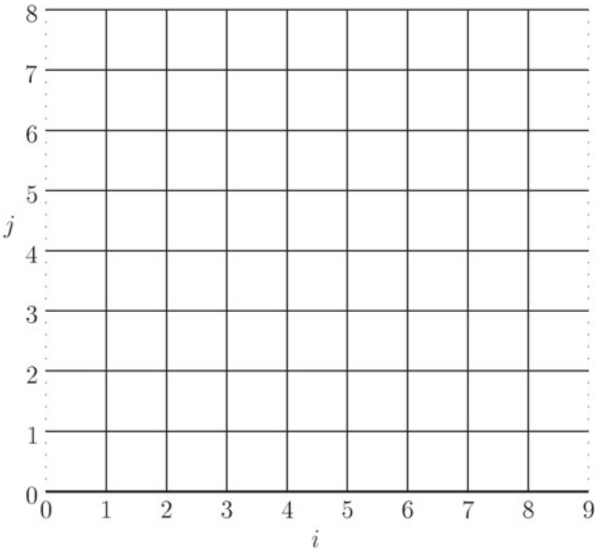
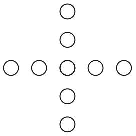


**Fig. 9.2** An example grid for the KW method

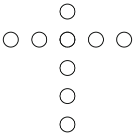
for the boundary $C_0$ values, that at the right the bulk values. The vertical line at $i = 8$ lies at $X_{max}$. In general, let there be $N + 2$ mesh points in the horizontal, $X$-direction and $M + 1$ in the vertical, $T$-direction, that is, $N \times M$ points to be calculated aside from boundary points.

Nguygen and White [15] used such a grid, albeit for the purpose of solving an elliptic problem, not involving time, so that the vertical axis was along $y$, the other spatial dimension. The method therefore involved second spatial differences in both directions, and they used three-point discretisations. What made their approach special is that, rather than writing one large system of equations for all $N \times M$ unknowns, which leads to a banded system, they wrote a system of matrix equations, each unknown being the whole horizontal vector. This gave them a block tridiagonal system, solvable by the available routine BANDJ by Newman [16].

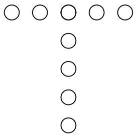
This early paper was followed by another one in 1990 by Kimble and White [14], now applying the method to a diffusion problem, and using 5-point approximations in both directions. As before, the problem was cast into a block matrix, but because of the 5 points used for the discretisations, this was block-pentadiagonal. For most node points in the figure, the 5-point approximations yield the following computational molecule or stencil.



All points have been drawn empty, indicating what is special about the KW method. If the stencil had been drawn, as one might expect, with all points filled (known) except the top one, this would indicate that the method marches forward in time, using an explicit central differences form. Such central difference forms are all known to be unstable. The classic one is the 3-point leap-frog scheme of Richardson [17], which appears attractive intuitively, being second-order in time, but was proved unconditionally unstable in 1950 [18]. The same holds for central difference schemes using a larger number of points, as here. Neither does it make a difference simply to put all the equations into one large system; the instability still appears. What made the difference here, as shown in [19], is the device the authors employed at the top of the grid. The 5-point temporal discretisation can only be used up to line $M - 2$ (the top one being at index $M$). For index $M - 1$, an asymmetric backward form is needed, using points at indices $M - 4 \ldots M$ (form $y'_4(5)$ in Table A.1, Appendix A):



and for the top line at index $M$, a BDF form, $y'_5(5)$ in the Table, was used. It is this "cap" on the whole system that stabilises it [19].



Similarly, asymmetric forward forms are used at the bottom end.

Kimble and White were aware that leapfrog methods are unstable and simply remark that this did not seem to apply to their method. Also, they mention the use of five points for all approximations but their table of discretisations shows that they used six points at the edges for the spatial second derivative. This is no doubt because, as Collatz already mentions in 1960 [20], the asymmetric five-point second derivative is only third-order, while a six-point formula is fourth-order, like the symmetrical five-point ones used in the bulk of the grid. So, for the second spatial derivative at index $i = 1$, the form $y''_2(6)$ was used, and the reverse, form $y''_5(6)$ at $i = N$.

All this leads to a block-pentadiagonal system of equations in the unknown vectors representing the horizontal lines in the figure. Results appeared to be good [14] but clearly, the drawback of the method is that, for any reasonably sized grid, the system becomes large. Probably for this reason, the method has not taken on. It might, however, have application in the *ode* field, where the computational molecule reduces, as it were, to a single vertical column.

Another potential use for the KW method is as a high-order start for the BDF method, as indeed suggested by Feldberg and Goldstein [21], who dubbed this the "hyperimplicit" approach. As described in previous Chaps. 4 and 8, BDF has the problem of requiring starting values. If the grid in Fig. 9.2 is reduced to $k$ time levels for a $k$-point BDF variant, and solved for the $k - 1$ unknown levels, high-order starting values result. This process is described in Chap. 8, Sect. 8.5.2.

### 9.2.2 *Multi-Point Second Spatial Derivatives*

When using methods such as extrapolation or BDF, which are capable of high order results with respect to the time intervals, one finds that, going to orders higher than $O(\delta T^2)$ does not lead to improvements, certainly not to greater efficiency—rather the reverse, because more computing is done to achieve similar accuracies. The reason is that the error is a sum of terms involving $\delta T^p$ ($p$ being the particular method's order with respect to $\delta T$) and $H^2$ (for equal intervals), $H$ being the spatial interval. This comes from the three-point spatial second derivative usually used. This term soon dominates and renders high-order time schemes useless. Thus, higher-order second spatial derivatives might help and have been studied for equal intervals [22–25] as well as for unequal intervals [26]. Inspiration for this came from the KW method, described above. As with that work, six-point asymmetric discretisation becomes desirable at points next to the boundaries, in order for all discretisations to be fourth-order with respect to the spatial interval $H$. The equations then are in semidiscretised form (that is, leaving the left-hand side of the diffusion equation untouched for the moment)

$$\begin{aligned}
\frac{\mathrm{d}C_1}{\mathrm{d}T} &= \frac{1}{12H^2}\left(10C_0 - 15C_1 - 4C_2 + 14C_3 - 6C_4 + C_5\right)\\
\frac{\mathrm{d}C_2}{\mathrm{d}T} &= \frac{1}{12H^2}\left(-C_0 + 16C_1 - 30C_2 + 16C_3 - C_4\right)\\
&\dots\\
\frac{\mathrm{d}C_i}{\mathrm{d}T} &= \frac{1}{12H^2}\left(-C_{i-2} + 16C_{i-1} - 30C_i + 16C_{i+1} - C_{i+2}\right)\\
&\dots\\
\frac{\mathrm{d}C_{N-1}}{\mathrm{d}T} &= \frac{1}{12H^2}\left(-C_{N-3} + 16C_{N-2} - 30C_{N-1} + 16C_N - C_{N+1}\right)\\
\frac{\mathrm{d}C_N}{\mathrm{d}T} &= \frac{1}{12H^2}\left(C_{N-4} - 6C_{N-3} + 14C_{N-2} - 4C_{N-1} - 15C_N + 10C_{N+1}\right)
\end{aligned} \tag{9.16}$$

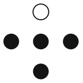
where the first line (index 1) uses the $y_2''(6)$ form in Table A.2, and that at index $N$ the symmetrically opposite form $y_5''(6)$. All other equations, indices $2 \dots N-1$, use the symmetrical five-point form $y_3''(5)$. There is, however, something special about lines 2 and $N-1$, in that, like the first and last lines, they include terms in boundary values. In practice, the $C_0$ values would be substituted by suitable expressions involving unknown concentration terms $C_1, \dots,$ according to what the boundary conditions are (Chap. 6). Also, the system (9.16), being pentadiagonal, requires something more complicated than the Thomas algorithm, and one has been described [22], based on such texts as that of Engeln-Müllges and Uhlig [27] and Fletcher [28]. It involves several sweeps and, depending on the boundary condition expressions, possibly some preliminary eliminations to reduce the matrix to pentadiagonal form, if these expressions produce some extra-long equations (typically, the first and second).

This was examined in a series of works, using BDF [22], extrapolation [22], RK (see below) [25] and BDF with the KW start [24]. The latter did produce highly accurate starting vectors but due to the high computational overhead, was found less efficient than some less accurate BDF starts. Overall, the most efficient methods were those employing fourth-order extrapolation, followed closely (and surprisingly) by the simple BDF start with the time correction [25], mentioned in Sect. 4.8.1 on page 69, and Sect. 8.5.2, page 159.

In practice, the (6,5) approach is, at present, limited by the fact that, in the form presented here, it applies to equal intervals. A slight improvement with unequal intervals, using a 4-point spatial second derivative, is described in Chap. 8, and this might be sufficient improvement, at little cost in terms of desk work [29]. It has been applied to the ultramicroelectrode [30] and for these, multi-point forms have been studied in detail [31], as is also mentioned again in Chap. 12.

### *9.2.3 DuFort–Frankel*

The (*ode*-) method called leapfrog has been mentioned in Chap. 4, where Eq. (4.38) describes it. This was used by Richardson [17] to solve a parabolic *pde*, apparently with success. The computational molecule corresponding to this method is



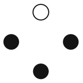
In this scheme, the temporal derivative is formed by the central (second-order) difference between the upper and lower points, the second spatial derivative being approximated as usual. This makes the discretisation at the index *i* in space,

$$\frac{C'_i - {}'C_i}{2\delta T} = \frac{1}{H^2}\left(C_{i-1} - 2C_i + C_{i+1}\right) \tag{9.17}$$

(adhering to the notation used for BDF as in Chap. 8, page 160, where ${}'C_i$ denotes $C_i$ at time $T - \delta T$). The scheme is clearly explicit.

Leapfrog is used with apparent success to solve hyperbolic *pde*s [32], but was proved unconditionally unstable for parabolic *pde*s in 1950 [18]. Richardson had been lucky, in that the instabilities had not made themselves felt in his (pencil and paper) calculations, in the course of the few iterations he worked.

DuFort and Frankel [33] devised a modification to this scheme in 1953 that stabilises it:



The time derivative is the same central difference but the spatial second derivative now leaves out the central point, substituting for it the mean of the past and future points. Thus, the discretisation is

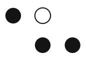
$$\frac{C'_i - {}'C_i}{2\delta T} = \frac{1}{H^2}\left(C_{i-1} - {}'C_i - C'_i + C_{i+1}\right) \tag{9.18}$$

which is still explicit for $C'_i$ when rearranged but known to be stable for all $\delta T/H^2$ [34]. This formula received some attention among electrochemists for some time [35–37], some [36] calling it "FQEFD" (fast quasi-explicit finite difference). It shares with BDF (or FIRM) the problem of start-up, since at the first step, a row of values at $T = -\delta T$ are needed. This was mentioned by Marques da Silva et al. [38], who studied this scheme, along with that of Saul'yev (see below) and its offshoots, and hopscotch (see also below). They also mention another problem with DuFort–Frankel, shared with hopscotch, as pointed out by Feldberg [39]. Both DuFort–Frankel and hopscotch, being stable for any $\lambda$, invite the use of large $\lambda$ or $\delta T$ values. This should result in a fast propagation of changes in the concentration profile from what is happening at the electrode. However, because these schemes are explicit, generating one new value at a time only from old values, such changes can only advance into the cell's interior by one spatial interval at a time. This compares unfavourably with implicit methods for which the whole profile is always calculated together at each step. For this reason, Feldberg [39] writes of the "propagational inadequacy" of hopscotch, and this was also noted by Marques da Silva et al. (private communication), also of the DuFort–Frankel scheme [38]. Both schemes thus perform less and less well at large $\delta T$, nullifying the advantages that might have come from the unconditional stability. This problem had been pointed out in the numerical text of Carnahan et al. [40, p. 440, Fig. 7.5] for the explicit method itself. Also, most textbooks mention the inconsistency problem of DuFort–Frankel in the case when $\delta T/H \gg 0$; in fact, the authors themselves mention this in their 1953 paper [33], as did Potter [41]. More about this can be read in Chap. 15.

The DuFort–Frankel scheme has apparently been dropped in favour of more interesting schemes such as BDF, which can be driven to higher-orders, and for which the start-up problem has been overcome (Chap. 4).

### *9.2.4 Saul'yev*

A perhaps more interesting method is that of Saul'yev [42] (and apparently independently, the same idea, of Barakat [43] a short time later). The method is explicit, which makes programming easier than implicit methods, and is capable of improvements over the original idea. There are two basic variants that make up the building blocks for improvements. The **LR** variant, as the name implies moves from left (that is, from $X = 0$) to right (higher $X$), generating new values at the next time level. The computational molecule for this is



and the diffusion equation is discretised from the four points in the form

$$\frac{C'_i - C_i}{\delta T} = \frac{1}{H^2}\left(C'_{i-1} - C'_i - C_i + C_{i+1}\right) \tag{9.19}$$

which is seen to be a sort of tilted second derivative on the right-hand side. The left-right progression is explicitly possible because the left-most element has already been computed in the previous step. The equation is rearranged into a form explicit for $C'_i$. Obviously, this leaves the problem of how to start, for which the boundary value $C'_0$ is needed. This will be described below. The above equation can be expressed in the form, explicit for $C'_i$,

$$C'_i = a_1(C'_{i-1} + C_{i+1}) + a_2 C_i \tag{9.20}$$

with the constants defined as

$$a_1 = \frac{\lambda}{1+\lambda} \tag{9.21}$$

and

$$a_2 = \frac{1-\lambda}{1+\lambda}\ . \tag{9.22}$$

The other variant is **RL**, moving from right to left:

○ ●

● ●

and the discretisation is

$$\frac{C'_i - C_i}{\delta T} = \frac{1}{H^2}\left(C_{i-1} - C_i - C'_i + C'_{i+1}\right)\ . \tag{9.23}$$

When rearranged so as to be explicit for $C'_i$, it becomes

$$C'_i = a_1(C_{i-1} + C'_{i+1}) + a_2 C_i \tag{9.24}$$

with the same definitions of the constants. The only difference is in the superscripts of the first term on the right-hand side, and in the order of evaluation, here from right to left.

It remains to describe how to handle the boundary value $C'_0$. Clearly, for the RL variant, there is no problem because the last concentration value calculated is $C'_1$, and $C'_0$ can then be computed from all the other $C'$ values, now known, according to the boundary condition. This leaves the LR problem. If the boundary concentration is determined as such (the Dirichlet condition, for example the Cottrell experiment), then this is simply applied. It is with derivative (Neumann) boundary conditions that there is a (small) problem. Here, we know an expression for the gradient $G$ at the electrode. For simplicity, assume a two-point gradient approximation at time $T + \delta T$ ($G'$),

$$G' = \frac{C'_1 - C'_0}{H} \tag{9.25}$$

and this can be coupled with the first LR expression for $C'_1$ from (9.20), setting $i = 1$, and the two equations solved for $C'_0$ (and $C'_1$). The LR process can then begin. If more points are to be used for the gradient approximation, then more LR expressions must also be added, and a correspondingly larger system of equations needs to be solved. This has been described [44] in some detail.

This is as much as will be said here about the mechanics of the Saul'yev method; the reader can take it from here, as it is quite simple. Some further remarks are however in order.

Both the LR and RL variants, despite being explicit, are said to be stable for all $\lambda$ values, which is a great advantage. Also, the method does not share with DuFort–Frankel and hopscotch the propagational inadequacy problem [39] mentioned above because both variants amount to a recursive algorithm, each newly calculated element carrying with it some component from all previously calculated elements.

There are drawbacks, however. It is clear from the above computational molecules that the second, spatial derivative is approximated in an asymmetric manner, and although these approximations are in fact second-order with respect to the interval $H$, they are not as good as, say, the Crank–Nicolson ones. Both LR and RL, taken by themselves, do not produce very good results. It was not long after Saul'yev's book in 1964 that Larkin (in the same year) published some extensions, as did other workers [45–47]. The asymmetry of each of the two variants suggests combining them in some manner. Larkin [48] listed four strategies:

1. use the LR variant only;
2. use the RL variant only;
3. use the LR and RL variants alternately at each iteration;
4. use the LR and RL variants independently at each iteration and average the result.

Liu [46, 47] later added a modification, using one extra point at the bottom advancing end of the molecules shown above, and showed that this made the schemes more accurate and that they were still stable. Evans and Abdullah [45] developed what they called group explicit methods (GEM) based on Saul'yev, in which the LR and RL schemes were combined in larger computational molecules.

Electrochemists first investigated the Saul'yev method in 1988 and 1989 [38, 49], including GEM, and the incorporation of implicit boundary values was added later [44]. The result of these studies is broadly that the last of Larkin's options above, averaging LR and RL, is the best. This has about the same accuracy as Crank–Nicolson, and could be considered to be easier to program. The third option, alternating LR with RL, produces oscillations.

The stability of the Saul'yev schemes in the electrochemical context with mixed boundary conditions was examined [50, 51]. Surprisingly, it was found that the LR variant can be unstable with mixed boundary conditions. There exists, for any number $N$ of intervals in space, a maximum $\lambda$ value in the discrete equation, above which the LR scheme becomes unstable. Fortunately, it is rather difficult to attain this condition in practice. Since these studies, Deng [52] has used the various Saul'yev schemes and offshoots, and cites several Chinese studies also using Saul'yev variants, but they have found little application elsewhere.

### *9.2.5 Hopscotch*

In 1965, Gordon [53] reported some studies of what he called nonsymmetric difference equations, meaning schemes like that of Saul'yev and the Peaceman-Rachford ADI scheme (see Chap. 12), in which not all points are treated alike. One of his new ideas was what he called the "explicit–implicit" scheme. It is as follows (using the simple example of a 1-D simulation). As usual, we move along time with index $j$ and along space ($X$) with index $i$, starting with $j = 1$, having set the initial values for $j = 0$. The $X$ points are indexed from zero to $N + 1$, with $X_0 = 0$ lying at the electrode and $X_{N+1}$ lying in the bulk, outside the diffusion space.

If $j$ is even, then we first explicitly compute new points for all odd $i$, that is,

$$C'_i = C_i + \lambda(C_{i-1} - 2C_i + C_{i+1}) \, , \tag{9.26}$$

and then using the implicit formula (the same as backward implicit BI) on all points with even $i$,

$$C'_i = C_i + \lambda(C'_{i-1} - 2C'_i + C'_{i+1}) \, . \tag{9.27}$$

The interesting thing here is that in contrast with BI, the values for $C'_{i-1}$ and $C'_{i+1}$ are already known from the run of (9.26), lying at odd values of $i$, so that (9.27) can be rearranged explicitly for $C'_i$. At the next iteration, $j$ will be odd, and the explicit calculation is done on all even $i$, followed by the implicit calculation on all points with odd $i$. In this way, alternating the sets being computed explicitly and implicitly, a certain symmetry is produced.

Gordon also showed that the scheme is convergent and stable for all $\lambda$. The scheme was taken up by Gourlay in 1970 [54], who tightened up the mathematical notation, and applied the scheme to 2-D numeric problems, as well as introducing the trick of overwriting values in the first (explicit) step, so that only one array of values is needed. Gourlay coined the name "hopscotch" for this method, by which it has been known since then, and usually only Gourlay is cited. There were follow-up papers [55, 56]. It also became clear that the method was closely related to others like ADI [54] (to be mentioned for 2-D).

The enthusiasm for hopscotch arose from the fact that here was a method with an accuracy thought to be almost comparable with that of Crank–Nicolson, but which was an explicit computation at every step, not requiring the solution of linear systems of equations, as other implicit methods do. It was also stable for all $\lambda$, thus making it possible to use larger time steps, for example. The convenience of the point-by-point calculation has occasionally led workers to call the method "fast" [57].

Shoup and Szabo [58–61] brought the method to electrochemistry, using it to simulate diffusion at a microdisk electrode. This was a problem at the time. A proper implicit scheme leads to rather large banded systems of equations (see Chap. 12), and workers tended to use ADI (and still do so), which leads to (much smaller) tridiagonal systems of equations. Hopscotch seemed to be the answer, as one could recalculate all points explicitly, and use large $\lambda$ values (in both directions). Feldberg [57] used the method to simulate processes at a rotating ring-disk electrode, citing stability and ease of use. Other electrochemists followed [62–69]. There soon appeared criticisms, however. Ruzić [70] commented that Shoup and Szabo had misrepresented the normal explicit method by stating that it required two concentration arrays, and showed how this could easily be avoided by using two scalar concentration values trailing behind the values treated, while overwriting all values as they are calculated (this is the trio of variables, `C1`, `C2`, `C3`, used in the example program `COTT_EX`, see Appendix E). Also, Ruzić showed that some simple known improvements [71] to Feldberg's explicit method improved its accuracy to something close to that of the hopscotch method, so that the latter was not needed. Shoup and Szabo had indeed shown in their 1984 paper [59] that hopscotch's accuracy declines badly at $\lambda$ values exceeding unity, so the ability to use large $\lambda$ cannot be cited as an advantage of the method. Ruzić's polemic was rebutted by Shoup and Szabo [72], who admitted some of the points made but then launched into a discussion on the precise implementation of the Feldberg (box-) method which, unlike the point method, allows a number of interpretations and tricks to improve the results. The improvement described by Ruzić, shown in his example program and based on Sandifer and Buck [71], amounts to the use of the point method.

In 1987, Feldberg [39] pointed out the most serious drawback of hopscotch. The problem is that, at each step forward in time, application of the two Eqs. (9.26) and (9.27) can propagate a perturbation at a given point in the profile (for example, at the electrode) only by a single interval in space. If large time intervals are used, then one would expect such changes to make themselves felt over a number of neighbouring points, but hopscotch cannot do this. Feldberg writes of the "propagational inadequacy" of the hopscotch method. As mentioned above, it shares this with the DuFort–Frankel method and also with the explicit method. With the latter, however, the stability limit on $\lambda$ prevents the use of time intervals large enough for this inadequacy to matter, while for hopscotch (and DuFort–Frankel) there is the possibility and temptation to use larger time intervals. At values that can be used in the explicit method, hopscotch is only marginally better than explicit and this, together with the propagational inadequacy feature, suggests that hopscotch is not a method of choice, despite the ease of programming, both for one- and two-dimensional simulations. Its use has declined to almost zero in recent years; we are aware of only a single recent electrochemical work using hopscotch [73].

### 9.2.6 Runge–Kutta

The RK variants are described for *ode*s in Chap. 4, from page 64 and, for a system of *ode*s, from page 77. There, only the Euler method is detailed, but in terms of RK terminology, from which the higher-order variants follow easily. The description there will not be expanded here, for reasons given below. When solving a *pde* or a system of such, one way is to "semidiscretise" the equation(s), meaning that only the right-hand side is discretised, leaving the time derivative as it is. This yields a system of *ode*s, such as (4.49) on page 77 and one can then proceed with that. This is called the MOL. One can either treat the boundary conditions separately, or add them to the system, in which case the system becomes a Differential Algebraic Equations (**DAE**) system, and requires other methods to solve for it, as briefly mentioned in Chap. 4.

RK initially attracted attention in electrochemical digital simulation because of homogeneous chemical reactions. With explicit simulations, it was realised that there was a problem if the term $K\delta T$ was of appreciable magnitude[74, 75]. Nielsen et al. [76] pointed out that, if this term causes more than a few percent change in a concentration upon a single step forward in time, the simulation will be inaccurate. Early on it was suggested to treat the chemical term more accurately. Feldberg and Auerbach [74] used the known analytical solutions for a first- and second-order chemical reaction for the chemical term, and Flanagan and Marcoux [75] followed, suggesting RK integration for those cases in which analytical solutions are not known. The RK method was then used by Nielsen et al. [76].

It was realised then that the method of Nielsen et al. [76] had a defect, limiting its accuracy. The diffusional and chemical terms were calculated separately, in sequence. That is, first diffusional changes are applied to the concentrations, and then the chemical reaction is allowed to run, on the changed concentrations. This is the "sequential method" also used for the plain explicit (Euler) method for both terms, where it has been shown to be consistent mathematically [77]; see also Appendix C. No proof of the consistency of the method, when RK is applied to the chemical terms, is known, however. Clearly, this technique uncouples the two processes taking place, diffusion and chemical reaction. This was remedied [78] in a work where RK was applied to the whole system of equations, thus taking care of the coupled nature of the two processes. It was found [78] that using RK2, a modest efficiency gain of about a factor 3, in terms of computer time used, was achieved, compared with the plain explicit method, in order to reach a given target accuracy in model simulations. This is not very much and the method has the additional drawback of a limit on the size of $\lambda$, the same as the explicit method, 0.5. Nevertheless, this whole-system RK method has seen some use since then, notably by the Lemos school [79–83], who emphasise the MOL nature of their approach, Gosser [84], Barker [85] and Hayase et al. [86]. Accuracy contours were computed for the method, among others [87] and a stability analysis was published [88], as affected by the chemical reaction (the reaction lowers the limit on $\lambda$). In the course of an investigation of higher-order discretisations of the spatial second derivative, RK was once again tested [25] and once again found not especially promising; using 5-point discretisation, the limit on $\lambda$ decreases to 0.375.

It is therefore concluded that this method, using explicit RK as described in Chap. 4, is not worthwhile mainly because of the $\lambda$ limitation.

There are, however, implicit variants of RK, and these may have promise. There are several classes of these, see a thorough text on the subject [89, 90]. One of these classes, the Rosenbrock method, has been examined in the electrochemical context [91, 92, and see the references therein] and found very efficient. This is described in its own Sect. 9.4, below.

### 9.2.7 *Hermitian Methods*

Kopal [93] describes Hermitian interpolation, as used by Hermite. The essence of this is that not only function values at grid points are used, but also derivative values. For a given number of grid points used in a particular approximation formula, this results in a higher-order accuracy with respect to the grid intervals. Although Hermite used this only for interpolation, the term is now used more generally, referring to the characteristics mentioned above. Three Hermitian methods have been used in electrochemical simulations, up to the time of writing, and two of them are due to Bieniasz.

#### 9.2.7.1 Numerov/Douglas

In 1924, the Russian astronomer Numerov (transliterating his own name as Noumerov), published a paper [94] in which he described some improvements in approximations to derivatives, to help with numerical simulations of the movement of bodies in the solar system. His device has been adapted to the solution of *pde*s, and was introduced to electrochemistry by Bieniasz in 2003 [10]. The method described by Bieniasz is also called the Douglas equation in some texts such as that of Smith [95], where a rather clear description of the method is found. With the help of the Numerov method, it is possible to attain fourth order accuracy in the spatial second derivative, while using only the usual three points. The first paper by Bieniasz on this method treated equally spaced grids, and it was followed by another on unequally spaced grids [96]. The method makes it practical to use higher-order time derivative approximations without the complications of, say, the (6,5)-point scheme described above, which makes the solution of the system of equations a little complicated (and computer-time consuming).

The description in Smith [95, pp. 137-] is followed here. It starts with a statement that a second derivative can be approximated by

$$\frac{\partial^2 u}{\partial x^2} \approx \left(\delta_x^2 u - \tfrac{1}{12}\delta_x^4 u + \tfrac{1}{90}\delta_x^6 u - \dots\right) , \tag{9.28}$$

where the symbol $\delta_x^n u$ denotes the operation $\delta_x^n$ on $u$ corresponding to an approximation to the $n$th derivative with respect to $x$. In particular, we have

$$\delta_x^2 u = u_{i-1} - 2u_i + u_{i+1} , \tag{9.29}$$

the familiar three-point form, second-order in the interval $h$ between the (equally spaced) points at indices $i-1, i, i+1$. It does not contain the interval $h$. Also, if $\delta^2$ operates on itself, it becomes $\delta^4$, etc. It is an operator, but can in this sense be treated as a multiplier. It will be seen that we do not need to define higher derivatives than the second, operator $\delta^2$.

Smith does not explain the origin of (9.28), but a derivation can be found in Lapidus and Pinder [34, pp. 39-], to which the reader is referred. The form seen in (9.28) is one of several equally valid forms, but is the one chosen in this context, as it allows the Numerov device.

The method will be described as applied to BI, which is the basis for both extrapolation and BDF, both of which can be driven to fourth order accuracy, which is also achieved by the Numerov device applied to the right-hand side of the diffusion equation,

$$\frac{\partial C}{\partial t} = \frac{\partial^2 C}{\partial X^2} . \tag{9.30}$$

Using the usual notation, $C'_i$ denoting the next point in time after the present value $C_i$, $i$ being the index along the $X$ axis, we now discretise the left-hand side only, according to BI (see Chap. 8, but now assuming equal intervals), and use (9.28) for the right-hand side:

$$C'_i - C_i = \lambda\left(\delta_X^2 C'_i - \tfrac{1}{12}\delta_X^4 C'_i + \tfrac{1}{90}\delta_X^6 C'_i - \dots\right) . \tag{9.31}$$

We need not concern us with the implementation of the higher-order derivatives, as will shortly be clear. Now both sides are operated on by $(1 + \frac{1}{12}\delta^2)$, which is the same as adding to each side the operation $\frac{1}{12}\delta_X^2$ on that side. This gives

$$\begin{aligned} &C'_i - C_i + \tfrac{1}{12}\delta_X^2(C'_i - C_i) \\ &\quad = \lambda\left(\delta_X^2 C'_i - \tfrac{1}{12}\delta_X^4 C'_i + \tfrac{1}{90}\delta_X^6 C'_i - \dots + \tfrac{1}{12}\delta_X^4 C'_i - \tfrac{1}{144}\delta_X^6 C'_i + \dots\right) \end{aligned} \tag{9.32}$$

and it is seen that on the right-hand side, there are now only terms in $\delta_X^2$ and $\delta_X^6$, the $\delta_X^4$ terms having cancelled out. We can safely ignore the $\delta_X^6$ and higher terms, and now we have only $\delta_X^2$ terms on both sides of the equation. Now expanding according to the definition of the operation $\delta_x^2$ (9.29), we get

$$C'_i - C_i + \tfrac{1}{12}(C'_{i-1} - 2C'_i + C'_{i+1} - C_{i-1} + 2C_i - C_{i+1}) = \lambda(C'_{i-1} - 2C'_i + C'_{i+1}) \tag{9.33}$$

(recall $\lambda = \delta T/H^2$) which, multiplying by 12 and gathering terms, becomes the familiar form seen in Chap. 8, the $i$th equation of system (8.11) for a general implicit method,

$$C'_{i-1} + aC'_i + C'_{i+1} = b_i \ , \tag{9.34}$$

where now

$$\begin{aligned} a &= \frac{10 + 24\lambda}{1 - 12\lambda} \\ b_i &= \frac{C_{i-1} + 10C_i + C_{i+1}}{1 - 12\lambda} \ . \end{aligned} \tag{9.35}$$

The difference is that this discretisation is $O(H^4)$. The system can be solved as easily by the Thomas algorithm as, say, the usual Laasonen or CN system, but now it will be worthwhile applying a high-order process in the time direction. The easiest one is extrapolation, described above, and it ought to be fourth order, so as to match that of the spatial second derivative. Bieniasz tested the method with three simulation algorithms, comparing with the normal, second-order discretisation: BI (no difference, because of the first-order time derivative), second-order extrapolation (not much difference, the second-order not providing a match for the fourth order) and the Rosenbrock scheme using ROWDA3, which showed a marked improvement in efficiency. Unfortunately, he did not attempt fourth-order extrapolation, which might be expected to perform about as well as Rosenbrock, and would be easier to implement, being simply a series of BI steps.

This method is worth investigating further. An analysis of the stability of the formulae resulting from the method is yet to be done. There are some features to note. Considering the constants definitions above (9.35), there is an apparent problem if $\lambda = 1/12$. In fact, if the whole Eq. (9.34) is multiplied by $1 - 12\lambda$, then this problem becomes a possible advantage, as that equation then simplifies, for $\lambda = 1/12$, to

$$12C'_i = C_{i-1} + 10C_i + C_{i+1} \ . \tag{9.36}$$

It is not clear whether this is a good formula in practice, and in any case, the $\lambda$ value is inconveniently small.

Note also that, if there are homogeneous chemical reaction terms on the right-hand side of (9.30), they can be accommodated without problems; they will lead to some additional terms operated on by $\delta_X^2$. What must not be present are convection terms, since these are spatial first derivatives, making the Numerov method, in this form, impossible to use. However, Bieniasz has devised an improved version, called the “extended Numerov method” [97], which indeed can handle first spatial derivatives and thus convective systems.

#### 9.2.7.2 Hermitian Current Approximation

As already described in some detail in Chap. 3, a one-sided first derivative such as the current approximation $G$ can be raised to higher-order by a Hermitian scheme, as introduced to electrochemistry by Bieniasz [10]. This can then be used both to obtain better current approximations, and also in those cases where $G$ enters a boundary condition. For the simpler case of the current approximation on a spatial grid of concentrations already calculated, see the relevant Sect. 3.6 in Chap. 3. Here we need to go into some detail on the boundary conditions application.

There are simulation cases (for example, using unequal intervals) where it is desirable to use a two-point approximation for $G$, both for the evaluation of a current and as part of the boundary conditions. In that case, an improvement over the normally first-order two-point approximation is welcomed, and Hermitian formulae can achieve this. Two cases of such schemes are now described: that of controlled current and that of an irreversible reaction, as described in Chap. 6, Sect. 6.2.2, using the single-species case treated in that section, for simplicity. The reader will be able to extend the treatment to more species and other cases, perhaps with the help of Bieniasz’ seminal work on this subject [10]. Both the 2(2) and 2(3) forms are given. It is assumed that we have arrived at the reduced didiagonal system (6.3) (page 103) and have done the **u-v** calculation (here, only $u_1$ and $v_1$ are needed).

We must also specify the time integration method used, because the Hermitian scheme makes use of terms in $\mathrm{d}C/\mathrm{d}T$, which must be consistent with the time integration. We assume the three-point BDF method, second-order in time, so that an improvement in the usual two-point $G$-approximation to second or perhaps third-order (in space) will be appropriate.

In the cases to be described below, we have a simple $F$-function, containing only a term in $\mathrm{d}C/\mathrm{d}T$ (see the outline in Chap. 3), which needs approximating. We follow the description given by Bieniasz [10]. With BDF, this is consistently represented as

$$\frac{\mathrm{d}C_i}{\mathrm{d}T} = \frac{{}'C_i - 4C_i + 3C'_i}{2\delta T} \tag{9.37}$$

with ${}'C_i$ the concentration at $T - \delta T$. If the rational BDF start is used (Sect. 4.8.1), then the simulation will start with a single BI step, for which we have

$$\frac{\mathrm{d}C_i}{\mathrm{d}T} = \frac{C'_i - C_i}{\delta T} \; . \tag{9.38}$$

For **controlled current** $G$ and the 2(2) form, the boundary condition becomes the corrected form

$$G = \frac{C'_1 - C'_0}{H} - \frac{H}{2}F_0 \tag{9.39}$$

and multiplying both sides by $H$ and expanding $F_0$ for the BI step results in

$$GH = C'_1 - C'_0 - \frac{H^2}{2}\left(\frac{C'_0 - C_0}{\delta T}\right) . \tag{9.40}$$

Substituting $C'_1 = u_1 + v_1 C'_0$ and rearranging, we obtain the solution

$$C'_0 = \frac{2\delta T(GH - u_1) - H^2 C_0}{2\delta T(v_1 - 1) - H^2} . \tag{9.41}$$

For the subsequent BDF steps, this is

$$GH = C'_1 - C'_0 - \frac{H^2}{2}\left(\frac{{}'C_0 - 4C_0 + 3C'_0}{2\delta T}\right) , \tag{9.42}$$

leading finally to

$$C'_0 = \frac{4\delta T(GH - u_1) + H^2({}'C_0 - 4C_0)}{4\delta T(v_1 - 1) - 3H^2} . \tag{9.43}$$

The 2(3) form starts with

$$G = \frac{C'_1 - C'_0}{H} - \frac{H}{3}F_0 - \frac{H}{6}F_1 . \tag{9.44}$$

Expanding and substituting for both $C'_0$ and $C'_1$ (the latter arising in $F_1$), the final result for the BI step is

$$C'_0 = \frac{6\delta T(GH - u_1) + H^2(u_1 - 2C_0 - C_1)}{6\delta T(v_1 - 1) - H^2(v_1 + 2)} , \tag{9.45}$$

while for the BDF steps it becomes

$$C'_0 = \frac{12\delta T(GH - u_1) + H^2(3u_1 + 2{}'C_0 - 8C_0 +{}'C_1 - 4C_1)}{12\delta T(v_1 - 1) - 3H^2(v_1 + 2)} . \tag{9.46}$$

For the **irreversible case** with dimensionless heterogeneous rate constant $K$, $G$ is given as

$$G = KC'_0 \tag{9.47}$$

and using this instead of $G$ as above, we have for the 2(2) scheme and BI,

$$KHC'_0 = C'_1 - C'_0 - \frac{H^2}{2}\left(\frac{C'_0 - C_0}{\delta T}\right) \tag{9.48}$$

which rearranges to

$$C'_0 = \frac{2\delta T u_1 + H^2 C_0}{2\delta T(KH - v_1 + 1) + H^2} \tag{9.49}$$

or for the BDF steps

$$C'_0 = \frac{4\delta T u_1 - H^2({}'C_0 - 4C_0)}{4\delta T(KH - v_1 + 1) + 3H^2}\;. \tag{9.50}$$

For the 2(3) scheme and BI,

$$C'_0 = \frac{6\delta T u_1 - H^2(u_1 - 2C_0 - C_1)}{6\delta T(KH - v_1 + 1) - H^2(v_1 + 2)} \tag{9.51}$$

and for the BDF steps,

$$C'_0 = \frac{12\delta T u_1 - H^2(3u_1 + 2'C_0 - 8C_0 +' C_1 - 4C_1)}{12\delta T(KH - v_1 + 1) - 3H^2(v_1 + 2)}\;. \tag{9.52}$$

Some experiments show that the 2(2) forms are sufficient here, the 2(3) forms not leading to further improvement in accuracy. This is no doubt because the three-point BDF algorithm used, started with a BI step, is second-order accurate in time, so a third-order form cannot improve the accuracy. A higher-order algorithm, such as ROWDA3 as used by Bieniasz [10] or indeed a higher-order BDF such as four-point, would make the higher 2(3) form more useful. Note also that in the above examples, we have approximated $\partial C/\partial T$ as the two-point form, and could probably do better using the three-point BDF form, consistent with the use of BDF3.

The example program `CHRONO_CN_HERM` shows a Crank–Nicolson implementation and again, the 2(3) form is only marginally better than 2(2). Bieniasz showed how to use this scheme for extrapolation and for the Rosenbrock ROWDA3 scheme [10]. The reader is referred to that paper for details, where still higher-order forms are found. The paper makes it clear that extremely small errors can be achieved by using this method.

#### 9.2.7.3 Method of Wu and White

Wu and White [98] have described a new method that is reminiscent of the earlier work of Kimble and White [14] but makes use of the Hermitian method (that is, using derivatives) to achieve higher-order solutions for several concentration rows at a time. They also suggest, but do not demonstrate, the use of their new scheme as a possible start-up for BDF. The reader is referred to their paper for details.

## 9.3 MOL and DAE

The **MOL** is not so much a particular method as a way of approaching numerical solutions of *pde*s. It is described well by Hartree [99] as the "replacement of the second-order (space) derivative by a finite difference"; that is, leaving the first (time) derivative as it is, thus forming from, say, the diffusion equation a set of ordinary differential equations, to be solved in an unspecified manner. Thus, a system such as (9.16) on page 184 can be written in the general vector–matrix form

$$\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}T} = \mathbf{A}\mathbf{C} + \mathbf{s}\,, \tag{9.53}$$

where **C** is the concentration vector, **A** is the matrix of coefficients in the system and **s** is a vector of known quantities arising from the particular boundary conditions. From this point on, a large variety of methods for solving this system can be used. This encompasses all the methods so far described, but the term MOL nowadays implies a particular method. This consists of using a variety of computer packages to solve the set of *ode*s, usually with a high degree of autonomy with respect to time intervals and if, for example, BDF is used (as it often is with these packages), with respect to BDF order. The word "lines" comes from the fact that the solution is advanced a "line" at a time, the line stretching along the space dimension, and advancing up the time axis.

The method has a long history. The name MOL seems to have become established around 1960. Before this, various authors either used the word "line" [100] or expressions like "on certain lines" [101] or a description of the idea. In the book by Kantorovich and Krylov [101], there is a reference to a 1934 paper [102]. It is also cited by Liskovets [103] as a source paper, along with Rothe [104], who might be the first. Hartree and Womersley [105] use, in their summary, the words "approximating by use of finite intervals in one variable, and integrating exactly in the other variable". The book by Schiesser [106] is the standard work now (he calls the method NUMOL, for numerical method of lines). Electrochemical use of MOL has been sparse. Lemos and coworkers [79–82] have investigated the method, using various solution methods; Lasia and Grégoire [107] used it in conjunction with a professional *ode* solver package, as did Zhang and Cheh [108].

MOL is intimately bound up with another method, that of using **DAE** sets. It can be thought of as an extension of MOL. Therefore, MOL should be described here, and it is in fact simple. In the most popular form of MOL, the diffusion equation is discretised on a grid in the spatial dimension(s) only, leaving the time derivative as it is. This results in a set of ordinary differential equations (9.53) as seen, for example, in the system (9.16) in this chapter. There, (6,5)-point approximations are used for the spatial derivative, but this is immaterial; three-point formulae such as described in Chap. 8, (8.1) on page 145 are more commonly used. In Chap. 6 the discretisation of boundary conditions is described. The idea there is that a system like (9.16) is solved in two steps by, for example, the Thomas algorithm. At the end of the first stage, boundary conditions are expressed discretely and used in the second stage. However, another approach is to add the discrete expressions for boundary conditions to the *ode* system. These expressions are algebraic equations. For example, for the Cottrell system, the expression

$$C_0 = 0 \tag{9.54}$$

might be added to the system. Or, if the experiment is that of chronopotentiometry, discretised derivative boundary conditions will be developed and added to the system, such as (6.4) or more complex discretisations. The result is always a system of equations, some of which are differential and some algebraic. This is a DAE system, and there are professional packages for their numerical solution. The standard text is that of Brenan et al. [109], in which references are to be found to existing packages such as LSODE and DASSL [110]. These can mostly be found at the `netlib` site [111]. More on these packages can be found in Chap. 17.

A little detail is appropriate here, especially as this is needed for the next section. Consider a set of *ode*s using three-point approximation in space, such as (8.1) on page 145, for simplicity. For $N$ internal points, there will then normally be $N$ such equations in the set. Up to this point, the method has been either to substitute for the boundary values according to the equations describing them (the boundary conditions) or, as in the case of the implicit methods described above, to perform a Thomas process going backward from the external boundary, and then to solve for the value of $C_0$, for example using the **u-v** process as described above. If Runge–Kutta is used, one begins by generating a particular $\mathbf{k}_i$ vector, then uses this to calculate the $k_i$ belonging to $C_0$ (and possibly to $C_{N+1}$, if that also changes with time), going on from there. These methods in one way or another separate the treatment of boundary values from that of the internal points. However, as mentioned above, the equations describing boundary values can also be added to the equation set. They are always algebraic equations, so that the whole set is then a DAE set. As a simple example, if we simulate chronopotentiometry using a two-point approximation for the current, and equal spatial intervals $H$, the DAE set corresponding to (8.1) becomes

$$\begin{aligned}
0 &= C_1 - C_0 - HG \\
\frac{\mathrm{d}C_1}{\mathrm{d}T} &= \frac{1}{H^2}\left(C_0 - 2C_1 + C_2\right) \\
\ldots
\end{aligned}$$

$$\begin{aligned}\frac{\mathrm{d}C_i}{\mathrm{d}T} &= \frac{1}{H^2}\left(C_{i-1} - 2C_i + C_{i+1}\right) \\ &\cdots \\ \frac{dC_N}{\mathrm{d}T} &= \frac{1}{H^2}\left(C_{N-1} - 2C_N + C_{N+1}\right) \\ 0 &= C_{N+1} - 1\ .\end{aligned} \tag{9.55}$$

The first equation is a description of the (controlled) current approximation, and the last equation expresses the unity value of the outer boundary value.

There are now two principal ways of handling this set. One is to decide on some discretisation of the time derivatives, rendering the *ode*s into algebraic equations, and solving the lot, for the next time step. The method chosen might be BDF, for example, which is indeed used in the DAE solver package DASSL [110]. Seen in this light, DAE sets might be considered always to be involved; when we use the **u-v** mechanism along with the Thomas algorithm, we are essentially solving the DAE set in an efficient manner. The other approach goes in the opposite direction, as it were. All the *ode*s in the set are left as such, and the algebraic equations are solved along with them, using an *ode* solver. One of these is Runge–Kutta but, as was mentioned above, explicit RK is not very efficient, so an implicit method suggests itself, such as Rosenbrock, described for sets of *ode*s in Chap. 4. This is dealt with in the next section.

## 9.4 The Rosenbrock Method

For the basics of this method, see Chap. 4. There it was mentioned that Bieniasz introduced this method to electrochemical simulation [91], preferring ROWDA3, a third-order variant that also has a smooth response. There exists a second-order variant with a smooth response, ROS2, due to Lang [112], which might be more appropriate if second-order spatial derivative approximations are to be used. Coefficients for some variants are given in Appendix A. The object here is to describe the way Rosenbrock methods are used in the present context. The Bieniasz paper [91] shows the way (but the standard symbols, as used in Chap. 4, are used here, rather than those used by Bieniasz).

The set (9.55) (or one like it, with whatever boundary conditions we might have) is written in the compact form

$$\mathbf{S}\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}T} = \mathbf{F}(T, \mathbf{C}) \tag{9.56}$$

where **S** is the selection matrix. It is diagonal, and contains zeroes in those positions where the DAE set has an algebraic equation (that is, zero on the left-hand side of, say, (9.55)) and unity in those positions corresponding to the *ode*s. In a sense, the zeroes say "zero $\mathrm{d}C/\mathrm{d}T$". On the right-hand side, the function $\mathbf{F}(T, \mathbf{C})$ is a matrix–vector function expressing the whole collection of the right-hand sides of, say, the set (9.55). The variable $T$ is included for the important generality for those cases where some variables are time-dependent. This is the case, for example, in LSV simulations, where the potential is a function of time, or cases where the electroactive substance itself undergoes a homogeneous reaction, as in the Reinert–Berg system or, of more interest in this context, the second-order Birk–Perone system, which gives rise to nonlinear equations, both described in the subsection beginning on page 22. As was seen in Chap. 4, one of the strengths of Rosenbrock is indeed its easy handling of nonlinear sets.

Applying (9.56) to the specific set 9.55 for illustration, we have

$$\mathbf{S} = \begin{bmatrix} 0 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix}, \tag{9.57}$$

that is, $\mathbf{S}$ is almost an identity matrix except that the first and last diagonal elements are zero, indicating algebraic equations. The function $\mathbf{F}(T, \mathbf{C})$ can be broken up into

$$\mathbf{F}(T, \mathbf{C}) = \mathbf{J}\mathbf{C} + \mathbf{s}\,, \tag{9.58}$$

in which the matrix $\mathbf{J}$ multiplies the concentration vector; it is written as $\mathbf{J}$ here because it is the Jacobian, and will later figure as such. Vector $\mathbf{s}$ arises from the constant terms of the DAE set. The example set (9.55) resolves into

$$\mathbf{J} = \frac{1}{H^2} \begin{bmatrix} -H^2 & H^2 & & & & \\ 1 & -2 & 1 & & & \\ & \ddots & & & & \\ & & & 1 & -2 & 1 \\ & & & & & H^2 \end{bmatrix}, \tag{9.59}$$

$$\mathbf{s} = \frac{1}{H^2} \begin{bmatrix} -H^3 G \\ 0 \\ \vdots \\ 0 \\ -H^2 \end{bmatrix} \tag{9.60}$$

and the $\mathbf{C}$ vector is indexed from 0 to $N + 1$.

We are now ready to invoke the Rosenbrock method. A number $s$ of $\mathbf{k}_i$ vectors must be computed, $s$ being the order chosen. The general equation for each one is an extension of that given for a pure *ode* set on page 82, Eq. (4.70), to the present DAE case, introducing the selection matrix $\mathbf{S}$ and following Bieniasz [91] (though with the more common notation):

$$\begin{aligned}[\mathbf{S} - \gamma \delta T \mathbf{F}_C(T, \mathbf{C})]\, \mathbf{k}_i =& \gamma \Big( \delta T \mathbf{F}\big(T + \alpha_i \delta T,\ \mathbf{C} + \sum_{j=1}^{i-1} a_{ij} \mathbf{k}_j\big) \\ & + \mathbf{S} \sum_{j=1}^{i-1} c_{ij} \mathbf{k}_j + \gamma_i \delta T^2 \mathbf{F}_T \Big)\ .\end{aligned} \tag{9.61}$$

Here, there appear the Jacobian $\mathbf{F}_C$, which is in fact $\mathbf{J}$ as defined above in (9.59), the function $\mathbf{F}$ itself, applying at partly augmented $T$ and $\mathbf{C}$ values, and, in case of time-dependent systems, the time derivative $\mathbf{F}_T$, written in short form, as it is applied to the present $T$ and $\mathbf{C}$. This last term is often zero, if the system does not include functions of time.

Although (9.61) may look formidable, there are some conveniences. First of all, for linear systems, the first matrix term on the left-hand side is a constant and can be evaluated once and for all. We write

$$\mathbf{M} = \mathbf{S} - \gamma \delta T \mathbf{F}_C\ . \tag{9.62}$$

In fact, in practice, some further tidying up is possible, by combining the quantities $\delta T$ and $\dfrac{1}{H^2}$ into the familiar $\lambda$ and dividing throughout by some factors, but this is a practical detail of no importance here. The right-hand side of (9.61) will need to be evaluated at every step, $s$ times. At each stage $i$, the equation can be written as

$$\mathbf{M}\mathbf{k}_i = \mathbf{B}_i \tag{9.63}$$

where $\mathbf{B}_i$ is the evaluated right-hand side of (9.61). Thus, a linear system must be solved to obtain each $\mathbf{k}_i$. The matrix $\mathbf{M}$ will normally be either tri- or pentadiagonal or, in cases of simulations in more dimensions, will be rather sparse, so that either the Thomas algorithm or an offshoot of it, or a sparse solver, can be used, for efficiency. Also, the favoured Rosenbrock variants such as ROS2 or ROWDA3 (which Bieniasz prefers) have some zero coefficients, resulting in calculations that need not be repeated after the first stage, thus further increasing efficiency. Ludwig et al. [113] however find that another variant, ROS3P, is more efficient than ROWDA3.

Having calculated all $s$ $\mathbf{k}_i$ vectors, the RK formula is then used, here

$$\mathbf{C}_{n+1} = \mathbf{C}_n + \sum_{i=1}^{s} m_i \mathbf{k}_i \tag{9.64}$$

to compute the next concentration vector.

There are advantages, and also drawbacks, of this method. The advantages are great efficiency, stability and a smooth error response if ROS2 or ROWDA3 are used (see a study by Bieniasz [91], albeit not including ROS2), and the easy handling of time-dependent and/or nonlinear systems. No Newton iterations are required for nonlinear systems (but see below). The most serious drawback is that the method does not lend itself to problems with sharp initial transients, such as a potential step method; at least, not for the very first step, as pointed out by Bieniasz [91]. The reason is inconsistency. For example, in the Cottrell experiment, it is not possible to calculate a derivative $\mathbf{J}$ at the starting point, $T = 0$. One way to overcome this, taken by Bieniasz [91], is to start by invoking the boundary condition for $T > 0$ even initially. This can work, but can also lead to a persistent degradation of the results. In practical terms, for the Cottrell system, where all $C_i$, including $C_0$ should be unity at $T = 0$, we set $C_0 = 0$ at that time, and proceed. Note that this is exactly what is done in the explicit method. In order to avoid the degradation in accuracy (desirable for efficiency), the proper way is to use a different algorithm for the very first step, choosing one that expresses derivatives at $T = \delta T$, that is, an implicit method such as BI (Laasonen), perhaps coupled with extrapolation for improved accuracy. This might be considered defeating the Rosenbrock advantages, because if the system is nonlinear, one now needs some Newton iterations after all, and it could be argued that since one has programmed BI/extrapolation, one may as well proceed with it over the whole simulation. But Rosenbrock may be more efficient, so this is a compromise between programming effort and efficiency of computation.

Finally, it is to be noted that, when using Rosenbrock for an LSV simulation, one must be aware that the potential $p$ at time $T$, at a given step goes to $p + \delta p$ at $T + \delta T$. It is the old value $p$ that must be used in the boundary expressions, augmented by the $\alpha$ coefficients in higher stages. It is incorrect, in other words, to add $\delta p$ to $p$ at the beginning of the iteration loop.

### 9.4.1 An Example, the Birk–Perone System

There is an example program described in Appendix E, `BPROS`, applying Rosenbrock to the Birk–Perone system, in which we have both time-dependence and nonlinear equations. It is described in Chap. 2, pages 23–24. Equation (2.72), with boundary conditions, when semidiscretised using equal intervals in space, leads to the DAE system

$$\begin{aligned}
0 &= C_0 \\
\frac{\mathrm{d}C_1}{\mathrm{d}T} &= \frac{1}{H^2}\left(C_0 - 2C_1 + C_2\right) - KC_1^2 \\
\dots \\
\frac{\mathrm{d}C_i}{\mathrm{d}T} &= \frac{1}{H^2}\left(C_{i-1} - 2C_i + C_{i+1}\right) - KC_i^2 \\
\dots \\
\frac{\mathrm{d}C_N}{\mathrm{d}T} &= \frac{1}{H^2}\left(C_{N-1} - 2C_N + C_{N+1}\right) - KC_N^2 \\
0 &= C_{N+1} - \frac{1}{1+KT} \,.
\end{aligned} \tag{9.65}$$

The last equation expresses the analytical solution for the time decay of the substrate. Here, then, we have selection matrix **S** as above (9.57), and the function $\mathbf{F}(T, \mathbf{C})$ is

$$\mathbf{F}(T, \mathbf{C}) = \begin{bmatrix} C_0 \\ \frac{1}{H^2}\left(C_0 - 2C_1 + C_2\right) - KC_1^2 \\ \vdots \\ \frac{1}{H^2}\left(C_{N-1} - 2C_N + C_{N+1}\right) - KC_N^2 \\ C_{N+1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ -\frac{1}{1+KT} \end{bmatrix}, \tag{9.66}$$

or regarding (9.58), $\mathbf{F}_C$, the derivative of $\mathbf{F}(T, \mathbf{C})$ with respect to **C**,

$$\mathbf{F}_C = \frac{1}{H^2}\begin{bmatrix} H^2 & & & & \\ 1 & -2(1+H^2KC_1) & 1 & & \\ & \ddots & & & \\ & 1 & -2(1+H^2KC_N) & 1 & \\ & & & & H^2 \end{bmatrix} \tag{9.67}$$

which contains concentration terms on the diagonal. For this reason, with this problem it is necessary to evaluate $\mathbf{F}_C$ at every step. Noting the form of (9.61) and (9.62), and recalling $\lambda = \delta T/H^2$, it is convenient here to redefine **M** as

$$\mathbf{M} = \frac{\mathbf{S}}{\gamma} - \delta T \mathbf{F}_C \tag{9.68}$$

and after division by $-\lambda$, resulting in

$$-\frac{\mathbf{M}}{\lambda}=\begin{bmatrix} H^2 & & & & \\ 1 & \left(-\frac{1}{\gamma\lambda}-2(1+H^2KC_1)\right) & 1 & & \\ & & \ddots & & \\ & & 1 & \left(-\frac{1}{\gamma\lambda}-2(1+H^2KC_N)\right) & 1 \\ & & & & H^2 \end{bmatrix} \tag{9.69}$$

(9.61) becomes

$$-\frac{\mathbf{M}}{\lambda}\mathbf{k}_i=-\frac{\delta T}{\lambda}\mathbf{F}\big(T+\alpha_i\delta T,\mathbf{C}+\sum_{j=1}^{i-1}a_{ij}\mathbf{k}_j\big)-\frac{\mathbf{S}}{\lambda}\sum_{j=1}^{i-1}c_{ij}k_j-\frac{\gamma_i\delta T^2}{\lambda}F_T(T,\mathbf{C})\ , \tag{9.70}$$

where $\mathbf{F}_T$ remains to be defined. It is the time-derivative of $\mathbf{F}(T,\mathbf{C})$ and only contains one non-zero element:

$$\mathbf{F}_T=\begin{bmatrix} 0 \\ \vdots \\ 0 \\ -\frac{K}{(1+KT)^2} \end{bmatrix} . \tag{9.71}$$

The above equation evaluates to a tridiagonal linear equation system, after some arrangement,

$$\begin{aligned} H^2k_0 &= b_0 \\ k_0+a_1k_1+k_2 &= b_1 \\ &\cdots \\ k_{l-1}+a_lk_l+k_{l+1} &= b_l \\ &\cdots \\ k_{N-1}+a_Nk_N+k_{N+1} &= b_N \\ H^2k_{N+1} &= b_{N+1} \end{aligned} \tag{9.72}$$

with

$$a_l=-\frac{1}{\gamma\lambda}-2(1+H^2KC_l) \tag{9.73}$$

with the vector **b** arising in an obvious manner from the evaluation of the right-hand side of (9.70). This is the usual form for implicit systems, as seen in Chap. 8, page 147, albeit for unknown concentrations, here for unknown **k**. Index $l$ is used, $i$ being reserved for the stage number here. The system is solved by the Thomas algorithm.

Execution of the program BPROS shows that it works well, attaining a relative accuracy of about $10^{-4}$ in about 100 steps of $\delta T = 0.01$, both with ROS2 and ROWDA3, the latter being slightly better (but using about 50 % more CPU time).

## 9.5 FEM, BEM, FVM and FAM (Briefly)

There is a class of methods called finite element method (FEM) and the related boundary element method (BEM), also called boundary integral element method (BIEM), and the finite analytical method (FAM). These will be given very short shrift, in part because they constitute a large subject, many textbooks being devoted to FEM and BEM alone. The usual approach is to use ready program packages. The only member of the FEM group that will be described here is orthogonal collocation, which has its own section (see below).

Roughly, FEM consists of choosing regions in the simulation space, marked by node points, and fitting "trial" functions to the regions, in some optimal manner. What is considered optimal is defined in several different ways. With BEM, only points on the boundary are chosen and a function fitted to the space delimited by these points is optimised. Thus, BEM uses fewer points than FEM. Both methods appear to be highly efficient. FAM is similar to FEM, but instead of fitting an arbitrary function to the elements (in FEM, usually polynomials), local analytical solutions are sought for each of the elements.

Here are a few brief references to recent or key works in which these methods have been described as used in electrochemical simulations. The interested reader is urged to look these up and follow the references contained in them to the seminal works and textbooks. Of necessity, much work is left uncited here.

Ferrigno et al. [114] describe the use of FEM for steady state simulations of recessed, flush and protruding ultramicrodisk electrodes, giving a good description of FEM. The method was made adaptive by Nann and Heinze [115, 116], and Harriman et al. later published an extensive series of papers on adaptive FEM [117–121]. More recent applications (of the last 5 years or so) are [122–133]. Many FEM simulations are conducted using software like COMSOL Multiphysics® since no programming knowledge is required of the user, but Morf et al. [134] used Excel, Sanecki et al. [128] used their own ESTYM_PDE (using orthogonal collocation) and Ueno et al. [131] used PDEASE (see Chap. 17 for these).

BEM might be thought of as best suited to steady state problems, and has been used for this, for example in corrosion simulations [135] and current distributions [136], but recently also for time-marching problems [137]. Electrochemical applications include [130, 136, 138–154].

FAM has been investigated by Jin and Qian et al. [155–160].

The finite volume method FVM should probably be studied more than it is for electrochemical applications, but it has been applied [161, 162]. It is possibly related to the box method. A good text on FVM is that of Patankar [11].

The newer method of Bortels et al., called multidimensional upwinding method (MDUM) should also be mentioned [163]. It was applied to a problem involving diffusion, convection and migration, both steady state and time-marching.

## 9.6 Orthogonal Collocation (OC)

This is one of the variants of the FEMs. The essence of orthogonal collocation (**OC**) is that a set of orthogonal polynomials is fitted to the unknown function, such that at every node point, there is an exact fit. The points are called collocation points, and the set of polynomials is chosen suitably, usually as Jacobi or Chebyshev polynomials. The optimal choice of collocation points is to make them the roots of the polynomials. There are tables of such roots, and thus point placements, in Appendix A. The notable things here are the small number of points used (normally, about 10 or so will do), their uneven spacing, crowding closer both at the electrode and (perhaps strangely) at the outer limit, and the fact that the outer limit is always unity. This is discussed below.

The method's historical origins are complex but electrochemists used OC first in 1970 [164], referring to an earlier work [165], for certain *ode*s. Caban and Chapman [166] then used OC to compute (steady state) current distributions, but the work most cited by later users of OC in electrochemistry is that of Whiting and Carr [167], who described its use in time-marching problems. They refer to the work of chemical engineers Villadsen and Stewart [168]. A later book [169] is a good source also, as is the chapter by Pons [170], drawing heavily on [169]. A number of electrochemists have published in the area, notably Pons and Speiser [171, 172], and later Speiser and coauthors; see the review by Speiser [173], for a complete list of references and a good description of OC, among other topics.

The possibly peculiar spacing of the collocation points, crowding close both at the electrode and at the outer diffusion limit, does not matter too much, and seems unnecessary. For example, using only five internal points (that is, five apart from zero and unity), they are placed at the values 0.047, 0.231, 0.5, 0.769, 0.953, a series that is symmetrical about the midway point at 0.500. This spacing has been circumvented by Yen and Chapman [174], using Chebyshev polynomials that open out towards the outer limit. Their work has apparently not been followed up.

OC is capable of high accuracy and efficiency. Some comparisons have been made with normal finite difference methods. Eddowes [175] found OC superior, while Magno et al. [176] found it inferior to plain EX with expanding intervals (this appears doubtful to the present authors). Bieniasz and Britz [177] cast some doubt on OC, pointing out possible problems with the fit in between the collocation points, possibly leading to negative concentrations or (see below) errors in the current values computed from it. This was rebutted by Speiser [178], rather convincingly.

The essence is that, if the concentration profile simulated is smooth (which it normally is), then the polynomials will be well behaved in between points and no such problems will be encountered. As is seen below, implicit boundary values can easily be accommodated, and by the use of spline collocation [179–181], homogeneous chemical reactions of very high rates can be simulated. This refers to the static placement of the points. Having, for example, the above sequence of points for five internal points, the point closest to the electrode is at 0.047. This will be seen, below, to be in fact further from the electrode than it seems, because of the way that distance $X$ is normalised so that, for very fast reactions that lead to a thin reaction layer, there might not be any points within that layer. Spline collocation thus takes the reaction layer and places another polynomial within it, while the region further out has its own polynomial. The two polynomials are designed such that they join smoothly, both with the same gradient at the join. This will not be described further here.

For the description of how OC works, assume for simplicity a single substance. The diffusion equation, including a homogeneous reaction, is

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} + f(c) \tag{9.74}$$

with $f(c)$ being the homogeneous reaction term, left unspecified. The equation is written in dimensioned form for a reason. In OC, the space axis is normalised in a manner different from the usual. It is here normalised by the total diffusion space width $L$, so that the range is $0 \leq X \leq 1$, in order to fit in with the range of the polynomial. The value of $L$ depends on the experiment being simulated, and will be a multiple of $\sqrt{D\tau}$, $\tau$ being some characteristic time. As explained in previous chapters, $\tau$ might be the duration of the experiment for pulse experiments or the length of time taken by the potential, for a linear sweep, to change by one dimensionless potential unit. In general, it is given by

$$L = f\sqrt{D\tau} \tag{9.75}$$

and thus,

$$X = x/(f\sqrt{D\tau}) \; . \tag{9.76}$$

Time and concentration are normalised as usual (Sect. 2.3), and this leads to

$$\frac{\partial C}{\partial T} = \frac{1}{f^2}\frac{\partial^2 C}{\partial X^2} + F(C) \; , \tag{9.77}$$

where we have the factor $1/f^2$, (and $F(C)$ is the dimensionless form of the rate equation for the homogeneous reaction term). The new factor is usually written as $\beta$, and is often discussed as an arbitrarily adjustable parameter. But it is not; it must be determined by $L$, which is known for a given experiment. For, say, the Cottrell experiment, $f = 6$, while for a linear or cyclic sweep, it is $6\sqrt{T}$, with $T$ being the total number of potential (or time) units swept during the experiment. If one takes $\beta$ to be arbitrary, one might either, by making it too small (i.e. $L$ too large), simulate a far too wide diffusion space and thus degrade the resolution near the electrode or, by making it too large, simulate in a confined space, so that the outer boundary concentration cannot be taken as constant throughout. The choice of $\beta$ is always rational.

As an aside, there have been some interesting attempts to make the diffusion space variable with time and to normalise by that variable. Yen and Chapman [174] used this, and Urban and Speiser [182]. The diffusion equation then normalises to a rather more complicated form, sometimes into a plain second-order *ode*, or in other cases, into a form including time-dependent terms in $\partial C/\partial X$. Results [182] appeared to be very good. This has apparently not been followed up, but perhaps it should be.

Now for the description of how OC works. Assume a number $N + 2$ points situated at $X_0, X_1, \ldots, X_N, X_{N+1}$, and $X$ lying in the interval $[0, 1]$ by the normalisation described above. The points are chosen, following the work of Whiting and Carr [167], as the roots of shifted Jacobian polynomials with parameters as given in Tables A.5, A.6, and A.7 in Appendix A. The tabled values were computed using the subroutine JCOBI mentioned in Appendix E. The concentration profile is approximated by the polynomial $P(X)$,

$$C(X) \approx P(X) = \sum_{j=0}^{N+1} b_j X^j \tag{9.78}$$

where $b_j$ are coefficients which, as it happens, we never need to find. The OC method assumes that $P(X)$ exactly fits $C(X)$ at each of the collocation points. We have some derivatives,

$$\frac{\mathrm{d}C}{\mathrm{d}X} = \frac{\mathrm{d}}{\mathrm{d}X} P(X) = \sum_{j=0}^{N+1} j b_j X^{j-1} \tag{9.79}$$

and

$$\frac{\mathrm{d}^2 C}{\mathrm{d}X^2} = \frac{\mathrm{d}^2}{\mathrm{d}X^2} P(X) = \sum_{j=0}^{N+1} j(j-1) b_j X^{j-2} \,. \tag{9.80}$$

Equations (9.78)–(9.80) can be written out for every value of $X_j$, leading to the systems of equations

$$\begin{aligned} C_0 &= \sum_{j=0}^{N+1} b_j\, X_0^j \\ C_1 &= \sum_{j=0}^{N+1} b_j\, X_1^j \\ &\dots \\ C_{N+1} &= \sum_{j=0}^{N+1} b_j\, X_{N+1}^j \end{aligned} \tag{9.81}$$

for the concentrations themselves,

$$\begin{aligned} \frac{\mathrm{d}C_0}{\mathrm{d}X} &= \sum_{j=0}^{N+1} j\, b_j\, X_0^{j-1} \\ \frac{\mathrm{d}C_1}{\mathrm{d}X} &= \sum_{j=0}^{N+1} j\, b_j\, X_1^{j-1} \\ &\dots \\ \frac{\mathrm{d}C_{N+1}}{\mathrm{d}X} &= \sum_{j=0}^{N+1} j\, b_j\, X_{N+1}^{j-1} \end{aligned} \tag{9.82}$$

for the first derivatives with respect to $X$, and

$$\begin{aligned} \frac{\mathrm{d}^2C_0}{\mathrm{d}X^2} &= \sum_{j=0}^{N+1} j(j-1)\, b_j\, X_0^{j-2} \\ \frac{\mathrm{d}^2C_1}{\mathrm{d}X^2} &= \sum_{j=0}^{N+1} j(j-1)\, b_j\, X_1^{j-2} \\ &\dots \\ \frac{\mathrm{d}^2C_{N+1}}{\mathrm{d}X} &= \sum_{j=0}^{N+1} j(j-1)\, b_j\, X_{N+1}^{j-2} \end{aligned} \tag{9.83}$$

for the second derivatives. These equations are now written in matrix form:

$$\begin{aligned}\mathbf{C} &= \mathbf{Qb}\\ \frac{\mathrm{d}\mathbf{C}}{\mathrm{d}X} &= \mathbf{Rb}\\ \frac{\mathrm{d}^2\mathbf{C}}{\mathrm{d}X^2} &= \mathbf{Sb}\,,\end{aligned}\tag{9.84}$$

with **Q**, **R** and **S** obvious from the systems above.

So far, we have the set of coefficients, vector **b**, which we do not know. These are eliminated by expressing the first equation of the set (9.84) explicitly for **b**:

$$\mathbf{b} = \mathbf{Q}^{-1}\mathbf{C}\tag{9.85}$$

and substituting for it in the other two, giving

$$\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}X} = \mathbf{RQ}^{-1}\mathbf{C} = \mathbf{VC}\tag{9.86}$$

and

$$\frac{\mathrm{d}^2\mathbf{C}}{\mathrm{d}X^2} = \mathbf{SQ}^{-1}\mathbf{C} = \mathbf{WC}\tag{9.87}$$

with

$$\mathbf{V} = \mathbf{RQ}^{-1}\tag{9.88}$$

and

$$\mathbf{W} = \mathbf{SQ}^{-1}\,.\tag{9.89}$$

The above presupposes that **Q** is invertible, and this is the case, as the system (9.81) has no linearly dependent pairs of rows. The interesting thing is that **V** and **W** can be precomputed, for a given $N$, once and for all, simply from the Jacobi roots. Equation (9.87) can now be inserted in (9.77), to produce

$$\frac{\partial\mathbf{C}}{\partial T} = \frac{1}{f^2}\mathbf{WC} + F(\mathbf{C})\tag{9.90}$$

or, the set of $N+2$ equations

$$\begin{aligned}
\frac{\partial C_0}{\partial T} &= \frac{1}{f^2}(W_{0,0}C_0 + W_{0,1}C_1 + \cdots + W_{0,N}C_N + W_{0,N+1}C_{N+1}) + F(C_0) \\
\frac{\partial C_1}{\partial T} &= \frac{1}{f^2}(W_{1,0}C_0 + W_{1,1}C_1 + \cdots + W_{1,N}C_N + W_{1,N+1}C_{N+1}) + F(C_1) \\
\frac{\partial C_2}{\partial T} &= \frac{1}{f^2}(W_{2,0}C_0 + W_{2,1}C_1 + \cdots + W_{2,N}C_N + W_{2,N+1}C_{N+1}) + F(C_2) \\
&\ldots \qquad (9.91) \\
\frac{\partial C_N}{\partial T} &= \frac{1}{f^2}(W_{N,0}C_0 + W_{N,1}C_1 + \cdots + W_{N,N}C_N + W_{N,N+1}C_{N+1}) \\
&\quad + F(C_N)\ . \\
\frac{\partial C_{N+1}}{\partial T} &= \frac{1}{f^2}(W_{N+1,0}C_0 + W_{N+1,1}C_1 + \cdots + W_{N+1,N}C_N \\
&\quad + W_{N+1,N+1}C_{N+1}) + F(C_{N+1})
\end{aligned}$$

This is written out in order to make the next point. The first and last equation in the set are superfluous, because the boundary concentrations $C_0$ and $C_{N+1}$ are not subject to diffusion changes, but to other conditions. Also, where the boundary values appear in the other equations, they must be replaced with what we can substitute for them. The outer boundary value, $C_{N+1}$, is (almost always) equal to the initial bulk concentration $C^*$, usually equal to unity in its dimensionless form. This means that the last term in each equation separates out as a constant term and makes for a constant vector $\left[W_{1,N+1}C^* \;\; W_{2,N+1}C^* \;\; \ldots \;\; W_{N,N+1}C^*\right]^T$, which will be called $\mathbf{Z}$ here. The concentration at the electrode $C_0$ is handled according to the boundary condition. For Cottrell, for example, it is set to zero throughout and thus simply drops out of the set. For other conditions, for example constant current or an irreversible reaction, a gradient $G$ is involved, as described in Chap. 6. In that chapter, the gradient was expressed as a possibly multi-point approximation, but here we have a better device: the use of matrix $\mathbf{V}$, applying it to obtain $G = \mathrm{d}C/\mathrm{d}X(X=0)$:

$$G = \sum_{j=0}^{N+1} V_{0,j}C_j\ . \qquad (9.92)$$

This is written explicitly for $C_0$,

$$C_0 = \beta_1 C_1 + \beta_2 C_2 + \cdots + \beta_N C_N + (\beta_{N+1}C^* - G/V_{0,0}) \qquad (9.93)$$

where $\beta_j = -V_{0,j}/V_{0,0}$. This can be substituted into the $N$ equations of the set, which adds terms to the $W$-coefficients and the constant term (the one in brackets on the right-hand side of (9.93)) to the constant vector $\mathbf{Z}$. We thus obtain a smaller $N \times N$ equation set,

$$\frac{\partial \mathbf{C}}{\partial T} = \frac{1}{f^2}\mathbf{W}'\mathbf{C} + \mathbf{Z} + F(\mathbf{C}) \,. \tag{9.94}$$

For more than one species, the development is clear, based on Chap. 6, leading to larger systems of equations.

The big advantage here is that the matrix $\mathbf{W}$ is always the same for a given $N$. If one works always with some favourite value, such as 10 (a good value), then one needs to compute $\mathbf{W}$, and indeed $\mathbf{V}$, only once, and use it as input data thereafter.

We have arrived at the point where a choice needs to be made of how to proceed with the simulation. Note that (9.94) is in fact of the same form as that obtained when using MOL, being of the same form as the set (9.16), but with more coefficients on every line. From here on, one can use a variety of methods to do the time-march. All of the methods considered in earlier chapters, and this chapter, can be used. As with FEM, however, there is a certain tradition here, for using ready-made *ode* solvers. Villadsen and Michelsen [169] use an implicit Runge–Kutta algorithm devised by Caillaud and Padmanabhan [183], implementing it in their routine `STIFF3`. The routine was reproduced by Pons [170] in his chapter, mentioning that he too finds the method of Caillaud and Padmanabhan best. Whiting and Carr [167] wrote their own solver, based on a predictor–corrector algorithm. Speiser [173] describes several other subroutine packages, such as `DDEBDF` arising from the work of Gear [184] or `LSODE` of Hindmarsh [185]. Bieniasz and Britz [177] found the routine `STINT` [186] more efficient than `STIFF3`.

Another possible approach, apparently not taken by any electrochemical simulator, is to render the equation set into a DAE set. Instead of substituting for $C_0$ as described above, one replaces the first equation of (9.91) by the algebraic equation for the boundary condition, and uses one of the available DAE packages to solve the system.

Both the approaches described above, that is, substituting for boundary concentrations, or adding algebraic equations to express boundary conditions, can be applied to more complex mechanisms involving more than one species, including coupled systems. With the latter, there is probably not much to be gained by the block-matrix method, because of the number of coefficients on each line.

The present authors have used the simple second-order extrapolation technique to proceed from (9.94), and this simple approach led to highly accurate results. Using just 10 points (8 internal points) and 100 steps in time for a Cottrell simulation, the current was in error by only 0.01 %. Using only 3 internal points, there was a 10 % error in the current. These are remarkable results. For this, second-order extrapolation was used in combination with BI.

### 9.6.1 Current Calculation with OC

The normalised current, that is the gradient $G$, is given by using matrix $\mathbf{V}$ and (9.86). The operation returns gradients at all collocation points, and one just takes the first of these, which refers to $X = 0$. Alternatively, one can multiply just the top row of $\mathbf{V}$ with the concentration vector $\mathbf{C}$, which gives $\mathrm{d}C/\mathrm{d}X(X = 0)$ directly. Note that this is not our usual $G$ yet, because of the way $X$ is normalised here. Regarding (9.76), clearly,

$$G = \frac{1}{f}\frac{\mathrm{d}C}{\mathrm{d}X}(X = 0)\ . \tag{9.95}$$

Alternatively, one can simply work with the $X$-scaling as it is, and change the analytical solution correspondingly. For example, for the Cottrell experiment, with the usual normalisation, the gradient $G$ at time $T$ has the analytical solution $1/\sqrt{\pi T}$ (see (2.44) on page 19) while, with the normalisation as used in OC (9.76), the analytical solution for the gradient at the electrode (Eq. (2.36), page 16) becomes

$$G = \left.\frac{\partial C}{\partial X}\right|_{X=0} = f\frac{1}{\sqrt{\pi T}}\ . \tag{9.96}$$

In the case of LSV, however, not dividing by the factor $f$ would lead to currents that are hard to compare with tabled values or values one expects.

### 9.6.2 A Numerical Example

For those wanting to try OC, here is a guide for checking the work. This follows the example given by Whiting and Carr [167].

Assume a Cottrell simulation and the use of only five points, giving just three internal points. From Table A.5, this places the internal points at the positions (0.1127, 0.5000, 0.8873), here presenting fewer digits than in the table. Using equation sets (9.81)–(9.83) and the definitions (9.84), we then have

$$\mathbf{Q} = \begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 1.0000 & 0.1127 & 0.0127 & 0.0014 & 0.0002 \\ 1.0000 & 0.5000 & 0.2500 & 0.1250 & 0.0625 \\ 1.0000 & 0.8873 & 0.7873 & 0.6986 & 0.6198 \\ 1.0000 & 1.0000 & 1.0000 & 1.0000 & 1.0000 \end{bmatrix} \tag{9.97}$$

$$\mathbf{R} = \begin{bmatrix} 0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.2254 & 0.0381 & 0.0057 \\ 0.0000 & 1.0000 & 1.0000 & 0.7500 & 0.5000 \\ 0.0000 & 1.0000 & 1.7746 & 2.3619 & 2.7943 \\ 0.0000 & 1.0000 & 2.0000 & 3.0000 & 4.0000 \end{bmatrix} \tag{9.98}$$

and

$$\mathbf{S} = \begin{bmatrix} 0.0000 & 0.0000 & 2.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 2.0000 & 0.6762 & 0.1524 \\ 0.0000 & 0.0000 & 2.0000 & 3.0000 & 3.0000 \\ 0.0000 & 0.0000 & 2.0000 & 5.3238 & 9.4476 \\ 0.0000 & 0.0000 & 2.0000 & 6.0000 & 12.0000 \end{bmatrix} . \tag{9.99}$$

From these, using (9.88) and (9.89), we obtain

$$\mathbf{V} = \begin{bmatrix} -13.0000 & 14.7883 & -2.6667 & 1.8784 & -1.0000 \\ -5.3238 & 3.8730 & 2.0656 & -1.2910 & 0.6762 \\ 1.5000 & -3.2275 & 0.0000 & 3.2275 & -1.5000 \\ -0.6762 & 1.2910 & -2.0656 & -3.8730 & 5.3238 \\ 1.0000 & -1.8784 & 2.6667 & -14.7883 & 13.0000 \end{bmatrix} \tag{9.100}$$

and

$$\mathbf{W} = \begin{bmatrix} 84.0000 & -122.0632 & 58.6667 & -44.6035 & 24.0000 \\ 53.2379 & -73.3333 & 26.6667 & -13.3333 & 6.7621 \\ -6.0000 & 16.6667 & -21.3333 & 16.6667 & -6.0000 \\ 6.7621 & -13.3333 & 26.6667 & -73.3333 & 53.2379 \\ 24.0000 & -44.6035 & 58.6667 & -122.0632 & 84.0000 \end{bmatrix} . \tag{9.101}$$

Matrix **V** is needed to generate $G$, and **W** is now stripped of its outer frame to produce (9.94),

$$\frac{\partial \mathbf{C}}{\partial T} = \frac{1}{f^2} \begin{bmatrix} -73.3333 & 26.6667 & -13.3333 \\ 16.6667 & -21.3333 & 16.6667 \\ -13.3333 & 26.6667 & -73.3333 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix} + \begin{bmatrix} 6.7621 \\ -6.0000 \\ 53.2379 \end{bmatrix} + F(\mathbf{C}) . \tag{9.102}$$

The above set of *ode*s is now solved, choosing some algorithm. Nothing has been specified about the homogeneous chemical reaction function $F(\mathbf{C})$, but it will add terms to the matrix $\mathbf{W}'$ when specified. After the time derivative is discretised in some way, the equation can be rearranged into the same form as described in Chap. 8 and solved using the same methods or, as mentioned above, solved using a professional *ode* or DAE solver.

## 9.7 Eigenvalue–Eigenvector Method

Yet another, quite different, approach to solving a system of *ode*s, such as one obtains as an intermediate step when using, for example, MOL or OC, is the eigenvalue–eigenvector method. Its use for electrochemical simulations was described in two papers: Friedrichs et al. [187] in 1989 and Kavanaugh et al. [188] in 1990. The method has some drawbacks, and does not appear to have seen much use since these two papers (but see below). It does have one unique feature: there is no discretisation of time. A solution is generated by the algorithm, at any chosen time. So, although the method may at times be fairly inefficient, if one wants a current or concentrations at only one or a few time points, this could be faster than a time march with the usually small time intervals.

The method is also described rather clearly by Smith [95], whose description will be followed here.

The method starts with a system of *ode*s, represented as in (9.53),

$$\frac{\mathrm{d}\mathbf{C}}{\mathrm{d}T} = \mathbf{AC} \,, \tag{9.103}$$

simplified here so that the vector coming from boundary conditions is not included. It can be included but then the argument is less focussed. There is only one boundary condition,

$$\mathbf{C}\ (T = 0) = \mathbf{C}(0) \,. \tag{9.104}$$

Instead of now discretising the left-hand side of the equation in some way (explicit BI, CN, etc.) and stepping forward in time by small time intervals, the equation is solved analytically; the solution at time $T$ is

$$\mathbf{C}(T) = \exp(T\mathbf{A})\,\mathbf{C}(0) \,. \tag{9.105}$$

This has an exponential of a matrix. It is defined in terms of the expansion of the exponential function, see, for example, Smith [95, pp. 134–135]. Now, the usual eigenvalue–eigenvector equation can be written in compact form,

$$\mathbf{AX} = \mathbf{XD} \,, \tag{9.106}$$

where $\mathbf{D}$ is the diagonal matrix containing all the eigenvalues and $\mathbf{X}$ is the matrix containing the eigenvectors corresponding to the eigenvalues. $\mathbf{X}$ is also called the modal matrix of $\mathbf{A}$. $\mathbf{X}$ and $\mathbf{D}$ can be computed by available subroutines, and Friedrichs et al. describe very efficient ways of calculating them [187, 188]. The method rests crucially on these values.

Equation (9.106) can be written explicitly for $\mathbf{D}$:

$$\mathbf{X}^{-1}\mathbf{AX} = \mathbf{D} \,. \tag{9.107}$$

Smith also shows that it follows from this that

$$\mathbf{X}^{-1}\exp(\mathbf{A})\mathbf{X} = \exp(\mathbf{D}) \tag{9.108}$$

which will be useful below.

Let a new matrix $\mathbf{Y}(T)$ be defined by

$$\mathbf{C}(T) = \mathbf{X}\mathbf{Y}(T) \tag{9.109}$$

with the time-dependent vector $\mathbf{Y}$ as yet unknown. We can however solve for $\mathbf{Y}(0)$ by setting $T = 0$ in (9.109), since we know $\mathbf{X}$ and the initial condition $\mathbf{C}(0)$. Once we have, for any other $T$, the vector $\mathbf{Y}(T)$, it can be used, via (9.109), to compute the desired vector $\mathbf{C}$, by multiplication with $\mathbf{X}$.

Combining (9.105) with (9.109), we can write

$$\mathbf{X}\mathbf{Y}(T) = \exp(T\mathbf{A})\,\mathbf{X}\mathbf{Y}(0) \tag{9.110}$$

and, multiplying by $\mathbf{X}^{-1}$,

$$\mathbf{Y}(T) = \mathbf{X}^{-1}\exp(T\mathbf{A})\mathbf{X}\mathbf{Y}(0)\ . \tag{9.111}$$

Equation (9.108) allows us to write this as

$$\mathbf{Y}(T) = \exp(T\mathbf{D})\mathbf{Y}(0)\ . \tag{9.112}$$

Here we have an exponential of the matrix $T\mathbf{D}$. The matrix is zero except on the diagonal (containing the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_N$), and, as Smith proves, this and the definition of a matrix exponential lead to the simple result that

$$\exp(T\mathbf{D}) = \begin{bmatrix} \exp(T\lambda_1) & & & & \\ & \exp(T\lambda_2) & & & \\ & & \exp(T\lambda_3) & & \\ & & & \ddots & \\ & & & & \exp(T\lambda_N) \end{bmatrix}, \tag{9.113}$$

so that finally the solution is

$$\begin{bmatrix} C_1(T) \\ C_2(T) \\ \vdots \\ C_N(T) \end{bmatrix} = \mathbf{X}\begin{bmatrix} Y_1(0)\,\exp(T\lambda_1) \\ Y_2(0)\,\exp(T\lambda_2) \\ \vdots \\ Y_N(0)\,\exp(T\lambda_N) \end{bmatrix}. \tag{9.114}$$

This entails matrix multiplication, and Smith suggests an approximation. The eigenvalues $\lambda_j$ are negative and of increasing magnitude as $j$ increases, so if $T$ is not too small, the top term of the right-hand vector is dominating, and can in itself produce a good approximation to the solution. This has the drawback that no reliable solution can be found for small $T$, and Friedrichs et al. [187] do not suggest this approximation but rather, a more efficient way to calculate the eigenvalues and -vectors. Their solutions are fairly accurate.

As mentioned, the procedure has the advantage that the time variable $T$ is part of the solution expression, so that if solutions at only a few time values, or even just one such $T$, are sought, the method might be competitive with the more usual time-marching schemes. Also, although the above description has been simplified by leaving out the boundary condition vector in (9.103), its addition still leaves the method intact. As shown in the second paper by Kavanaugh et al. [188], LSV simulations and quasireversible systems can be handled. For some reason, however, the method has not seen any use in electrochemistry since these two seminal papers, with the exception of one study, comparing several methods of simulating the ultramicrodisk electrode, to find out what is most efficient [31]. The eigenvector, -value method did not generally turn out very efficient, although each evaluation at a given time, once the eigenvectors and -values have been computed, is quite fast. A curious result of the study was waves in the errors of the current computed at a number of times, with some two-dimensional transformations but not all (see Chap. 12 for the transformations). They were called waves rather than oscillations, which (as with CN) arise from error propagation, because there is no propagation in this method, each computation at a new time value being independent of all others. It has not been found out where these waves come from. They decrease with time values, and also with increasingly fine spatial meshes.

One interesting application of the solution (9.105) for (9.103) is the solution for a single step in time of size $\delta T$. As pointed out by Smith [95, p. 117], we can write (in our notation)

$$\mathbf{C}' = \exp(\delta T\mathbf{A})\mathbf{C}\,. \tag{9.115}$$

This seems to go back to Lawson [189]. By approximating the matrix exponential by a Padé approximant, of which there are many, one can derive all the usual simulation methods. This is described in detail in Smith, and was used by Strutwolf and Schoeller [190] to analyse the extrapolation method, introduced by these authors to electrochemistry. According to which Padé approximant one uses, the explicit, BI or CN can be generated, and there are other, unexplored, possibilities that may be worth studying. Momoniat, in another context, used MATHEMATICA [191] to express the matrix exponential without approximation and derived what he calls the Lawson-Euler and Lawson-RK schemes [192], and other schemes are possible, and are left to the reader.

## 9.8 Integral Equation Method

This is an alternative method of simulation. Bieniasz, who has worked extensively on the method [193–210] has published a thorough monograph of the method [211].

One begins with an attempt at an analytical solution, by means of the Laplace transformation, see also such standard texts such as Bard and Faulkner [212] or Galus [213] for a description of this procedure. The result, after back transformation, is an integral equation, which must be solved numerically. Regarding the description of the mathematics of LSV in Chap. 2, from page 27, and taking the example of a simple reversible system, the result is the equation, for the normalised current (Randles–Ševčík function $\chi(z)$),

$$\int_0^{at} \frac{\chi(z)}{\sqrt{at-z}} \mathrm{d}z = \frac{1}{1+\xi\Theta S(at)} , \tag{9.116}$$

where $\xi = (D_O/D_R)^{\frac{1}{2}}$, $\Theta = \exp\left\{\frac{n\mathcal{F}}{\mathcal{RT}}(E_i - E^0)\right\}$ and $S(at) = \exp(-at)$, $at$ being the normalised time variable and $E_i$ the initial potential at the start of the sweep. The equation is that arising from the simple reversible system, but other systems lead to equations of similar form, all Volterra equations.

In descriptions of this problem, the names of Randles [214] and Ševčík [215] are prominent. They both worked on the problem and reported their work in 1948. Randles was in fact the first to do electrochemical simulation, as he solved this system by explicit finite differences (and using a three-point current approximation), referring to Emmons [216]. Ševčík attempted to solve the system analytically, using two different methods. The second of these was by Laplace transformation, which today is the standard method. He arrived at (9.116) and then applied a series approximation for the current. Galus writes [213] that there was an error in a constant. Other analytical solutions were described (see Galus [213] and Bard and Faulkner [217] for references), all in the form of series, which themselves require quite some computation to evaluate. So the direct approximation of equations like (9.116) was an obvious step. This was taken in the classic paper of Nicholson and Shain [218], and continued, by Nicholson [219] and Nicholson and Olmstead [220] for systems other than the simple reversible. These used what is called the Huber method [221], integrating by summing a number of intervals into which the limit $at$ is divided. If there are $N$ intervals, the Huber method gives rise to a triangular system of $N$ simultaneous equations, which requires of the order $N^2$ operations for the solution. Bieniasz devised a better method [193, 194] that requires only of the order of $N$ operations. In an example using both the Huber and his new method, the Huber method required 39 min for a computation, while the improved method did the same in 0.13 min. This method is built into Bieniasz' simulation package ELSIM [222]. In his 1992 paper, he also points out the mathematical relation between the integral equation above, and the process of semi-integration, described by Oldham [223], for which there are also more and less efficient algorithms. Bieniasz lists [194] a table of the forms of the integral equation for a number of systems. More recently, Mirčeski has published an approximation to the integral, separating the current function out as a sum, which he claims is an efficient method of solving these equations. The interested reader is referred to the monograph by Bieniasz [211] for (many) further details.

We do not go into any detail of the integration methods here, as it seems to us that direct finite difference methods are preferable.

## 9.9 The Network Method

Since about 1989, Horno and coworkers have published a series of papers on their "network thermodynamic method" of simulation. Only a few of these will be cited here. In the first, the 1989 work, the method is described [224], and again in 1992–4 [225–227], adding cyclic voltammetry. In the 1994 paper [227], there is a good description of the method, and an indication of how it can be adapted to a multitude of different electrochemical systems. A Chinese group has also used this method [228–231], as well as some others [232, 233]. Moya et al. [234–240] continue with the work. Curiously, the method was suggested as early as 1946 for heat flow simulations by Eyres et al. [241, p. 31], referring to even earlier precursors.

It is all done by modelling derivatives, fluxes and homogeneous chemical reactions as electrical elements and current sources, and applying Kirchhoff's Law to them. After conversion to an analogue of an electric circuit, the standard package SPICE or PSPICE then does the rest [242, 243], using Gear's package for solving *ode*s [184]. The main work for the simulator is thus the translation of the governing electrochemical equations into an electrical network and specifying it to the packages.

Very briefly, basing the description on [227], the diffusion-reaction equation of the form of (9.74) is semidiscretised as in MOL, to

$$\frac{\mathrm{d}C}{\mathrm{d}t} = \frac{D}{h^2}(C_{i-1} - C_i) + \frac{D}{h^2}(C_i - C_{i+1}) + f(C)\ . \tag{9.117}$$

Then both sides are multiplied by the spatial interval $h$, and the result expressed, term by term, as

$$J_{\gamma i} = J_{i-1} - J_i + J_{Gi} \tag{9.118}$$

with

$$J_{\gamma i} = h\frac{\mathrm{d}C_i}{\mathrm{d}t}\ , \tag{9.119}$$

regarded as a capactive flux due to "capacity" $h$ and "voltage" $C_i$. The two terms $J_i$ and $J_{i-1}$ are regarded as resistive fluxes due to "resistance" $h/D$ and again "voltage" $C$, and finally $J_{Gi}$ corresponds to the homogeneous reaction term, seen as a current source, which might depend on one or more “voltages” (concentrations), depending on the reaction. These elements are then arrayed in a suitable manner in a ladder network, and the input to SPICE or PSPICE is designed for that. It seems that this process of translation into a sequence of specifications to SPICE (which Horno indeed calls a program) is the main work, and appears to the present authors to be rather indirect and cumbersome. Probably workers familiar with the method, as the Horno school is, have a different view. The method has been successfully applied to such difficult systems as catalytic second-order reactions [244], oscillating reaction-diffusion systems [245], the square scheme [225, 246], to migration problems [234, 235] and steady state colloidal systems, solving for potential fields [247]. The list of papers is only partial.

The method has not taken on elsewhere. One paper [248] reports the use of PSPICE, but for simulating actual resistance in an electrolyte, modelled as a resistance network. This is quite a different application, and much more directly relevant.

## 9.10 Treanor Method

In a paper reporting the results of some simulations of diffusion of hydrogen into palladium [249], the authors describe their method of solution as the Treanor method. This is described in a few texts [250, 251] and goes back to a paper by Treanor in 1966 [252].

The method is one way to handle a stiff set of *ode*s, and is an extension of fourth-order explicit Runge–Kutta. The function to be solved is approximated over the next time interval by a combination of a linear function of the dependent variable and a quadratic function of time (assuming that it is strongly time-dependent) and this increases the accuracy and stability of the fourth-order Runge–Kutta method considerably. Today, however, we have other methods of dealing with stiff sets of *ode*s, so this method might be said to have outlived its usefulness.

## 9.11 Monte Carlo Method

Diffusion is at base a process due to randomly moving particles, so it might be logical to model or simulate it as such. This has been done in a few works. Fanelli et al. [253, 254] thus simulated adsorption processes and fractal structures, using a method described earlier by Voss and Tomkiewicz [255] (used to study dendrite formation). Licht et al. [256] simulated concentration profiles around arrays of generator-collector microbands. Borkowski and Stojek simulated a CV at a microelectrode [257], referring to “diffusional noise”. Up to 35,000 particles were let loose to do a random walk. The result was a very rough but recognisable CV. Baur and Motsegood simulated a pair of coplanar disks [258]. Juwono et al.

used the method for lateral diffusion of adsorbed species [259], and there are a few publications from the Compton school on random walk simulation [125, 260] and a few others [261–263]. Nagy et al. [264, 265] studied, among other systems, growing spheres and a hemisphere array, and Yang et al. used it on microdialysis [266]. This is interesting but hardly appears very useful, having no obvious advantages over other methods and being presumably somewhat time-consuming, although Baur and Motsegood argue for its use [258].

## References

1. Feldberg SW (1969) Digital simulation: a general method for solving electrochemical diffusion-kinetic problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 3. Marcel Dekker, New York, pp 199–296
2. Rudolph M, Reddy DP, Feldberg SW (1994) A simulator for cyclic voltammetry responses. Anal Chem 66:589A–600A
3. Seeber R, Stefani S (1981) Explicit finite difference method in simulating electrode processes. Anal Chem 53:1011–1016
4. Feldberg SW (1981) Optimization of explicit finite-difference simulation of electrochemical phenomena utilizing an exponentially expanded space grid. Refinement of the Joslin-Pletcher algorithm. J Electroanal Chem 127:1–10
5. Rudolph M (1995) Digital simulations with the fast implicit finite difference algorithm: the development of a general simulator for electrochemical processes. In: Rubinstein I (ed) Physical electrochemistry. Marcel Dekker, New York, pp 81–129
6. Rudolph M (2002) Digital simulation on unequally spaced grids. Part 1. Critical remarks on using the point method by discretisation on a transformed grid. J Electroanal Chem 529:97–108
7. Rudolph M (2003) Digital simulations on unequally spaced grids. Part 2. Using the box method by discretisation on a transformed equally spaced grid. J Electroanal Chem 543:23–39
8. Rudolph M (2004) Digital simulations on unequally spaced grids. Part 3. Attaining exponential convergence for the discretisation error of the flux as a new strategy in digital simulations of electrochemical experiments. J Electroanal Chem 571:289–307
9. Rudolph M (2003) Reply to L.K. Bieniasz's comments on my paper [J Electroanal Chem 529:97 (2002)]. J Electroanal Chem 558:171–176
10. Bieniasz LK (2003) High order accurate one-sided finite-difference approximations to gradients at the boundaries, for the simulation of electrochemical reaction-diffusion problems in one-dimensional space geometry. Comput Biol Chem 27:315–325
11. Patankar SV (1980) Numerical heat transfer and fluid flow. Hemisphere Publishing Corp., New York
12. Bacha S, Bergel A, Comtat M (1993) Modelling of amperometric biosensors by a finite-volume method. J Electroanal Chem 359:21–38
13. Juozėnas A, Šidlauskas V, Jurevičius D (1993) Chronopotentiometry on partially blocked electrodes. Digital simulation. Chemija 13–17
14. Kimble MC, White RE (1990) A five-point finite difference method for solving parabolic differential equations. Comput Chem Eng 14:921–924
15. Nguyen TV, White R (1987) A finite difference procedure for solving coupled, nonlinear elliptic partial differential equations. Comput Chem Eng 11:543–546
16. Newman J (1973) Electrochemical systems. Prentice-Hall, Englewood Cliffs, NJ

17. Richardson LF (1911) The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. Philos Trans R Soc Lond Ser A 210:307–357
18. O'Brien GG, Hyman MA, Kaplan S (1950) A study of the numerical solution of partial differential equations. J Math Phys 29:223–251
19. Britz D (1999) An interesting global stabilisation of a locally short-range unstable high-order scheme for the digital simulation of the diffusion equation. Comput Chem Eng 23:297–300
20. Collatz L (1960) Numerische Behandlung von Differentialgleichungen. Springer, Heidelberg
21. Feldberg SW, Goldstein CI (1995) Examination of the behavior of the fully implicit finite-difference algorithm with the Richtmyer modification: behavior with an exponentially expanding time grid. J Electroanal Chem 397:1–10
22. Britz D, Strutwolf J (2000) Higher-order spatial discretisations in electrochemical digital simulation. 1. Combination with the BDF algorithm. Comput Chem 24:673–684
23. Strutwolf J, Britz D (2001) Use of high-order discretisations in digital simulation. 2. Combination with the extrapolation algorithm. Comput Chem 25:511–520
24. Britz D, Strutwolf J, Thøgersen L (2001) Investigation of some starting protocols for BDF (FIRM) in electrochemical digital simulation. J Electroanal Chem 512:119–123
25. Britz D, Østerby O, Strutwolf J, Svennesen TK (2002) High-order spatial discretisations in electrochemical digital simulations. 3. Combination with the explicit Runge-Kutta algorithm. Comput Chem 26:97–103
26. Britz D, Strutwolf J (2003) Higher-order spatial discretisations in electrochemical digital simulation. Part 4. Discretisation on an arbitrarily spaced grid. Comput Biol Chem 27:327–337
27. Engeln-Müllges G, Uhlig F (1996) Numerical algorithms with Fortran. Springer, Berlin, Heidelberg
28. Fletcher CAJ (1991) Computational techniques for fluid dynamics, vol I, 2nd edn. Springer, Berlin
29. Britz D (2003) Higher-order spatial discretisations in digital simulations. Algorithm for any multi-point first- or second derivative on an arbitrarily spaced grid. Electrochem Commun 5:195–198
30. Strutwolf J, Britz D (2004) Higher-order discretisations in electrochemical digital simulation. Part 5. Application to stationary ultramicrodisk electrode simulation. J Electroanal Chem 566:15–23
31. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376
32. Strikwerda JC (1989) Finite difference schemes and partial differential equations. Wadsworth and Brooks/Cole, Pacific Grove, CA
33. DuFort EC, Frankel SP (1953) Stability conditions in the numerical treatment of parabolic differential equations. Math Tables Aids Comput 7:135–152
34. Lapidus L, Pinder GF (1982) NumeriLng. Wiley, New York
35. Bond AM, Mahon PJ (1997) Linear and non-linear analysis using the Oldham- Zoski steady-state equation for determining heterogeneous electrode kinetics at microdisk electrodes and digital simulation of the microdisk geometry with the fast quasi-explicit finite difference method. J Electroanal Chem 439:37–53
36. Feldberg SW (1990) A fast quasi-explicit finite difference method for simulating electrochemical phenomena. Part I. Application to cyclic voltammetric problems. J Electroanal Chem 290:49–65
37. Lerke SA, Evans DH, Feldberg SW (1990) Digital simulation of the square scheme in cyclic voltammetry. A comparison of methods. J Electroanal Chem 296:299–315
38. Marques da Silva B, Avaca LA, Gonzalez ER (1989) New explicit finite difference methods in the digital simulation of electrochemical problems. J Electroanal Chem 269:1–14
39. Feldberg SW (1987) Propagational inadequacy of the hopscotch finite difference algorithm: the enhancement of performance when used with an exponentially expanding grid for simulation of electrochemical diffusion problems. J Electroanal Chem 222:101–106

40. Carnahan B, Luther HA, Wilkes JO (1969) Applied numerical methods. Wiley, NY
41. Potter D (1973) Computational physics. Wiley, London
42. Saul'yev VK (1964) Integration of equations of parabolic type by the method of nets. Pergamon Press, New York
43. Barakat HZ, Clark JA (1966) On the solution of the diffusion equations by numerical methods. Trans ASME J Heat Transfer 421–427
44. Britz D, da Silva BM, Avaca LA, Gonzales ER (1990) The Saul'yev method of digital simulation under derivative boundary conditions. Anal Chim Acta 239:87–93
45. Evans DJ, Abdullah ARB (1983) Group explicit methods for parabolic equations. Int J Comput Math 14:73–105
46. Liu SL (1967) Numerical solution of two-point boundary value problems in simultaneous second-order nonlinear ordinary differential equations. Chem Eng Sci 22:871–881
47. Liu SL (1969) Stable explicit difference approximations to parabolic partial differential equations. AICHE J 15:334–338
48. Larkin BK (1964) Some stable explicit difference approximations to the diffusion equation. Math Comput 18:196–202
49. Marques da Silva B, Avaca LA, Gonzalez ER (1988) On the use of the Saul'yev algorithms in the digital simulation of electrochemical processes. J Electroanal Chem 250:457–460
50. Bieniasz LK, Britz D (1993) Electrochemical kinetic simulations of mixed diffusion/homogeneous reaction problems by the Saul'yev finite difference algorithms. Anal Chim Acta 278:59–70
51. Bieniasz LK, Østerby O, Britz D (1995) Numerical stability of the Saul'yev finite difference algorithms for electrochemical kinetic simulations: matrix stability analysis for an example problem involving mixed boundary conditions. Comput Chem 19:357–370
52. Deng ZX, Lin XQ, Tong ZH (2002) Single alternating group explicit (SAGE) method for electrochemical finite difference digital simulation. Chin J Chem 20:252–262
53. Gordon P (1965) Nonsymmetric difference equations. J Soc Ind Appl Math 13:667–678
54. Gourlay AR (1970) Hopscotch: a fast second-order partial differential equation solver. J Inst Math Appl 6:375–390
55. Evans NTS, Gourlay AR (1977) The solution of a two-dimensional time-dependent diffusion problem concerned with oxygen metabolism in tissues. J Inst Math Appl 19:239–251
56. Gourlay AR, McGuire GR (1971) General hopscotch algorithm for the numerical solution of partial differential equations. J Inst Math Appl 7:216–227
57. Feldberg SW, Bowers ML, Anson FC (1986) Hopscotch-finite-difference simulation of the rotating ring-disc electrode. J Electroanal Chem 215:11–28
58. Shoup D, Szabo A (1982) Chronoamperometric current at finite disk electrodes. J Electroanal Chem 140:237–245
59. Shoup D, Szabo A (1984) Hopscotch: an algorithm for the numerical solution of electrochemical problems. J Electroanal Chem 160:1–17
60. Shoup D, Szabo A (1984) Chronoamperometry at an ensemble of microdisk electrodes. J Electroanal Chem 160:19–26
61. Shoup D, Szabo A (1984) Influence of insulation geometry on the current at microdisk electrodes. J Electroanal Chem 160:27–31
62. Amatore CA, Fosset B (1992) Space variables well fitted for the study of steady state and near-steady-state diffusion at a microdisk. J Electroanal Chem 328:21–32
63. Fosset B, Amatore CA, Bartelt JE, Michael AC, Wightman RM (1991) Use of conformal maps to model the voltammetric response of collector-generator double-band electrodes. Anal Chem 63:306–314
64. Fosset B, Amatore CA, Bartelt JE, Wightman RM (1991) Theory and experiment for the collector-generator triple-band electrode. Anal Chem 63:1403–1408
65. Lavagnini I, Pastore P, Magno F, Amatore CA (1991) Performance of a numerical method based on the hopscotch algorithm and on an oblate spheroidal space coordinate- expanding time grid for simulation of voltammetric curves at an inlaid disk microelectrode. J Electroanal Chem 316:37–47

66. Lavagnini I, Pastore P, Magno F (1992) Application of cyclic voltammograms under mixed spherical/semi-infinite linear diffusion at microdisk electrodes for measurement of fast electrode kinetics. J Electroanal Chem 333:1–10
67. Michael AC, Wightman RM, Amatore CA (1989) Microdisk electrodes. Part 1. Digital simulation with a conformal map. J Electroanal Chem 267:33–45
68. Pastore P, Magno F, Lavagnini I, Amatore C (1991) Digital simulation via the hopscotch algorithm of a microelectrode-based channel flow-through amperometric detector. J Electroanal Chem 301:1–13
69. Safford LK, Weaver MJ (1991) Cyclic voltammetric wave-shapes for microdisk-electrodes: coupled effects of solution resistance, double-layer capacitance, and finite electrochemical kinetics. J Electroanal Chem 312:69–96
70. Ružić I (1986) Comments on the paper "Hopscotch: an algorithm for the numerical solution of electrochemical problems" by Shoup and Szabo. J Electroanal Chem 199:431–435
71. Sandifer JR, Buck RP (1974) Improvements in digital simulation. J Electroanal Chem 49:161–170
72. Shoup D, Szabo A (1986) Explicit hopscotch and implicit finite-difference algorithms for the Cottrell problem: exact analytical results. J Electroanal Chem 199:437–441
73. Gaidamauskaite E, Baronas R (2007) A comparison of finite difference schemes for computational models of biosensors. Nonlinear Anal Modell Control 12:359–369
74. Feldberg SW, Auerbach C (1964) Model for current reversal chronopotentiometry with second-order kinetic complications. Anal Chem 36:505–509
75. Flanagan JB, Marcoux L (1973) Digital simulation of edge effects at planar disc electrodes. J Phys Chem 77:1051–1055
76. Nielsen MF, Almdal K, Hammerich O, Parker VD (1987) The application of Runge-Kutta integration in digital simulation of electroanalytical experiments. An accurate treatment of the homogeneous kinetics. Acta Chem Scand A 41:423–440
77. Ružić I, Britz D (1991) Consistency proof of the sequential algorithm for the digital simulation of systems involving first-order homogeneous kinetics. Acta Chem Scand 45:1087–1089
78. Britz D (1988) Electrochemical digital simulation by Runge-Kutta integration. J Electroanal Chem 240:17–26
79. Lemos MANDA, Pombeiro AJL (1992) Digital simulation of cyclic voltammetry - influence of the space discretization technique. Port Electrochim Acta 10:89–99
80. Lemos MANDA, Pombeiro AJL (1993) A comparative study of numerical methods for cyclic voltammetry digital simulation of an electrochemical process with a coupled chemical reaction. In: Pombeiro AJL, McCleary JA (eds) Molecular electrochemistry of inorganic, bioinorganic and organometallic compounds. Kluwer Academic, Dordrecht, pp 477–482
81. Lemos MA (1997) Digital simulation for electrochemical processes. The Sherlock Holmes magnifying glass. Port Electrochim Acta 15:163–187
82. Lemos MANDA, Lemos F, Papadopoulos N, Pombeiro AJL (1998) Virtual cyclic voltammetry. Port Electrochim Acta 16:175–180
83. Ribeiro LMD, Lemos MANDA, Pombeiro AJL, Sobota P (1995) Electrochemical study of some chloro complexes of titanium, molybdenum, iron aluminium or tin in high oxidation states. Russ J Electrochem 31:1009–1015
84. Gosser DK, Rieger PH (1988) Treatment of homogeneous kinetics in electrochemical digital simulation programs. Anal Chem 60:1159–1167
85. Barker PD, Hill HAO, Walton NJ (1989) Fast second order electron transfer reactions coupled to redox protein electro-chemistry. Experiment and digital simulation. J Electroanal Chem 260:303–326
86. Hayase M, Hatsuzawa T, Fukuizumi A (2002) Electric field analysis in a dilute solution for the vibrating electrode technique. J Electroanal Chem 537:173–181
87. Britz D, Nielsen MF (1991) Accuracy contours in $(n_T, \lambda)$ space in electrochemical digital simulations. Collect Czechoslov Chem Commun 56:20–41

88. Bieniasz LK (1993) The von Neumann stability of finite-difference algorithms for the electrochemical kinetic simulation of diffusion coupled with homogeneous reactions. J Electroanal Chem 345:13–25
89. Hairer E, Nørsett SP, Wanner G (1987) Solving ordinary differential equations I. Nonstiff problems. Springer, Berlin
90. Hairer E, Wanner G (1991) Solving ordinary differential equations II. stiff and differential-algebraic problems. Springer, Berlin
91. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
92. Bieniasz LK, Britz D (2001) Chronopotentiometry at a microband electrode: simulation study using a Rosenbrock time integration scheme for differential-algebraic equations and a direct sparse solver. J Electroanal Chem 503:141–152
93. Kopal Z (1955) Numerical analysis. Chapman & Hall, London
94. Noumerov BV (1924) A method of extrapolation of perturbations. Mon Not R Astron Soc 84:592–601
95. Smith GD (1985) Numerical solution of partial differential equations, 3 edn. Oxford University Press, Oxford
96. Bieniasz LK (2003) Comments on the paper by M. Rudolph, entitled “Digital simulations on unequally spaced grids. Part 1. Critical remarks on using the point method by discretisation on a transformed grid” [J Electroanal Chem 529:97 (2002)]. J Electroanal Chem 558:167–170
97. Bieniasz LK (2004) Improving the accuracy of the spatial discretisation in finite-difference electrochemical kinetic simulations, by means of the extended Numerov method. J Comput Chem 25:1075–1083
98. Wu B, White RE (2004) One implementation variant of the finite difference method for solving ODEs/DAEs. Comput Chem Eng 28:303–309
99. Hartree DR (1958) Numerical analysis. Oxford University Press, Oxford
100. Fox L (1962) Parabolic equations in two dimensions. II. In: Fox L (ed) Numerical solution of ordinary and partial differential equations. Pergamon Press, Oxford, pp 242–254
101. Kantorowitsch LW, Krylow WI (1956) Näherungsmethoden der höheren Analyse. VEB Deutscher Verlag der Wissenschaften, Berlin
102. Kantorovich LV (1934) On a method of approximate solution of partial differential equations. Doklady Akad Nauk 2:532–536. In Russian, with a French translation added
103. Liskovets OA (1965) Metod pryamykh [method of lines]. Diff Urav 1:1662–1677
104. Rothe E (1930) Zweidimensionale parabolische Randwertaufgaben als Grenzfall eindimensionaler Randwertaufgaben. Math Ann 102:651–670
105. Hartree DR, Womersley JR (1937) A method for the numerical or mechanical solution of certain types of partial differential equations. Proc R Soc Lond A 161:353–367
106. Schiesser WE (1991) The numerical method of lines integration of partial differential equations. Academic Press, San Diego
107. Lasia A, Grégoire D (1995) General model of electrochemical hydrogen absorption into metals. J Electrochem Soc 142:3393–3399
108. Zhang Y, Cheh HY (1999) Modelling of cylindrical alkaline cells. VIII. Solution of the model by exploiting its differential algebraic equation structure. J Electrochem Soc 146:850–856
109. Brenan KE, Campbell SL, Petzold LR (1996) Numerical solution of initial-value problems in differential-algebraic equations. SIAM, Philadelphia
110. Petzold L (1983) A description of DASSL - a differential/algebraic system solver. In: Stepleman RS, Carver M, Peskin R, Ames WF, Vichnevetsky R (eds) Scientific computing, volume 1, IMACS Trans. Sci. Comp., 10th IMACS world congress on systems simulation and scientific computation, Montreal, Canada, August 1982. North Holland, Amsterdam, pp 65–68
111. http://www.netlib.org
112. Lang J (2001) Adaptive multilevel solution of nonlinear parabolic PDE systems. Springer, Berlin

113. Ludwig K, Speiser B (2006) EChem++ - an object-oriented problem solving environment for electrochemistry: part 4. Adaptive multilevel finite elements applied to electrochemical models. Algorithm and benchmark calculations. J Electroanal Chem 588:74–87
114. Ferrigno R, Brevet PF, Girault HH (1997) Finite element simulation of the chronoamperometric response of recessed and protruding microdisc electrodes. Electrochim Acta 42:1895–1903
115. Nann T (1997) Digitale Simulation in der Elektrochemie mit der Methode der Finiten Elementen. Ph.D. thesis, Albert-Ludwigs-Universität zu Freiburg im Breisgau. Publ. by Shaker Verlag, Aachen
116. Nann T, Heinze J (1999) Simulation in electrochemistry using the finite element method. Part 1. The algorithm. Electrochem Commun 1:289–294
117. Harriman K, Gavaghan DJ, Houston P, Kay D, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. *ECE* and $EC_2E$ mechanisms at channel microband electrodes. Electrochem Commun 2:576–585
118. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. An E reaction at a channel microband electrode. Electrochem Commun 2:567–575
119. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Application to a simple model problem. Electrochem Commun 2:150–156
120. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. First-order EC' mechanism at inlaid and recessed discs. Electrochem Commun 2:163–170
121. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Theory. Electrochem Commun 2:157–162
122. Carneiro-Neto EB, Sikora MS, Pereira EC, Lopes MC (2014) Probing the numerical convergence of a commercial finite element software in electrochemical simulations. Electrochemistry (Japan) 82:966–973
123. Chen X, Xiang J (2011) Solving diffusion equation using wavelet method. Appl Math Comput 217:6426–6432
124. Holm T, Sunde S, Seland F, Harrington DA (2015) A semianalytical method for simulating mass transport at channel electrodes. J Electroanal Chem 745:72–79
125. Krause KJ, Kätelhön E, Lemay SG, Compton RG, Wolfrum B (2014) Sensing with nanopores - the influence of asymmetric blocking on electrochemical redox cycling current. Analyst 139:5499–5503
126. Luo L, White HS (2013) Electrogeneration of single nanobubbles at sub-50-nm-radius platinum nanodisk electrodes. Langmuir 29:11169–11175
127. Michel R, Montella C, Verdier C, Diard JP (2010) Numerical computation of the Faradaic impedance of inlaid microdisk electrodes using a finite element method with anisotropic mesh adaptation. Electrochim Acta 55:6263–6273
128. Sanecki PT, Skitał PM, Kaczmarski K (2010) The mathematical models of the stripping voltammetry metal deposition/dissolution process. Electrochim Acta 55:1598–1604
129. Strutwolf J, Scanlon MD, Arrigan DWM (2009) Electrochemical ion transfer across liquid/liquid interfaces confined within solid-state micropore arrays - simulations and experiments. Analyst 134:148–158
130. Träuble M (2004) Modellierung und Simulation elektrochemischer Prozesse mit Randelementmethoden. Ph.D. thesis, Universität Oldenburg, Oldenburg, Germany. In German
131. Ueno K, Kim HB, Kitamura N (2003) Characteristic electrochemical responses of polymer microchannel-microelectrode chips. Anal Chem 75:2086–2091
132. Wu ZQ, Zhou T, Wang K, Zhang JR, Xia XH (2010) Current distribution at electrode surfaces as simulated by finite element method. Electrochim Acta 55:4870–4875

133. Zaino LP III, Contento NM, Branagan SP, Bohn PW (2014) Coupled electrokinetic transport and electron transfer at annular nanoband electrodes embedded in cylindrical nanopores. Chem Electrochem 1:1570–1576
134. Morf WE, Koudelka-Hep M, de Rooij NF (2006) Theoretical treatment and computer simulation of microelectrode arrays. J Electroanal Chem 590:47–56
135. Aoki S, Kishimoto K, Miyasaka M (1988) Analysis of potential and current density distributions using a boundary element method. Corrosion 44:926–932
136. Deconinck J, Magetto G, Vereecken J (1985) Calculation of current distribution and electrode shape change by the boundary element method. J Electrochem Soc 132:2960–2965
137. Qiu FL, Fisher AC, Henley IE, Dryfe RAW (2003) The boundary element method: the simulation of voltammetry at immiscible liquid/liquid interfaces. Electrochem Commun 5:169–174
138. Białecki R, Nahlik R, Łapkowski M (1984) Applying the boundary element method to electrochemical calculations of primary current distribution. Electrochim Acta 29:905–910
139. Burchardt M, Träuble M, Wittstock G (2009) Digital simulation of scanning electrochemical microscopy approach curves to enzyme films with Michaelis-Menten kinetics. Anal Chem 81:4857–4863
140. Cahan BD, Scherson D (1988) I-BIEM. An iterative boundary integral equation method for computer solutions of current distribution problems with complex boundaries - a new algorithm. I. Theoretical. J Electrochem Soc 135:285–293
141. Dukovic JO, Tobias CW (1990) Simulation of levelling in electrodeposition. J Electrochem Soc 137:3748–3755
142. Emmanuel B (2007) Computation of ac responses of arbitrary electrode geometries from the corresponding secondary current distributions: a method based on analytic continuation. J Electroanal Chem 605:89–97
143. Fan T, Mayle EJ, Kottke PA, Fedorov AG (2006) Simulation of electroanalysis using the boundary integral method. Trends Anal Chem 25:52–65
144. Froidevaux H, Mitha E, Salamin JY (1995) Modelization of a bipolar electrochemical reactor. Chimia 49:3–12
145. Fulian Q, Fisher AC, Denuault G (1999) Applications od the boundary element method in electrochemistry: scanning electrochemical microscopy. J Phys Chem B 103:4387–4392
146. Fulian Q, Fisher AC, Denuault G (1999) Applications od the boundary element method in electrochemistry: scanning electrochemical microscopy, Part 2. J Phys Chem B 103:4393–4398
147. Fulian Q, Williams NA, Fisher AC (1999) Computational electrochemistry: three-dimensional boundary element simulations of double electrode geometries. Electrochem Commun 1:124–127
148. Fulian Q, Ball JC, Marken F, Compton RG, Fisher AC (2000) Voltammetry of electroactive oil droplets. Part I: numerical modelling for three mechanistic models using the dual reciprocity finite element method. Electroanalysis 12:1012–1016
149. Gooch KA, Qiu FL, Fisher AC (2003) The digital simulation of voltammetry under stagnant and hydrodynamic conditions. In: Bard AJ, Stratmann M, Unwin PR (eds) Encyclopaedia of electrochemistry, volume 2, instrumentation and electroanalytical chemistry. Wiley-VCH, Weinheim, pp 122–142
150. Kelly JJ, Rahman KMA, Durning CJ, West A (1998) Effect of current distribution on quartz crystal microbalance measurements. J Electrochem Soc 145:492–497
151. Qiu FL, Fisher AC (2001) The boundary element method: applications to steady-state voltammetric simulations within domains extending to infinity. Electrochem Commun 3:117–121
152. Qiu FL, Fisher AC (2003) The boundary element method: chronoamperometric simulations at microelectrodes. Electrochem Commun 5:87–93
153. Sklyar O, Wittstock G (2002) Numerical simulations of complex nonsymmetrical 3D systems for scanning electrochemical microscopy using the boundary element method. J Phys Chem B 106:7499–7508

154. Sklyar O (2004) Modelling scanning electrochemical microscopy (SECM) experiments on microstructured functionalised surfaces. Ph.D. thesis, Universität Oldenburg, Oldenburg, Germany
155. Jin B, Qian W, Zhang Z, Shi H (1996) Application of the finite analytic numerical method. Part 1. Diffusion problems on coplanar and elevated interdigitated microarray band electrodes. J Electroanal Chem 411:29–36
156. Jin B, Qian W, Zhang Z, Shi H (1996) Application of the finite analytic numerical method. Part 3. Digital simulation of charge transfer to a micro-ring electrode interface. J Electroanal Chem 417:45–51
157. Jin B, Qian W, Zhang Z, Shi H (1996) Finite analytic numerical method - a new numerical simulation method for electrochemical problems. J Electroanal Chem 411:19–27
158. Qian W, Jin B, Diao G, Zhang Z, Shi H (1996) Application of a finite analytic numerical method. Part 2. Digital simulation of charge transfer to an oblate hemispheroid microelectrode and experiment verification. J Electranal Chem 414:1–10
159. Qian W, Jin B, Shi H, Zhang Z (1997) Finite analytic solution and finite analytic numerical method for solving two-dimensional diffusion problems on microelectrodes. J Electroanal Chem 439:29–36
160. Qian W, Jin BK, Shi HS, Yu JS, Zhang ZX (1997) Digital simulation of chronoamperometric current at microdisk electrode and its verification. Acta Chim Sinica 55:1108–1115
161. Aloui F, Rehimi F, Dumont E, Legrand J (2008) Inverse method applied for the determination of the wall shear rate in a scraped surface heat exchanger using the electrochemical technique. Int J Electrochem Sci 3:676–690
162. Barak-Shinar D, Rosenfeld M, Abboud S (2004) Numerical simulations of mass-transfer processes in 3D model of electrochemical sensor. J Electrochem Soc 151:H261–H266
163. Bortels L, Deconinck J, Bossche BVD (1996) The multi-dimensional upwinding method as a new simulation tool for the analysis of multi-ion electrolytes controlled by diffusion, convection and migration. Part 1. Steady state analysis of a parallel plane flow channel. J Electroanal Chem 404:15–26
164. Gray DG, Harrison JA (1970) Polymerisation of an electroactive species at the rotating disc electrode. J Electroanal Chem 24:187–194
165. Wright K (1964) Chebyshev collocation methods for ordinary differential equations. Comput J 6:358–365
166. Cabán R, Chapman TW (1976) Rapid computation of current distribution by orthogonal collocation. J Electrochem Soc 123:1036–1041
167. Whiting LF, Carr PW (1977) A simple, fast numerical method for the solution of a wide variety of electrochemical diffusion problems. J Electroanal Chem 81:1–20
168. Villadsen JV, Stewart WE (1967) Solution of boundary-value problems by orthogonal collocation. Chem Eng Sci 22:1483–1501
169. Villadsen J, Michelsen ML (1978) Solution of differential equation models by polynomial approximation. Prentice-Hall, Englewood Cliffs, NJ
170. Pons S (1984) Polynomial approximation techniques for differential equations in electrochemical problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 13. Marcel Dekker, New York, pp 115–190
171. Pons BS, Speiser B, McAleer JF (1982) Orthogonal collocation simulation of the rotating disc electrode. Electrochim Acta 27:1177–1179
172. Pons BS, Speiser B, McAleer JF, Schmidt PP (1982) Simulation of the dropping mercury electrode by orthogonal collocation. Electrochim Acta 27:1711–1714
173. Speiser B (1996) Numerical simulation of electroanalytical experiments: recent advances in methodology. In: Bard AJ, Rubinstein I (eds) Electroanalytical chemistry, vol 19. Marcel Dekker, New York, pp 1–108
174. Yen SC, Chapman TW (1982) Simulation of voltammetry by orthogonal collocation. J Electroanal Chem 135:305–312
175. Eddowes MJ (1983) Numerical methods for the solution of the rotating disc electrode system. J Electroanal Chem 159:1–22

176. Magno F, Bontempelli G, Perosa D (1983) A comparison of some recently developed procedures for digital simulation in electroanalytical research. Anal Chim Acta 147:65–76
177. Bieniasz LK, Britz D (1993) Efficiency of electrochemical kinetic simulations by orthogonal collocation and finite difference methods. A comparison. Acta Chem Scand 47:757–767
178. Speiser B (1993) Electrochemical simulations. Part 15. Advanced orthogonal collocation techniques in problem situations of the $EC_{cat}$ mechanism. A comment on the paper 'Efficiency of electrochemical kinetic simulations by orthogonal collocation and finite difference methods. A comparison' by L.K. Bieniasz and D. Britz. Acta Chem Scand 47:1238–1240
179. Hertl P, Speiser B (1987) Electroanalytical investigations. Part IV. The simulation of fast chemical equilibrium reactions in cyclic voltammetric reaction-diffusion models with spline collocation. J Electroanal Chem 217:225–238
180. Pons BS, Schmidt PP (1980) Global spline collocation in the simulation of electrochemical diffusion equations. Electrochim Acta 25:987–993
181. Pritzker MD (1988) Voltammetric response for the diffusion-controlled electrodeposition onto growing hemispherical nuclei. J Electroanal Chem 243:57–80
182. Urban P, Speiser B (1988) Electroanalytical investigations. Part VIII. The use of an expanding simulation space in the simulation of electrochemical reaction-diffusion models with orthogonal collocation. J Electroanal Chem 241:17–31
183. Caillaud JB, Padmanabhan L (1971) An improved semi-implicit Runge-Kutta method for stiff systems. Chem Eng J 2:227–232
184. Gear CW (1969) The automatic integration of stiff ordinary differential equations. In: Morrel AJH (ed) Information processing 68. North-Holland, Amsterdam, pp 187–193
185. Hindmarsh AC, Petzold LR (1995) Algorithms and software for ordinary differential equations and differential-algebraic equations. Part II: Higher-order methods and software packages. Comput Phys 9:148–155
186. Rice JR (1983) Numerical methods, software, and analysis. McGraw-Hill International, Auckland
187. Friedrichs MS, Friesner RA, Bard AJ (1989) A new approach to electrochemical simulations based on eigenvector-eigenvalue solutions of the diffusion equations. Part I. Potentiostatic boundary conditions. J Electroanal Chem 258:243–264
188. Kavanaugh TC, Friedrichs MS, Friesner RA, Bard AJ (1990) A new approach to electrochemical simulations based on eigenvalue-eigenvector solutions of the diffusion equation. Part II. Cyclic voltammetry and heterogeneous kinetics. J Electroanal Chem 283:1–14
189. Lawson JD (1967) Generalized Runge-Kutta processes for stable systems with large Lipschitz constants. SIAM J Numer Anal 4:372–380
190. Strutwolf J, Schoeller WW (1997) Digital simulation of potential step experiments using the extrapolation method. Electroanalysis 9:1403–1408
191. https://www.wolfram.com/mathematica/
192. Momotenko D, Periera CM, Girault HH (2012) Differential capacitance of liquid/liquid interfaces of finite thicknesses: a finite element study. Phys Chem Chem Phys 14:11268–11272
193. Bieniasz LK (1992) An efficient numerical method of solving the Abel integral equation for cyclic voltammetry. Comput Chem 16:311–317
194. Bieniasz LK (1993) An efficient numerical method of solving integral equations for cyclic voltammetry. J Electroanal Chem 347:15–30
195. Bieniasz LK (2008) An adaptive Huber method with local error control, for the numerical solution of the first kind Abel integral equations. Computing 83:25–39
196. Bieniasz LK (2008) Cyclic voltammetric current functions determined with a prescribed accuracy by the adaptive Huber method for Abel integral equations. Anal Chem 80:9659–0665
197. Bieniasz LK (2010) Automatic simulation of cyclic voltammograms by the adaptive Huber method for weakly singular second kind Volterra integral equations. Electrochim Acta 55:721–728

198. Bieniasz LK (2010) Automatic simulation of cyclic voltammograms by the adaptive Huber method for systems of weakly singular Volterra integral equations. J Electroanal Chem 642:127–134
199. Bieniasz LK (2010) An adaptive Huber method for weakly singular second kind Volterra integral equations with non-linear dependencies between unknowns and their integrals. Computing 87:35–54
200. Bieniasz LK (2011) An adaptive Huber method for non-linear systems of weakly singular second kind Volterra integral equations. Appl Math Comput 217:5622 – 5631
201. Bieniasz LK (2011) Analysis of the applicability of the integral equation method in the theory of transient electroanalytical experiments for homogeneous reaction-diffusion systems: the case of planar electrodes. J Electroanal Chem 657:91–97
202. Bieniasz LK (2011) Extension of the adaptive Huber method for solving integral equations occurring in electroanalysis, onto Kernel function representing fractional diffusion. Electroanalysis 23:1506–1511
203. Bieniasz LK (2011) A highly accurate, inexpensive procedure for computing integral transformation kernel and its moment integrals for cylindrical wire electrodes. J Electroanal Chem 661:280–286
204. Bieniasz LK (2011) Automatic simulation of electrochemical transients at cylindrical wire electrodes, by the adaptive Huber method for Volterra integral equations. J Electroanal Chem 662:371–378
205. Bieniasz LK (2011) Extension of the adaptive Huber method for Volterra integral equations arising in electroanalytical chemistry, to convolution kernels $\exp\left[-\alpha(t-\tau)\right]$ erex $\{[\beta(t-\tau)]^{1/2}\}$ and $\exp\left[-\alpha(t-\tau)\right]$ daw $\{[\beta(t-\tau)]^{1/2}\}$. J Comput Methods Sci Eng 11:323–338
206. Bieniasz LK (2012) Automatic simulation of electrochemical transients by the adaptive Huber method for Volterra integral equations involving kernel terms $\exp[-\alpha(t-\tau)]\mathrm{erex}\{[\beta(t-\tau)]\}$ and $\exp[-\alpha(t-\tau)]\mathrm{daw}\{[\beta(t-\tau)]\}$. J Math Chem 50:765–781
207. Bieniasz LK (2012) Automatic solution of integral equations pertinent to diffusion with first order homogeneous reactions at cylindrical wire electrodes. J Electroanal Chem 674:38–47
208. Bieniasz LK (2012) Automatic simulation of electrochemical transients assuming finite diffusion space at planar interfaces, by the adaptive Huber method for Volterra integral equations. J Electroanal Chem 684:20–31
209. Bieniasz LK (2013) Automatic solution of the Singh and Dutt integral equations for channel or tubular electrodes, by the adaptive Huber method. J Electroanal Chem 693:95–104
210. Bieniasz LK (2013) Automatic solution of integral equations describing electrochemical transients at dropping mercury electrodes. J Electroanal Chem 705:44–51
211. Bieniasz LK (2015) Modelling electroanalytical experiments by the integral equation approach. Springer, Heidelberg
212. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
213. Galus Z (1994) Fundamentals of electrochemical analysis, 2nd edn. Ellis Horwood, New York. (trans: Chalmers RA, Bryce WAJ (eds))
214. Randles JEB (1948) A cathode-ray polarograph. Part II - The current-voltage curves. Trans Faraday Soc 44:327–338
215. Ševčík A (1948) Oscillographic polarography with periodical triangular voltage. Collect Czechoslov Chem Commun 13:349–377
216. Emmons HW (1944) The numerical solution of partial differential equations. Q Appl Math 2:173–195
217. Bard AJ, Mirkin MV (2001) Scanning electrochemical microscopy. Marcel Dekker, New York
218. Nicholson RS, Shain I (1964) Theory of stationary electrode polarography. Single scan and cyclic methods applied to reversible, irreversible, and kinetic systems. Anal Chem 36:706–723
219. Nicholson RS (1965) Some examples of the numerical solution of nonlinear integral equations. Anal Chem 37:667–671

220. Nicholson RS, Olmstead ML (1972) Numerical solution of integral equations. In: Mattson J, Mark HB Jr, MacDonald HC Jr (eds) Computers in chemistry and instrumentation, vol 2. Marcel Dekker, New York, pp 119–139
221. Huber A (1939) Eine Näherungsmethode zur Auflösung Volterrascher Integralgleichungen. Monatsh Math Phys 47:240–246
222. Bieniasz LK (1993) ELSIM - a PC program for electrochemical kinetic simulations. Version 2.0 - solution of the sets of kinetic partial differential equations in one-dimensional geometry, using finite difference and orthogonal collocation methods. Comput Chem 17:355–368
223. Oldham KB (1981) An algorithm for semiintegration, semidifferentiation and other instances of differintegration. J Electroanal Chem 121:341–342
224. Horno J, González-Fernández CF, Hayas A, González-Caballero F (1989) Simulation of concentration polarization in electrokinetic processes by network thermodynamic methods. Biophys J 55:527–535
225. González CF, García-Hernández MT, Horno J (1992) Network simulation of a reversible electron-transfer under cyclic voltammetric conditions. Collect Czechoslov Chem Commun 57:1373–1380
226. Horno J, García-Hernández MT (1993) Digital simulation of electrochemical processes by the network approach. J Electroanal Chem 352:83–97
227. Horno J, García-Hernández, González-Fernández (1994) A network thermodynamic method for computer simulation of cyclic voltammetry. J Electroanal Chem 377:53–60
228. Deng Z, Lin X (1999) Simulation of ultra-fast cyclic voltammetric curve by the exponentially expand network method. Chin J Anal Chem 27:1376–1380. [In Chinese, Engl. abstract]
229. Deng ZX, Lin XQ, Tong ZH (2003) Exponentially expanded grid network approach EEGNA. An efficient way for the simulation of stiff electrochemical problems. Chin J Chem 21:1137–1145
230. Deng ZX, Tong ZH, Lin XQ (2004) Global evaluation of linear sweep voltammetric responses with electroactive species confined at the electrode surface. J Electroanal Chem 568:235–245
231. Deng ZX, Lin XQ, Tong ZH (2004) Universal electrochemical/chemical simulator based on an exponentially expanding grid network approach. Chin J Chem 22:719–726
232. Burney HS, White RE (1988) Predicting shunt currents in stacks of bipolar plate cells with conducting manifolds. J Electrochem Soc 135:1609–1612
233. Castilla J, García-Hernández MT, Hayas A, Horno J (1996) Simulation of non-stationary electrodiffusion processes in charged membranes by the network approach. J Membr Sci 116:107–116
234. Moya AA, Hayas A, Horno J (1995) Study of electrical migration in electrochemical cells by the network method. Ber Bunsenges Phys Chem 99:1037–1042
235. Moya AA, Horno J (1996) Simulation of nonstationary diffusion-migration processes in electrochemical cells using the network method. Electrochim Acta 41:285–290
236. Moya AA, Hayas A, Horno J (1996) A network approach to simulation of electrical properties of symmetric electrochemical cells. J Electroanal Chem 413:1–7
237. Moya AA, Hayas A, Horno J (1996) A network approach to the simulation of electrical properties of asymmetric electrochemical cells. J Electroanal Chem 413:9–14
238. Moya AA, Hayas A, Horno J (2000) Steady-state, transient and small-amplitude AC responses of an electrochemical cell with immobile background charge: a network approach. Solid State Ionics 130:9–17
239. Moya AA (2014) Electrochemical impedance of ion-exchange membranes in ternary solutions with two counterions. J Phys Chem C 118:2539–2553
240. Moya AA (2015) Theory of the formation of the electric double layer at the ion exchange membrane-solution interface. Phys Chem Chem Phys 17:5207–5218
241. Eyres NR, Hartree DR, Ingham J, Jackson R, Sarjant RJ, Wagstaff JB (1946) The calculation of variable heat flow in solids. Philos Trans R Soc Lond A 240:1–57
242. Nagel LW (1977) SPICE (simulation program with integrated circuit emphasis). Technical Report ERL-m382-1977, Electronics Research laboratory, University of California, Berkeley

243. Twinanga EW (1992) A guide to circuit simulation and analysis using PSPICE. Prentice-Hall, Englewood Cliffs, NJ
244. Horno J, García-Hernández MT, Castilla J, González-Fernández CF (1996) Network simulation of the first- and second-order catalytic mechanism for chronoamperometry. Electroanalysis 8:1145–1149
245. Horno J, González CF, Hayas A (1995) The network method for solutions of oscillating reaction-diffusion systems. J Comput Phys 118:310–319
246. García-Hernández MT, Castilla J, González-Fernández CF, Horno J (1997) Application of the network method to simulation of a square scheme with Butler-Volmer charge transfer. J Electroanal Chem 424:207–212
247. Lopéz-García JJ, Grosse C, Horno J (2002) Numerical study of the equilibrium properties of suspended particles surrounded by a permeable membrane with adsorbed charges. J Colloid Interf Sci 254:287–295
248. Coles BA, Compton RG, Brett CMA, Brett AMCFO (1995) Ohmic distortion of current-potential curves at wall-jet electrodes. J Electroanal Chem 381:99–104
249. Zhang WS, Zhang XW (1998) A numerical approach to the voltammograms of a thick plate Pd/H electrode. J Electroanal Chem 445:55–62
250. Jain MK (1984) Numerical solution of differential equations, 2nd edn. Wiley Eastern, New Delhi
251. Lapidus L, Seinfeld JH (1971) Numerical solution of ordinary differential equations. Academic Press, New York
252. Treanor CE (1966) A method for the numerical integration of coupled first-order differential equations with greatly different time constants. Math Comput 20:39–45
253. Fanelli N, Záliš S, Pospíšil C (1989) The growth of compact layers at the electrode interface. Part III. Monte Carlo simulations of the formation of fractal structures by diffusion-limited aggregation. J Electroanal Chem 262:35–44
254. Fanelli N, Záliš S, Pospíšil C (1990) Monte Carlo simulations of adsorption/desorption processes related to adsorptive stripping voltammetry. J Electroanal Chem 288:263–269
255. Voss RF, Tomkiewicz M (1985) Computer simulation of dendritic electrodeposition. J Electrochem Soc 132:371–375
256. Licht S, Cammarata V, Wrighton MS (1990) Direct measurements of the physical diffusion of redox active species: microelectrochemical experiments and their simulation. J Phys Chem 94:6133–6140
257. Borkowski M, Stojek Z (1992) Monte Carlo simulation of diffusional noise at microelectrodes. Electroanalysis 4:615–621
258. Baur JE, Motsegood PN (2004) Diffusional interactions at dual disk microelectrodes: comparison of experiment with three-dimensional random walk simulations. J Electroanal Chem 572:29–40
259. Juwono T, Hamad IA, Rikvold PA (2013) Effects of lateral diffusion on the dynamics of desorption. J Solid State Electrochem 17:379–384
260. Cutress IJ, Dickinson EJF, Compton RG (2011) Electrochemical random-walk theory. Probing voltammetry with small numbers of molecules: stochastic versus statistical (Fickian) diffusion. J Electroanal Chem 655:1–8
261. Byers JC, Nadappuram BP, Perry D, McKelvey K, Colburn AW, Unwin PR (2015) Single molecule electrochemical detection in aqueous solutions and ionic liquids. Anal Chem 87:10450–10456
262. Sliusarenko O, Oleinick A, Svir I, Amatore C (2015) Development and validation of an analytical model for predicting chronoamperometric responses of random arrays of micro- and nanodisk electrodes. Chem Electrochem 2:1279–1291
263. Wang LL, Tanb TL, Johnson DD (2015) Nanoalloy electrocatalysis: simulating cyclic voltammetry from configurational thermodynamics with adsorbates. Phys Chem Chem Phys 17:28103–28111

264. Nagy G, Sugimoto Y, Denuault G (1997) Three-dimensional random walk simulation of diffusion controlled electrode processes: (I) a hemisphere, disc and growing hemisphere. J Electroanal Chem 433:167–173
265. Nagy G, Denuault G (1997) Three-dimensional random walk simulation of diffusion controlled electrode processes: (II) arrays of growing hemispheres. J Electroanal Chem 433:175–180
266. Yang H, Peters JL, Allen C, Chern SS (2000) A theoretical description of microdialysis with mass transport coupled to chemical events. Anal Chem 72:2042–2049

# Chapter 10
# Adsorption

In this chapter it is shown how to simulate the adsorption of a substance, not taking into account any electrochemical reactions the substance may undergo. That is, only the adsorption itself is dealt with here. In Chap. 2, Sect. 2.5, some theory is presented, laying the groundwork for the simulation. It is noted there that adsorption may be controlled by transport and the adsorption isotherm, in which case there is equilibrium at all times between the solution and surface phases; or that the adsorption step itself may limit the rate of adsorption. In this latter case, there are rate constants whose values must be known. In both cases, for isotherms more complicated than the Henry isotherm (2.103), nonlinear terms will enter the equations to be solved in a simulation.

Simulation of adsorption kinetics is not given as much attention as electron transfer, but some work has been done over the years. Analytical solutions are few and far between, as mentioned in Chap. 2. so, as for electron transfer, simulation is needed. Some simulation work has been done. Rampazzo [1] was one of the first, using a numerical solution of the Volterra integral equation describing the adsorption kinetics. Flanagan et al. [2] mention nonlinear terms in a simulation, as do Miller and coworkers [3–6], and Lovrić et al. [7]. Britz et al. [8] considered nonlinear isotherms as part of the boundary conditions in an implicit simulation and Hsu et al. [9] modelled adsorption at an air–water interface. Bieniasz introduced the concept of an "interfacial species" in his work [10], and has incorporated adsorption kinetics in his program package ELSIM [11–13], specifically in [14], as did Ludwig et al. [15, 16] in their work on `Echem++`.

Other simulation works to be mentioned are (pure diffusion and isotherm control) [15, 17–33], or (finite adsorption rate) [9, 34, 35], or both [6, 36].

Rather than the integral equation approach of Rampazzo [1], the direct simulation from the transport equations is used here. In order to obtain a certain surface concentration $\Gamma$ or fractional coverage $\theta$, the substance in question must first arrive at the electrode, by some transport process. As was shown in Chap. 2, the normalised

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_10

equation describing the accumulation of substance at the electrode is

$$\frac{\mathrm{d}\theta}{\mathrm{d}T} = KG \tag{10.1}$$

with $K$ being the normalising collection $c^* \sqrt{D\tau}/\Gamma_m$ and $G$ the dimensionless flux. This equation must be supplemented by another, describing the relation between the coverage $\theta$ and the concentration $C_0$ in solution at $X = 0$. This is either an equation involving an adsorption isotherm or one involving adsorption rates. For both these cases, explicit or implicit methods can be used.

There is an extreme case—that of very strong adsorption where the adsorption parameter $b$ as in (2.101) is large, leading to the approximate condition $c_0 \approx 0$ (all $t$). This is just like the electrochemical purely diffusion limited potential step case, for which we have the solution $G(T)$, Eqs. (2.44) and (2.26). $G$ can now be inserted into (10.1), and simple integration then gives:

$$\theta(T) = \frac{2c^*\sqrt{D\tau}}{\Gamma_m\sqrt{\pi}}\sqrt{T}\,. \tag{10.2}$$

This was also solved for the dropping mercury electrode by Koryta in 1953 [37]. Other cases, either fast adsorption with consideration of isotherms, or rate-limiting adsorption, will now be described. In all cases below, a new variable $\theta$ must be added to the unknowns, vector $\mathbf{C}$. Conveniently, we make it the first element of all the unknowns.

## 10.1 Transport and Isotherm Limited Adsorption

For this case, we have, apart from the usual diffusion equations, two boundary condition equations relating $C_0$ and $\theta$. They are

$$\begin{aligned} BC_0 &= I(\theta) \\ \frac{\mathrm{d}\theta}{\mathrm{d}T} &= KG \end{aligned} \tag{10.3}$$

where, as mentioned above, $K$ comes from the collection of parameters that go into the normalisation. $I(\theta)$ is the adsorption isotherm. To this small set must now be added the $N$ discretised equations describing the diffusion of the substance in solution.

There are now several choices of method. The simplest may be the explicit method. Using this, one starts at time $T$, where we know all values, and use them to proceed to the new time $T + \delta T$. First, one recalculates all $C_i, i = 1 \ldots N$. Parallel with this, from the value of $G$, one calculates a new $\theta$. Discretising the second equation of the set (10.3) and expanding $G$ as usual as an $n$-point approximation leads to

$$\theta' = \theta + \delta TK \sum_{i=0}^{n-1} \beta_i C_i \,. \tag{10.4}$$

Then, the new value of $\theta$ is used in the isotherm equation to recalculate $C_0$. This is simple, but has the drawback of poor accuracy and the limit on the $\lambda$ factor. Clearly, an implicit method is preferable, such as BI with extrapolation. The diffusion part of the whole set of equations will depend on the placing of the points in space, as described in Chap. 8, for example using the general three-point equation (8.8) on page 147, or a multi-point form such as (8.31), page 151. These lead to the usual system as (8.11) or its multi-point relative (8.33) on pages 147 and 151. The first step is to do the backward Thomas scan as described in that chapter, and to apply the $u$-$v$ procedure, resulting in a set of linear expressions for the first $n$ (as yet unknown) concentration values, in terms of $C'_0$,

$$C'_i = u_i + v_i C'_0 \,. \tag{10.5}$$

This can be substituted into the $G$ approximation above (10.4) and discretising the whole equation (10.4) according to the BI method (only new values used), we have

$$\theta' = \theta + K\delta T \left( \sum_{i=0}^{n-1} \beta_i (u_i + v_i C'_0) \right) \tag{10.6}$$

which can be rearranged in terms of the two remaining unknowns $\theta'$ and $C'_0$, into

$$\theta' = P + QC'_0 \tag{10.7}$$

with $P$ and $Q$ obvious from (10.6). This must now be combined with the first boundary condition of (10.3). If the Henry isotherm holds, this is simply

$$BC'_0 = \theta' \tag{10.8}$$

and the solution follows easily. If, on the other hand, a nonlinear isotherm such as Langmuir or Frumkin isotherm holds, we have a nonlinear pair of equations, which can be solved using the Newton method. It will normally be aided by the fact that at a given step, both $\theta$ and $C_0$ change only a little, so the Newton process will probably converge rapidly. Details are left to the reader.

An obvious alternative choice of method, given the probably nonlinear form of the isotherm boundary condition is to use a Rosenbrock method. Then, the two boundary conditions are simply the first two equations in a whole DAE set, the first of the pair (10.3) being an algebraic equation, the second an *ode*. The Rosenbrock method is described in Chap. 9, Sect. 9.4 starting on page 200.

## 10.2 Adsorption Rate Limited Adsorption

If adsorption itself is a slow process, then rate equations for that process apply, as outlined in Chap. 2, from page 34. As with the isotherm-dependent boundary conditions, we may have nonlinear equations, such as (2.120). The boundary conditions, inserting (2.120) into (2.116), are

$$
\begin{aligned}
\frac{\mathrm{d}\theta}{\mathrm{d}T} &= V_f - V_b \\
\frac{\mathrm{d}\theta}{\mathrm{d}T} &= KG\,.
\end{aligned}
\tag{10.9}
$$

These can be made into a two-equation DAE set, by equating the two right-hand sides and using one of the equations, conveniently the simpler, second one. This yields the DAE system

$$
\begin{aligned}
0 &= V_f - V_b - K\sum_{i=0}^{n-1}\beta_i C'_i \\
\frac{\mathrm{d}\theta}{\mathrm{d}T} &= K\sum_{i=0}^{n-1}\beta_i C'_i\,.
\end{aligned}
\tag{10.10}
$$

As for the transport- and isotherm-controlled case above, these equation sets can now be handled either using a standard implicit method or, perhaps logically in the case of nonlinear isotherms, Rosenbrock.

## References

1. Rampazzo L (1969) Diffusion to a plane with adsorption according to Frumkin's isotherm. Electrochim Acta 14:733–739
2. Flanagan JB, Takahashi K, Anson FC (1977) Reactant adsorption in differential pulse polarography. Effects of adsorptive depletion of reactant, nonlinear adsorption isotherms and uncompensated resistance. J Electroanal Chem 81:261–273
3. Miller R, Lunkenheimer K (1978) Zur Adsorptionskinetik an fluiden Phasen-grenzen. Eine numerische Lösung für den diffusionskontrollierten Adsorptionsvorgang. Z Phys Chem (Leipzig) 259:863–868
4. Miller R, Lunkenheimer K, Kretzschmar G (1979) Ein Modell für die diffusions-kinetik-kontrollierte Adsorption von Tensidgemischen an fluiden Phasengrenzen. Colloid Polym Sci 257:1118–1120
5. Miller R, Kretzschmar G (1980) Numerische Lösung für ein gemischtes Modell der diffusions-kinetik-kontrollierten Adsorption. Colloid Polym Sci 258:85–87
6. Miller R (1981) On the solution of diffusion controlled adsorption kinetics for any adsorption isotherms. Colloid Polym Sci 259:375–381

7. Lovrić M, Kormorsky-Lovrić Ŝ (1981) O adsorpciji kontroliranoj difuzijom. Bull Soc Chim Beograd 46:93–98
8. Britz D, Heinze J, Mortensen J, Störzbach M (1988) Implicit calculation of boundary values in digital simulation applied to several types of electrochemical experiment. J Electroanal Chem 240:27–43
9. Hsu CT, Shao MJ, Lin SY (2000) Adsorption of $C_{12}E_4$ at the air-water interface: adsorption onto a fresh interface. Langmuir 16:3187–3194
10. Bieniasz LK (1996) A method-oriented approach to the formulation of algorithms for electrochemical kinetic simulations. Part 2. Extension to kinetic problems characterized by the simultaneous presence of bulk and interfacial species. J Electroanal Chem 404:195–208
11. http://www.cyf-kr.edu.pl/~nbbienia/elsim3ad.html
12. Bieniasz LK (1992) ELSIM - a user-friendly PC program for electrochemical kinetic simulations. Version 1.0 - solution of integral equations for linear scan and cyclic voltammetry. Comput Chem 16:11–14
13. Bieniasz LK (1993) ELSIM - a PC program for electrochemical kinetic simulations. Version 2.0 - solution of the sets of kinetic partial differential equations in one-dimensional geometry, using finite difference and orthogonal collocation methods. Comput Chem 17:355–368
14. Bieniasz LK (1997) ELSIM - a problem-solving environment for electrochemical kinetic simulations. Version 3.0 - solution of governing equations associated with interfacial species, independent of spatial coordinates or in one-dimensional space geometry. Comput Chem 21:1–12
15. Ludwig K, Speiser B (2007) EChem++ - an object-oriented problem solving environment for electrochemistry. Part 5. A differential-algebraic approach to the error control of adaptive algorithms. J Electroanal Chem 608:91–101
16. Ludwig K, Morales I, Speiser B (2007) EChem++ - an object-oriented problem solving environment for electrochemistry. Part 6. Adaptive finite element simulations of controlled-current electrochemical experiments. J Electroanal Chem 608:102–110
17. Calvente JJ, Kováčová Z (1996) Numerical simulation of desorption transients at electrodes on the basis of non-linear adsorption isotherms. J Chem Soc Faraday Trans 92:3701–3708
18. Calvente JJ, Andreu R (2011) Accurate analytical expressions for stripping voltammetry in the Henry adsorption limit. Anal Chem 83:6401–6409
19. Chen L, Lv C, Chen J, Bi S (2013) Numerical simulation study on cyclic reciprocal derivative chronopotentiometry of reversible electrode reaction coupled with Langmuir adsorption. Electrochim Acta 93:222–229
20. Chevallier FG, Klymenko OV, Jiang L, Jones TGJ, Compton RG (2004) Mathematical modelling and numerical simulation of adsorption processes at microdisk electrodes. J Electroanal Chem 574:217–237
21. Ellis JS, Strutwolf J, Arrigan DWM (2012) Finite-element simulations of the influence of pore wall adsorption on cyclic voltammetry of ion transfer across a liquid-liquid interface formed at a micropore. Phys Chem Chem Phys 14:2494–2500
22. Engelman EE, Evans DH (1992) Explicit finite-difference digital simulation of the effects of rate-controlled product adsorption or deposition in double-potential-step chronocoulometry. J Electroanal Chem 331:739–749
23. Fanelli N, Záliš S, Pospíšil C (1990) Monte Carlo simulations of adsorption/desorption processes related to adsorptive stripping voltammetry. J Electroanal Chem 288:263–269
24. Fekner Z (2008) Digital simulation of cyclic chronopotentiometry and reciprocal derivative chronopotentiometry for linear adsorption systems. Collect Czechoslov Chem Commun 73:201–228
25. Feldberg SW (1972) Digital simulation of electrochemical surface boundary phenomena. Multiple electron transfer and adsorption. In: Mattson J, Mark HB Jr, MacDonald HC Jr (eds) Computers in Chemistry and Instrumentation, vol 2. Marcel Dekker, New York, pp 185–215
26. Hepel T (1985) Linear potential scan voltammetry for irreversible co- adsorption of electroactive species. J Electroanal Chem 193:89–101

27. Kirowa-Eisner E, Gepshtein R, Gileadi E (2005) Effect of diffusion in underpotential deposition: simulated and experimental results. J Electroanal Chem 583:273–285
28. Kobayashi K (1988) Digital simulation of normal pulse polarographic adsorption waves of methyl viologen. Chem Lett 1243–1246
29. Kobayashi K, Minami N, Yamauchi S (1997) Digital simulation of the adsorption behavior of methyl viologen on mercury electrode. Proc Electrochem Soc 97-19:287–297
30. Leverenz A, Speiser B (1991) Electroanalytical simulations. Part 13. The simulation of adsorption processes at an electrode by orthogonal collocation algorithms. J Electroanal Chem 318:69–89
31. Łobacz M, Orlik M, Stroka J, Galus Z (2002) Method of separation and determination of the characteristics of the adsorbed and nonadsorbed states of electroactive substances on electrodes. Langmuir 18:2765–2770
32. Martinet S, Bouteillon J, Caire JP (1998) Modelling of cyclic voltammograms for two-step metal deposition on an inert electrode with adsorption. J Appl Electrochem 28:819–825
33. Schulz C, Speiser B (1993) Electroanalytical simulations. Part 14. Simulation of Frumkin-type adsorption processes by orthogonal collocation under cyclic voltammetric conditions. J Electroanal Chem 354:255–271
34. Juwono T, Hamad IA, Rikvold PA (2013) Effects of lateral diffusion on the dynamics of desorption. J Solid State Electrochem 17:379–384
35. Szulborska A, Baranski A (1994) Numerical simulation of kinetically controlled electrosorption processes under cyclic voltammetric conditions. J Electroanal Chem 377:23–31
36. Miller R (1980) Zur Adsorptionskinetik an der Oberfläche wachsender Tropfen. Colloid Polym Sci 258:179–185
37. Koryta J (1953) Über den Einfluss der Farbstoffe der Eosin-Gruppe auf die reversible Oxydo-Reduktion an der tropfenden Quecksilberelektrode. Collect Czechoslov Chem Commun 18:206–213

# Chapter 11
# Effects Due to Uncompensated Resistance and Capacitance

Electrochemists are aware of the annoying residual uncompensated solution resistance $R_u$ between the Luggin probe and the working electrode, see, for example, [1]. Although it is possible in principle to compensate fully for the $iR$ error thus introduced [2, 3], this is rarely done, as it introduces, in practice, undesirable instrumental oscillations or, in the case of damped feedback [3], sluggish potentiostat response.

The other often annoying fact electrochemists must live with is the double layer capacitance $C_{dl}$. This produces capacitive currents whenever the applied potential changes (see again [1]). The two effects work together, as capacitive currents also give rise to further $iR$ errors.

With potential step methods, the capacitive current is a transient, decaying with a time constant equal to $R_uC_{dl}$. The usual procedure is to wait several of these time constants before making the current measurement, by which time the capacitive current has declined to a negligible value. It is therefore not a serious problem with potential step experiments.

Where both capacitive current and $iR$ do interfere is with a.c. voltammetry (not gone into here) and LSV experiments. An early classic study is that of Nicholson [4], who investigated the effects of $iR$ alone, pointing out that a simple correction, from measured currents and known $R_u$, for the potentials, does not work. The LSV curve becomes distorted and such a correction does not retrieve the shape of the curve as it would be in the absence of an $iR$ effect. The reason is that the varying current during the sweep changes the electrode potential by a varying amount $iR_u$, and thus the potential program, that was intended to be linear with time, is no longer so.

Bowyer et al. [5] and Strutwolf [6] show examples of such distorted potential-time relations and also distorted LSV curves, see also below.

The simulation literature deals with this problem sporadically, although it is often simply ignored. The $iR$ effect introduces nonlinear boundary conditions (see below), and these have been dealt with in various ways. Gosser [7] advocates simple subtraction, using known measured currents of the experiment one is simulating in order to fit some parameter. Deng et al. [8] use a stepwise procedure that

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_11

successively solves for each of the several unknowns without iteration. Iteration using binary searches have been used [6, 9, 10], as well as a Gauss–Seidel method [11]. Safford et al. [12] rejected binary searching as too slow and Newton–Raphson iteration as unreliable, and used the van Wijngaarden–Dekker–Brent root-finding method, as described in Press et al. [13]. This is as reliable as a binary search (bisection) but faster, using a parabolic fit at each step. The best method is probably Newton–Raphson iteration, as used by Rudolph [14], despite the misgivings of some.

Simulations must thus handle the nonlinear boundary conditions. Some have taken the easy way out and used explicit methods [15–18], others used hopscotch [12, 19], ADI (for a two-dimensional problem) [20, 21] and other methods [4, 5, 22–26]. Bieniasz [27] used the Rosenbrock method (see Chap. 9), which makes sense because it effectively deals with nonlinearities without iterations at a given time step. Some have simulated both resistance and capacitive effects [12, 15, 16, 20–22, 25].

The work of Bowyer et al. [5] is interesting. Recognising that *iR* effects will distort the LSV signal from the nominal linear change, they took account of the measured current and knowing the resistance, calculated the resulting distorted potential sweep signal, and fed this into the simulation. They were thus able to produce a simulated SV (not LSV) response to match with the experimental one.

The classic work in this connection is that by Imbeaux and Savéant [22], who took the integral equation approach (see Chap. 9), incorporating the *iR* effects. They also established the formulation of the problem and the way to normalise both the uncompensated resistance $R_u$ and double layer capacitance $C_{dl}$, which has been followed by most workers since then. Their normalisation of $R_u$ followed that of Nicholson [4].

In what follows only the LSV problem will be considered, since it is here that the major problems lie. The capacitive current component is, at any given time, given by

$$i_c = -C_{dl}\frac{\mathrm{d}E}{\mathrm{d}t} \tag{11.1}$$

where the negative sign is intended to produce a (positive) cathodic current from a cathodic-going sweep. This current will give rise to an *iR* error in the applied potential, equal to $+i_cR_u$ (that is, the applied potential will be a little more positive than intended). The Faradaic current will contribute a similar *iR* error.

First we must normalise some quantities, to make them compatible with the other dimensionless parameters already used. We refer to the normalisation formulae on p. 28. Recall that we have normalised voltage by the factor $\frac{n\mathcal{F}}{\mathcal{R}\mathcal{T}}$ and that the time unit $\tau$ for LSV is equal to $\frac{\mathcal{R}\mathcal{T}}{n\mathcal{F}v}$ ($v$ being the sweep rate), or the time the sweep takes to traverse one normalised potential unit $p$.

Resistance has units of Volts per Ampere, and thus must be converted to $p$ units per $G$ units. Using the normalisations in Chap. 2, this comes to

$$\rho = R_u \frac{n\mathcal{F}}{\mathcal{RT}} n\mathcal{F} D^{\frac{1}{2}} c^* \sqrt{\frac{n\mathcal{F}v}{\mathcal{RT}}} \,. \tag{11.2}$$

This is as presented in [22], and is not normally simplified further. For capacity, which has units of current × time per Volt, these become $GT$ units per $p$ units here and conversion leads to

$$\gamma_c = C_{dl} \frac{1}{n\mathcal{F} D^{\frac{1}{2}} c^*} \sqrt{\frac{\mathcal{RT}v}{n\mathcal{F}}} \,, \tag{11.3}$$

also normally written in this unsimplified form.

## 11.1 Boundary Conditions

It is solely in the boundary conditions that simulations differ from those without *iR* effects. We find that for a general electrochemical reaction (ignoring homogeneous reactions in this context), involving the two species A and B, and a set nominal potential $p_{nom}$, we have six boundary quantities and thus six equations for them. In fact, it is quite easy to reduce them to a set of four by elimination of two of the currents, but it seems clearer not to do so and to formulate all six equations.

We have the following unknown boundary values: the two species' near-surface concentrations $C_{A,0}$ and $C_{B,0}$, the two species' fluxes, respectively $G_A$ and $G_B$, the additional capacitive flux $G_c$, and the potential $p$, differing (for $\rho > 0$) from the nominal, desired potential $p_{nom}$ that was set in, for example, an LSV sweep or a potential step experiment. Five of the six required equations are common to all types of experiments, but the sixth (here, the first one given below) depends on the reaction. That might be a reversible reaction, in which case a form of the Nernst equation must be invoked, or a quasi-reversible reaction, in which case the Butler–Volmer equation is used (see Chap. 6 for these). Let us now assume an LSV sweep, the case of most interest in this context. The unknowns are all written as future values with apostrophes, because they must, in what follows below, be distinguished from their present counterparts, all known.

The unknown capacitive flux $G'_c$ is derived as follows. Equation (11.1) becomes, in dimensionless terms,

$$G'_c = -\gamma_c \frac{\mathrm{d}p}{\mathrm{d}T} \,. \tag{11.4}$$

Imbeaux and Savéant [22] provide the equation for the changed potential, which translates in present terms into the equation

$$p = p_1 - T + \rho(G_c + G_A) \tag{11.5}$$

(sweeping in the negative direction) and this gives, after differentiation,

$$\frac{\mathrm{d}p}{\mathrm{d}T} = -1 + \rho\left(\frac{\mathrm{d}G_A}{\mathrm{d}T} + \frac{\mathrm{d}G_c}{\mathrm{d}T}\right) , \tag{11.6}$$

that is, both the Faradaic and capacitive currents affect the potential if there is an *iR* drop. We can now construct all the needed boundary equations.

In the reversible case we have

$$C'_{A,0} - \exp(p')\, C'_{B,0} = 0 \tag{11.7}$$

as the first equation, with three unknowns (including $p'$). For a quasireversible system, the Butler–Volmer equation applies, instead,

$$G'_A = K_f\, C'_{A,0} - K_b\, C'_{B,0} \tag{11.8}$$

as described on page 109. We then have the flux equality equation

$$G'_A + G'_B = 0 , \tag{11.9}$$

and the numerical approximation to the two fluxes

$$\begin{aligned} G'_A - \sum_{i=0}^{n-1} \beta_i C'_{A,i} &= 0 \\ G'_B - \sum_{i=0}^{n-1} \beta_i C'_{B,i} &= 0 . \end{aligned} \tag{11.10}$$

The capacitive flux equation (11.4) is combined with (11.6) and the time derivatives are approximated by the two-point formula

$$\frac{\mathrm{d}G}{\mathrm{d}T} \approx \frac{G' - G}{\delta T} . \tag{11.11}$$

From the vantage point of time $T + \delta T$, these are backward differences. This gives

$$G'_c + \frac{\gamma_c \rho}{\delta T}\left(G'_c + G'_A\right) = \gamma_c + \frac{\gamma_c \rho}{\delta T}\left(G_c + G_A\right) . \tag{11.12}$$

Lastly, (11.5) is put into the form

$$p' - \rho(G'_c + G'_A) = p_1 - T' = p'_{nom} \; . \tag{11.13}$$

It will be noted that the equation pair (11.10) contains further unknowns $C'_{A,i}$ and $C'_{B,i}$, for $i > 0$. These can however be eliminated as described in Chap. 6, using the **u-v** mechanism. We assume that some implicit method is used here and that the first, backward, Thomas scan has been performed. Then, as described in that chapter, Sect. 6.2 or, for coupled systems, Sect. 6.4, concentrations can be expressed in the form

$$C'_i = u_i + v_i \, C'_0 \tag{11.14}$$

for both species. Equation (11.10) then become

$$\begin{aligned} G'_A - C'_{A,0} \sum_{i=0}^{n-1} \beta_i v_{A,i} &= \sum_{i=0}^{n-1} \beta_i u_{A,i} \\ G'_B - C'_{B,0} \sum_{i=0}^{n-1} \beta_i v_{B,i} &= \sum_{i=0}^{n-1} \beta_i u_{B,i} \; , \end{aligned} \tag{11.15}$$

now only containing concentrations at $X = 0$ or $i = 0$ as unknowns. For convenience, we rewrite these as

$$\begin{aligned} G'_A - V_A \, C'_{A,0} &= U_A \\ G'_B - V_B \, C'_{B,0} &= U_B \end{aligned} \tag{11.16}$$

with the four constants obvious from (11.15).

As mentioned above, the two unknown fluxes $G'_A$ and $G'_B$ appearing in the set can be eliminated by the application of the approximation pair (11.16), but it might be clearer not to do this and leave the full set of six unknowns as they are.

Essentially everything has now been given. The six-equation set must be solved numerically, and the Newton method works very well, requiring normally only two to three iterations at most, since the changes over a given time interval are relatively small. For this purpose the unknowns are gathered into the unknowns vector $\mathbf{X} \equiv [C'_{A,0} \; C'_{B,0} \; G'_A \; G'_B \; G'_c \; p']^T$. Further treatment is now confined to a concrete example.

### 11.1.1 An Example

The case of a reversible reaction is assumed, requiring (11.7). The set of six equations given above are written as the system

$$\mathbf{F}(\mathbf{X}) = 0 \tag{11.17}$$

or, detailed,

$$\begin{array}{rcl}
C'_{A,0} - \exp(p')C'_{B,0} & = & 0 \\
V_A C'_{A,0} + G'_A - U_A & = & 0 \\
V_B C'_{B,0} + G'_B - U_B & = & 0 \\
G'_A + G'_B & = & 0 \\
\rho \dfrac{G'_A}{\delta T} + \left(1 + \dfrac{\rho}{\delta T}\right) G'_c - \gamma_c - \rho \dfrac{(G_A + G_c)}{\delta T} & = & 0 \\
-\rho G'_A - \rho G'_c + p' - p'_{nom} & = & 0\,.
\end{array} \tag{11.18}$$

The Newton–Raphson method will now be described very briefly. For a more detailed description, see, for example, Press et al. [13]. We assume that the present vector **X** is in error by a small amount $\delta\mathbf{X}$, and a short Taylor expansion leads to

$$F(\mathbf{X} + \delta\mathbf{X}) = F(\mathbf{X}) + \mathbf{J}(\delta\mathbf{X})\,. \tag{11.19}$$

**J** is the Jacobian of the system (11.18), that is, the derivatives matrix, with respect to all the variables, of the system. It is

$$\mathbf{J} \equiv \begin{bmatrix}
1 & -\exp(p) & 0 & 0 & 0 & -\exp(p)C_{B,0} \\
V_A & 0 & 1 & 0 & 0 & 0 \\
0 & V_B & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & \frac{\gamma_c \rho}{\delta T} & 0 & 1 + \frac{\gamma_c \rho}{\delta T} & 0 \\
0 & 0 & -\rho & 0 & -\rho & 1
\end{bmatrix}\,. \tag{11.20}$$

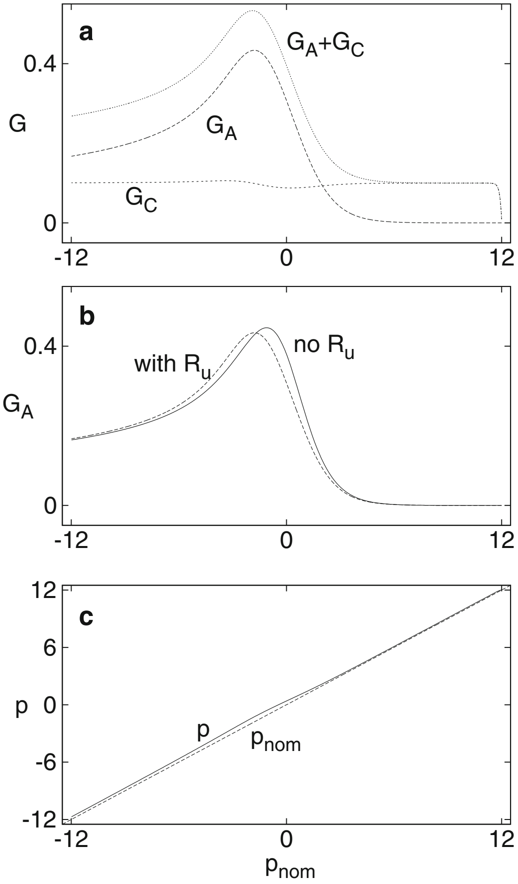
The variables have been written without apostrophes. At the beginning of the Newton process, they have the old values, while the function $\mathbf{F}(\mathbf{X})$ supplies new constant values, driving the calculation. We demand that the corrected vector $\mathbf{X}+\delta\mathbf{X}$ satisfies (11.17), that is, that the left-hand side of (11.19) is zero. This leaves

$$\mathbf{J}\,\delta\mathbf{X} = -\mathbf{F}(\mathbf{X})\,, \tag{11.21}$$

a linear system that can be solved easily, for example using LUD decomposition.

Example program LSV4IRC (Appendix E) does this calculation. It is of interest to look at some results. The program was run with values $\rho = 1$ and $\gamma_c = 0.1$. Figure 11.1a shows all three fluxes. The dot-dashed (top) line is the total flux. Note that the LSV sweep goes from right to left. Note also the initial rise of the total current to the capacitive value, delayed by the time constant $\rho\gamma_c$. The solid line represents $G_A$ alone, while the dashed (lowest at the left) line represents the capacitive current alone. In the absence of uncompensated resistance, this would rise to the constant value of 0.1 as soon as the sweep starts, because of the constant

**Fig. 11.1** LSV faradaic currents and potentials with and without uncompensated resistance. Curve (**a**) The individual currents and their sum with *iR* and capacitance present; (**b**) Faradaic current $G_A$ with and without *iR* error; (**c**) nominal (linear) and actual potential against applied potential, with and without *iR* effect



change in potential, and remain at that value. However, because of the *iR* effect, its rise is delayed, and its value does not remain constant as the faradaic current begins to rise, since then there are changes in potential beyond those due to the sweep, because of contributions by $\rho G_A$ to the potential. It is also of interest to see how much the faradaic current itself differs from that in the absence of *iR* effects. This is shown in Fig. 11.1b. If one were to simulate only for $G_A$, one might conclude from that figure that the effect of *iR* is slight, but clearly, considering the total current shown in Fig. 11.1a, it is not. It is this kind of curve one would obtain from an experiment, and would compare with a simulation. These marked changes arise from quite (visually) small deviations of the potential from the linear sweep, as seen in Fig. 11.1c.

Thus we can conclude that *iR* effects ought to be included in LSV simulations.

# References

1. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
2. Britz D (1978) IR elimination in electrochemical cells. J Electroanal Chem 88:309–352
3. Britz D (1980) 100% IR compensation by damped positive feedback. Electrochim Acta 25:1449–1452
4. Nicholson RS (1965) Some examples of the numerical solution of nonlinear integral equations. Anal Chem 37:667–671
5. Bowyer WJ, Engelman EE, Evans DH (1989) Kinetic studies by cyclic voltammetry at low temperatures using microelectrodes. J Electroanal Chem 262:67–82
6. Strutwolf J (1995) Digitale Simulation elektrochemischer Systeme: Untersuchungen zeitabhängiger Phänomene an rotierenden Scheibenelektroden und Analyse von Cyclovoltammogrammen durch direkte Simulation. Ph.D. thesis, Universität Bielefeld, Bielefeld
7. Gosser DK Jr (1993) Cyclic voltammetry. VCH, New York, Weinheim
8. Deng ZX, Lin XQ (1999) Digital simulation of fast cyclic voltammogram by integration of double layer charging programs. J Electroanal Chem 464:215–221
9. Chen T, Dong S, Xie Y (1994) Influence of the ohmic polarization effect on thin-layer spectroelectrochemistry. J Electroanal Chem 379:239–245
10. Goldberg IB, Bard AJ (1972) Resistive effects in thin electrochemical cells: digital simulations of current and potential steps in thin layer electrochemical cells. J Electroanal Chem 38:313–322
11. Wipf DO, Wightman RM (1988) Submicrosecond measurements with cyclic voltammetry. Anal Chem 60:2460–2464
12. Safford LK, Weaver MJ (1991) Cyclic voltammetric wave-shapes for microdisk-electrodes: coupled effects of solution resistance, double-layer capacitance, and finite electrochemical kinetics. J Electroanal Chem 312:69–96
13. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical recipes in Fortran. The art of scientific computing, 2nd edn. Cambridge University Press, Cambridge
14. Rudolph M (1995) Digital simulations with the fast implicit finite difference algorithm: the development of a general simulator for electrochemical processes. In: Rubinstein I (ed) Physical electrochemistry. Marcel Dekker, New York, pp 81–129
15. Bond AM, Mahon PJ, Oldham KB, Zoski CG (1994) Investigation of the influence of residual uncompensated resistance and incomplete charging current correction on the calculation of electrode kinetics when global and convolution analysis methods are used. J Electroanal Chem 366:15–27
16. Bond AM, Feldberg SW (1998) Analysis of simulated reversible cyclic voltammetric responses for a charged redox species in the absence of added electrolyte. J Phys Chem B 102:9966–9974
17. Flanagan JB, Takahashi K, Anson FC (1977) Reactant adsorption in differential pulse polarography. Effects of adsorptive depletion of reactant, nonlinear adsorption isotherms and uncompensated resistance. J Electroanal Chem 81:261–273
18. Orlik M (1997) Digital simulation of cyclic voltammetry in a two-electrode system and its application to the kinetics of bis(biphenyl)chromium(I) in N,N-dimethyl formamide. J Electroanal Chem 434:139–152
19. Safford LK, Weaver MJ (1989) The combined influences of solution resistance and charge-transfer kinetics on microelectrode cyclic voltammetry. J Electroanal Chem 261:241–247
20. Amatore C, Oleinick A, Svir I (2008) Capacitive and solution resistance effects on voltammetric responses of a thin redox layer attached to disk microelectrodes. Anal Chem 80:7957–7963
21. Amatore C, Oleinick A, Klymenko OV, Svir I (2009) Capacitive and solution resistance effects on voltammetric responses at a disk microelectrode covered with a self-assembled monolayer in the presence of electron hopping. Anal Chem 81:8545–8556
22. Imbeaux JC, Savéant JM (1970) Linear sweep voltammetry. Effect of uncompensated cell resistance and double layer charging on polarization curves. J Electroanal Chem 28:325–338

23. Myland JC, Oldham KB (2002) Convolutive modelling in the absence of supporting electrolyte: coping with migration and changing resistance in predicting voltammetry. J Electroanal Chem 529:66–74
24. Santhanam S, Ramani V, Srinivasan R (2012) Numerical investigations of solution resistance effects on nonlinear electrochemical impedance spectra. J Solid State Electrochem 16:1019–1032
25. Sher AA, Bond AM, Gavaghan DJ, Harriman K, Feldberg SW (2004) Resistance, capacitance, and electrode kinetic effects in Fourier-transformed large-amplitude sinusoidal voltammetry: emergence of powerful and intuitively obvious tools for recognition of patterns of behavior. Anal Chem 76:6214–6228
26. Weidner JW, Fedkiw PS (1991) Effect of Ohmic, mass-transfer, and kinetic resistances on linear-sweep voltammetry in a cylindrical-pore electrode. J Electrochem Soc 138:2514–2526
27. Bieniasz LK (2002) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 12. Patch-adaptive simulation of example transient experiments described by kinetic models defined over multiple space intervals in one-dimensional space geometry. J Electroanal Chem 527:21–32. Corrigendum: ibid. 565:141 (2004)

# Chapter 12
# Two (and Three) Dimensions

Electrochemical cells are of course three-dimensional (and in fact there is a section in this chapter on three-dimensional geometry, Sect. 12.4.1). In preceding chapters, symmetry or the absence of concentration gradients in two of these dimensions has been assumed, thus conveniently reducing the system to one dimension. This is not always possible, and in fact in recent decades, some of the most popular electrodes require at least two dimensions for reasonable simulations. These are first and foremost the ultramicroelectrodes (UMEs), in their various forms of disk and band electrodes, among others. Some modern electrochemical techniques such as scanning electrochemical microscopy employ UMEs. UMEs have also been assembled into arrays of such, increasing the simulation difficulties. Fortunately, the vast majority of these UMEs have zero gradients in one of the three directions, or two of the three directions share the same geometry and thus require "only" two dimensions for their representation. Going from one to two dimensions, however, is a major step, requiring programming sophistication in order to avoid using too much computing time for a given simulation. In what follows here, the ultramicrodisk electrode (UMDE) will serve as the model for how to proceed, although the others are mentioned, and references and some theory are provided.

A word on the names given to these electrodes is in order. As Amatore notes [1], there is some confusion about what is "ultramicro-" and what is "micro-". He suggests that "ultramicro-" should be applied to electrodes of such small size that they eliminate effects of natural convection, whereas "microelectrodes" are affected by convection. This would still depend on the time scale of the experiment. Bard and Faulkner [2, p. 169] decided that it is the size that rules, and regard electrodes with at least one dimension smaller than 25 μm as ultramicro-sized, and in this chapter this definition will be used. It will be seen that arrays of these UMEs are larger than this, so they are microelectrodes when functioning as a single unit, as they do at longer times. This is in line with the definition of "microelectrodes" by Laitinen and Kolthoff [3] who experimented with millimetre-sized electrodes, calling them microelectrodes.

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8_12

Until the 1960s, the dropping mercury electrode (DME) dominated electroanalytical chemistry (see such standard texts as [2, 4, 5] for details, and complications). It could be idealised as a sphere (disregarding the shielding due to the capillary and the slight deviation from sphericity) and had the advantages of a clean and smooth electrode surface, and a very wide potential working range, due to the high hydrogen overvoltage for water reduction at mercury. However, it necessitated the handling of liquid mercury, and it has mostly been replaced by the very small solid electrodes used today. One of the first to appear was the rotating disk electrode (RDE) [6] which is mentioned in Chap. 14, as it involves convection. It then led to stationary disk electrodes, and other stationary types.

Today, a number of UMEs are in use. They include the disk electrode, flat or hemispherical, the flat types being either inlaid—that is, flush with and embedded in an insulating plane—or recessed or protruding; band electrodes, either flat or hemicylindrical; and arrays of all these. The flat electrodes, flush with the insulating plane, have a problem of very large local current densities at the electrode edges. Bond et al. [7] mention this problem, and cite Engstrom et al. [8] for what they call an illuminating demonstration of the effect, by experiment. The interesting paper by Zeiri et al. [9] also gives evidence for high edge current densities. The edge effect is responsible for the problems in simulating these electrodes. No matter how closely one packs grid lines near the edges, appreciable errors are introduced and conformal mapping techniques are indicated.

The books by Fleischmann et al. [10] and (Eds.) Montenegro et al. [11] are a useful sources of information on all the used UMEs, both for experimental and theoretical work. Some reviews stand out, such as those of Aoki [12], Amatore [13] and Speiser [14], as well as the rather thorough section on UMEs in Bard and Faulkner [2] and the detailed review by Heinze [15] and the more recent [16], as well as others [17–22], covering a period up to 2014.

In this chapter, although the various ultramicroelectrode geometries are described and literature is provided both to theory and simulation work, the emphasis is on the UMDE. The principles of simulation at this electrode are then applicable to the others.

## 12.1 Theories

### *12.1.1 The Ultramicrodisk Electrode*

Initially, efforts were made to find expressions for the deviation of currents at the UMDE from that at a so-called planar electrode, which is the unidimensional case (called a “shrouded plane” by Oldham [23]). One can regard this as a disk (or any shape) at the bottom of an insulating deep well of the same cross-sectional shape as the electrode, so that the system can be reduced to one dimension. Here the Cottrell equation defines the current for a potential step, as in Chap. 2 (2.37) and (2.44).

At a flush UMDE, the current deviates from the Cottrell value very soon after the potential jump. Lingane [24] suggested that to a good approximation and for a range of small values of time $t$, the current $i_{UMDE}$ at a UMDE could be expressed as

$$\frac{i_{UMDE}}{i_{Cott}} = 1 + A\left(\frac{Dt}{a^2}\right)^{\frac{1}{2}} \tag{12.1}$$

where $a$ is the radius of the UMDE. He measured this experimentally, and found little deviation from the straight line for a range of $t$, and determined the slope $A$ to be 2.12. In the same year, Soos and Lingane studied the problem mathematically [25] and found that the slope was 2.26 or $4/\sqrt{\pi}$; they considered this value a slight overestimate but in good agreement with experiment. This factor was then the subject of further study following these two papers. It was measured again and found to be 1.79 [26], then determined by simulation by the same team [27], again 1.79. Heinze [28] measured it also in his classic simulation study of the UMDE, and found it to lie in the range 1.77 . . . 2.26. Flanagan and Marcoux [29] found, by simulation, a value of 1.92. Shoup and Szabo [30] corrected the value of Kakihana et al. [27], arriving at the correct one, $\sqrt{\pi}$ or 1.77. This is given again indirectly in the summary paper by Aoki [12], where other electrode geometries are also considered.

These attempts to express deviations of the current at a UMDE from the Cottrell current are somewhat fruitless because the expressions do not hold for other than rather small $t$ values or rather, dimensionless values of the normalised time, $Dt/a^2$. General solutions were—and are—needed. There have been no analytical solutions holding for all times, but some limiting expressions, and a rather accurate approximate one, have been derived.

Consider Fig. 12.1, depicting the UMDE in a cylindrical coordinate system. The electrode of radius $a$ is flush with an infinite insulating plane. The *pde* that governs diffusion around the UMDE is then

$$\frac{\partial c}{\partial t} = D\left(\frac{\partial^2 c}{\partial r^2} + \frac{1}{r}\frac{\partial c}{\partial r} + \frac{\partial^2 c}{\partial z^2}\right) \tag{12.2}$$

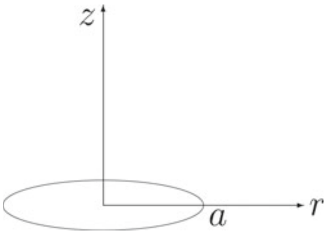


**Fig. 12.1** Ultramicrodisk electrode coordinate system

with boundary conditions, for the "Cottrell" case (potential jump to a potential very far from equilibrium), as follows:

$$\begin{aligned}
t = 0,\ \text{all}\ r, z: &\quad c = c^* \\
t > 0,\ r \le a,\ z = 0: &\quad c = 0 \\
r = 0,\ z > 0: &\quad \frac{\partial c}{\partial r} = 0 \\
r > a,\ z = 0: &\quad \frac{\partial c}{\partial z} = 0 \\
r \to \infty,\ z \to \infty: &\quad c = c^*\ .
\end{aligned} \tag{12.3}$$

For other kinds of experiments, the second, Cottrell condition, would be replaced with another.

For any given system, the current $i(t)$ at any one time is given by

$$i(t) = 2\pi n\mathcal{F}D \int_0^a r\frac{\partial c}{\partial z}\bigg|_{(z=0)} \mathrm{d}r\ . \tag{12.4}$$

The classic study of Saito [31] is often cited. Saito derived the steady state current $i_{ss}$ at a UMDE, setting the left-hand side of (12.2) to zero, and arrived at

$$i_{ss} = 4nFc^*Da\ . \tag{12.5}$$

Amatore [13] and Zoski [32] write that the problem was first solved by Saito. In fact, Soos and Lingane [25] preceded Saito by 2 years, having developed the same equation, in a slightly more complicated form, cited by Liu et al. [33], who themselves derived a more general solution for different shaped electrodes. Moreover, while these are the first solutions in the context of electrochemistry, the same mathematics applies to other phenomena such as electric fields and heat transport. Saito himself cites the book by Grigull [34], which is a later edition of the original 1921 book by Gröber [35], all presenting essentially the same solution for heat transport. Aoki [12] cites Tranter [36] for the same solution in the context of electric fields, and Weber (1873!) [37] is cited by Sarangapani and de Levie [38] for the solution, again with electric fields. That article also shows the singularity in the flux at the edge of a planar disk, predating the same discovery in electrochemistry by many years. Carslaw and Jaeger also solved the heat conduction problem at a circular disk, both for the field and flux [39], and in the first edition of 1947 for the field only. In this book, the solution (12.5) will hereafter be referred to as the Soos–Saito solution.

The steady-state value is the normalising quantity for the current as a function of time in most studies except those where the Cottrell current is used as the reference value. Saito also derived the concentration profile at steady state. It was printed incorrectly in the paper [31], and Crank and Furzeland [40] present the correct equation:

$$c = c^* \left(1 - \frac{2}{\pi} \sin^{-1} \left\{ \frac{2a}{\sqrt{z^2 + (a+r)^2} + \sqrt{z^2 + (a-r)^2}} \right\}\right) \tag{12.6}$$

for $z > 0$ and

$$c = c^* \left(1 - \frac{2}{\pi} \sin^{-1} \left\{ \frac{a}{r} \right\}\right) \tag{12.7}$$

for $z = 0, r > a$.

We want the current $i(t)$, a function of time. Aoki and Osteryoung [41] presented short- and long-time analytical expressions, the short-time one being the above (12.1), with $A = \sqrt{\pi} = 1.77$, that is

$$\frac{i_{UMDE}}{i_{Cott}} = 1 + \pi^{\frac{1}{2}} \left(\frac{Dt}{a^2}\right)^{\frac{1}{2}} . \tag{12.8}$$

This is better expressed directly without reference to the Cottrell current, and Aoki and Osteryoung [42] provide the short-time solution

$$i = i_{ss} \left( \frac{\sqrt{\pi}}{2\sqrt{\tau}} + \frac{\pi}{4} + 0.094\sqrt{\tau} \right) \tag{12.9}$$

where $i_{ss}$ is the steady-state current as given in (12.5) and $\tau$ is the normalised time (more on that below). However, this approximation is not quite correct. Aoki and Osteryoung state [42] that they have adjusted the third coefficient (0.094) so that the approximation better meshes with their long-time approximation (see below). This was pointed out by Phillips and Jansons [43], who then presented the correct series:

$$i = i_{ss} \frac{\sqrt{\pi}}{4} \left( 2\tau^{-\frac{1}{2}} + \pi^{\frac{1}{2}} + \tfrac{1}{4}\tau^{\frac{1}{2}} \right) \tag{12.10}$$

which makes the last coefficient in (12.9) 0.111 rather than 0.094. This slightly extends the range of applicability of the approximation.

For longer times, a complicated expression was derived by Aoki and Osteryoung [41], but it was incorrect, as pointed out by Shoup and Szabo [30], who gave the correct expression, also given by Aoki in 1993 [12], with one more term. The first few terms of the long-time solution [12] are

$$I = i_{ss} \left( 1 + 0.71835\tau^{-\frac{1}{2}} + 0.05626\tau^{-\frac{3}{2}} + 0.00646\tau^{-\frac{5}{2}} \ldots \right) \tag{12.11}$$

For very large $\tau$, this becomes the steady state Soos–Saito value (12.5). The steady state value might be considered the only exactly known expression, all others being approximations.

Shoup and Szabo also provide a general approximation that they state is accurate to 0.6 % at all values of $\tau$:

$$I = i_{ss}\left[0.7854 + 0.8862\tau^{-\frac{1}{2}} + 0.2146\exp(-0.7823\tau^{-\frac{1}{2}})\right] . \tag{12.12}$$

A word is needed here about the definition of normalised time, in this context usually given the symbol $\tau$. Most workers, including Shoup and Szabo, use the definition

$$\tau = \frac{4Dt}{a^2} . \tag{12.13}$$

This is the definition assumed in the work of Shoup and Szabo [30] and Aoki and coworkers [12, 41, 42] and also by Gavaghan in some recent works [44, 45]. The above three formulae (12.9), (12.11) and (12.12), are those for this definition of normalised time. One inconvenient side-effect of the definition is that, when one normalises the diffusion equation (12.2), using the new dimensionless variables' definitions

$$\begin{aligned} C &= c/c^* \\ R &= r/a \\ Z &= z/a \\ \tau &= 4Dt/a^2 , \end{aligned} \tag{12.14}$$

the diffusion equation becomes

$$\frac{\partial C}{\partial \tau} = \frac{1}{4}\left(\frac{\partial^2 C}{\partial R^2} + \frac{1}{R}\frac{\partial C}{\partial R} + \frac{\partial^2 C}{\partial Z^2}\right) \tag{12.15}$$

in which there is the factor 1/4. This is avoided by a different normalisation of time. Reverting now to the present context and using the symbol $T$ ($\tau$ being here reserved for an observation time), we have the normalisation

$$T = \frac{Dt}{a^2} \tag{12.16}$$

which eliminates the leading fraction. Using this definition, which the present authors prefer (and which was also used by Flanagan and Marcoux [29]), the dimensionless diffusion equation is now as one expects,

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial R^2} + \frac{1}{R}\frac{\partial C}{\partial R} + \frac{\partial^2 C}{\partial Z^2} \,. \tag{12.17}$$

The normalised set of boundary conditions is then

$$\begin{aligned}
T = 0,\ \text{all}\ R, Z: &\quad C = 1 \\
T > 0,\ R \le 1,\ Z = 0: &\quad C = 0 \\
R = 0,\ Z > 0: &\quad \frac{\partial C}{\partial R} = 0 \\
R > 1,\ Z = 0: &\quad \frac{\partial C}{\partial Z} = 0 \\
R \to \infty,\ Z \to \infty: &\quad C = 1
\end{aligned} \tag{12.18}$$

and the current integration formula (12.4) becomes

$$I(T) = \frac{\pi}{2}\int_0^1 R\frac{\partial C}{\partial Z}\bigg|_{Z=0} \mathrm{d}R \tag{12.19}$$

which converges to unity at very large $T$. Also, the constants in the above three solution approximations change. They become the following new formulae: The short-time solution of Aoki and Osteryoung (12.9) is then

$$I = \left(\frac{1}{4}\left(\frac{\pi}{T}\right)^{\frac{1}{2}} + \frac{\pi}{4} + 0.188\, T^{\frac{1}{2}}\right) \tag{12.20}$$

($i_{ss}$ does not of course change), whereas the more correct formula as presented by Phillips and Jansons [43] is

$$I = \frac{\pi^{\frac{1}{2}}}{4}\left(T^{-\frac{1}{2}} + \pi^{\frac{1}{2}} + \tfrac{1}{2}\, T^{\frac{1}{2}}\right) \,. \tag{12.21}$$

The long-time solution (12.11) becomes

$$I = \left(1 + 0.35918\, T^{-\frac{1}{2}} + 0.007033\, T^{-\frac{3}{2}} + 0.000202\, T^{-\frac{5}{2}} \ldots\right) \tag{12.22}$$

and the general approximation (12.12) becomes

$$I = \left(0.7854 + 0.4431\, T^{-\frac{1}{2}} + 0.2146 \exp(-0.3912\, T^{-\frac{1}{2}})\right) \,. \tag{12.23}$$

Note that the second coefficient in (12.22) was misprinted as 0.25918 in our paper [46], although all computations were performed with the correct value.

There is, however, a much better pair of solutions, obtained by Mahon and Oldham [47] and simplified a little a year later [48]. They used what they call the "Cope–Tallman" method, involving the Green function, to find much improved short-time and long-time solutions for the current at a disk electrode. Their formulae express currents at $T$ values as defined above (12.13) (previously designated by $\tau$), and normalised by $\pi nFDac^*$, rather than the steady-state value. Here they are converted to the present scale by the simple expedient of a multiplication factor. The short-time approximation is then[1]

$$I = \frac{\pi}{4}\left((\pi T)^{-\frac{1}{2}} + 1 + \frac{1}{2}\left(\frac{T}{\pi}\right)^{\frac{1}{2}} - 0.120031163\,T + 0.0132727696\,T^{\frac{3}{2}}\right) \tag{12.24}$$

and their long-time approximation is

$$I = 1 + \frac{\pi}{4}\Big(8\pi^{-\frac{5}{2}}\,T^{-\frac{1}{2}} + 8.9542\times10^{-3}\,T^{-\frac{3}{2}} - 2.5664\times10^{-4}\,T^{-\frac{5}{2}} \\ - 2.2312\times10^{-4}\,T^{-\frac{7}{2}} + 2.7628\times10^{-5}\,T^{-\frac{9}{2}}\Big)\,. \tag{12.25}$$

Unlike the approximations of Aoki and Osteryoung, these two actually overlap in their regions of applicability, as is seen in the next section.

#### 12.1.1.1 Ranges of Applicability

From some simulations, in which reference current values were computed over a large range of times [46], it was possible to assess the range of applicability of the approximations. If we require that currents be accurate within 0.1 %, then the short-time approximation of Aoki and Osteryoung, in its corrected form (12.21), is accurate in the range $0 \le T \le 0.03$, while their long-time expression (12.22) is applicable for $T > 0.5$. There is thus a gap in the range $0.03 < T < 0.5$, in which neither approximation yields good values. In the gap range, it was found that errors due to the approximations peak at about 0.8 %.

The universal approximation of Shoup and Szabo (12.23) has a similar, but wider gap range $0.002 \le T \le 10$, within which it has two excursions as high as 0.6 % in amplitude. This is in accord with the original statement in the work [30], guaranteeing a maximum error of 0.6 %.

[1] These values were kindly communicated to us by Dr. Peter Mahon, Swinburne University of Technology, Melbourne, Australia.

Finally, the same study [46] showed that the short-time solution of Mahon and Oldham (12.24) is 0.1 % accurate in the range $0 \leq T \leq 1$, and the long-time approximation in the range $T \geq 0.4$. Thus, the two formulae yield accurate current values over the whole time scale without a gap.

#### 12.1.1.2 LSV

For LSV, the diffusion equation for a UMDE is a little different from that for a potential jump. The LSV case can be considered as one of a group of possible cases, in which the characteristic time is defined independently of the disk radius while the space variables are rendered dimensionless using the disk radius, as described before. In general, let that characteristic time be $\tau$. For LSV, as described on page 29, it is the time taken by the potential to sweep over one dimensionless potential unit and the scan rate parameter $a$ was introduced, Eq. (2.91). It has the dimension of $\mathrm{s}^{-1}$, so that $\tau = a^{-1}$. Note that the scan rate parameter $a$ should not be confused with the radius of a UMDE which has the same symbol. The dimensionless time is then defined by

$$T = t/\tau = \frac{n\mathcal{F}}{\mathcal{RT}}vt, \tag{12.26}$$

while the distances $(r, z)$ and concentration $c$ are normalized, as above, by the disk radius $a$ and a reference concentration $c^*$, respectively. Introducing these dimensionless variables into Eq. (12.2) gives

$$\frac{\partial C}{\partial T} = \frac{1}{P^2}\left(\frac{\partial^2 C}{\partial R^2} + \frac{1}{R}\frac{\partial C}{\partial R} + \frac{\partial^2 C}{\partial Z^2}\right). \tag{12.27}$$

There is now an extra parameter $P$ in the dimensionless form of the diffusion equation, following the ideas of Heinze [49] and Aoki et al. [50]. For LSV, $P$ is given by

$$P = \left(\frac{a^2 n\mathcal{F}v}{\mathcal{RT}D}\right)^{\frac{1}{2}}, \tag{12.28}$$

or, in more general form,

$$P = \frac{a}{\sqrt{D\tau}}. \tag{12.29}$$

showing that $P$ is the ratio of the disk radius to the Nernst diffusion layer thickness. The symbol $P$ is normally rendered as $p$, but this collides with our $p$ for the dimensionless potential. Following definitions (12.28) $P$ can be regarded as a scan rate parameter for potential scan experiments at a disk electrode of radius $a$. A small

$P$ value means a slow LSV sweep rate for a given disk radius $a$, resulting in a sigmoidal steady-state response, while a large value means a fast sweep rate with the electrode behaving more like a planar (shrouded) electrode. Aoki et al. [50] present an approximate solution for the LSV current in terms of some integrals that must be evaluated numerically, and an expression for the peak current as a function of $P$, accurate to 0.23 % for any $P$.

The above impinges on the choice of a maximum $Z$ and $R$ values which must be set such that they contain a sufficient number of $\sqrt{D\tau}$ units, according to the experiment. This will be discussed in some detail in the simulation section, below.

## 12.1.2 Other UMEs

Theories for other UMEs are not as well developed as those for the UMDE but some approximations do exist. There are some reasonable approximations for the ultramicroband electrode, UMBE. This is a relatively long strip, most often flush with the insulating plane it is embedded in. Since it is relatively long, that dimension can be ignored in the *pde*s describing transport at the electrode. The diffusion equation is very similar to that for the UMDE (12.2), removing the term in $r^{-1}\partial c/\partial r$, and replacing the disk radius $a$ with the half-width $w/2$ of the strip, the normalisation is also very similar. There has been a series of theory papers on this electrode. Aoki et al. [51] provide, for short times, for the UMBE of half-width $w$ and length $l$,

$$\frac{i(T)}{nFDc^*l} = \frac{1}{\sqrt{\pi T}} + 1 - \frac{(2T)^{3/4}}{\pi} \exp(-1/(8T))\ U(2, (2T)^{-1/2}) + \dots\ , \tag{12.30}$$

where

$$T = Dt/w^2 \tag{12.31}$$

and $U$ is the parabolic cylinder function [52]. Szabo et al. [53] also derived the first two terms on the right-hand side and find that the formula then holds to within 1.3 % up to $T < 0.4$. A longer-time solution, for higher values of $T$, also accurate to 1.3 % is then given by them,

$$\frac{i(T)}{nFDc^*l} = \frac{\pi \exp(-2\sqrt{\pi T}/5)}{4\sqrt{\pi T}} + \frac{\pi}{\ln\left[(64\exp(-\gamma)T)^{\frac{1}{2}} + \exp(\frac{5}{3})\right]} \tag{12.32}$$

in which $\gamma = 0.5772156649\dots$ is the Euler constant. Szabo et al. do present a better long-time approximation, but it is in Laplace-transformed form. Their paper is interesting for another reason, the relationship between the flat band

and hemicyclindrical electrodes, see below. Sentamamarai and Rajendran have developed another model in terms of Padé approximations to their series solution [54] which seems quite accurate. Amatore et al. [55] published a general theoretical paper on these and state that at long times, the current at a band of width $w$ is the same as that at a cylinder of radius $w/4$, previously concluded in 1987 [53, 56].

Both Aoki et al. and Szabo et al. normalise time as $Dt/w^2$, whereas Coen and coworkers [57–59] who have presented solutions to the diffusion-limited current at a band electrode in the form of integral equations, normalise by the half-width of the band, resulting in $T = 4Dt/w^2$. The integral equations must be evaluated numerically, so this can be regarded as simulation. Some simulation details for the UMBE are given below. In this work, the normalisation by the half-width of the band, $w/2$ is also favoured, following Coen et al. Then the dimensionless diffusion equation, with $X$ the coordinate across the band, with $X = 0$ along the centre, and $Z$ the coordinate normal to the band surface both normalised by $w/2$, is

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} + \frac{\partial^2 C}{\partial Z^2}\,. \tag{12.33}$$

Taking into account the symmetry of the system and thus applying the above only to the right half of the band geometry, the boundary conditions are

$$\begin{aligned}
&T = 0,\ \text{all}\ X, Z: \quad C = 1\\
&T > 0:\\
&\quad Z = 0,\ 0 \le X \le 1: \quad C = 0\\
&\quad Z \to \infty,\ X \to \infty: \quad C = 1\\
&\quad X = 0,\ Z > 0: \quad \partial C/\partial X = 0\\
&\quad Z = 0,\ \ X > 1: \quad \partial C/\partial Z = 0
\end{aligned}$$

and the normalised current

$$I(T) = \frac{i(T)}{nFDc^* wL} = 2\int_0^1 \left.\frac{\partial C}{\partial Z}\right|_{Z=0} \mathrm{d}X \tag{12.34}$$

with $L$ the normalised length of the band. The integration from $X = 0$ and the factor 2 are given by symmetry.

Aoki and Tokuda [60] present an approximate solution for the LSV current in terms of integrals to be evaluated numerically, and for the peak current as a function of $P$, accurate to 2.1 % for any $P$.

A very recent paper by Bieniasz [61] describes general solutions of extraordinary accuracy valid for all times and these can now serve as standards. Bieniasz followed up with polynomial approximations of the same accuracy which speed up computation dramatically [62].

Other UMEs are those with hemispherical and hemicylindrical geometries, long since understood (see Bard and Faulkner [2] but see [63] for the capped or finite-length cylinder electrode), ultramicroring electrodes, which [58] can be regarded as infinite bands, a conical ultramicroelectrode [64], a sphere-cap ultramicroelectrode [65], the UMDE inside an insulating conical well [66]. Oblate hemispherical UMEs have been considered [67, 68] and a large variety of shapes by Oldham [69] and Zoski [32]. Arrays of all types of UMEs, see the later Sect. 12.5, have been used and analysed. A range of ultramicroring electrode thicknesses was considered by Amatore et al. [70], who found that a thick ring (that is, the ring width is comparable with the diameter) behaves more like a disk, while a thin ring approaches a band, as observed previously [58]. The Compton group has investigated what they call the shrouded ring system [71–73], as a model for a partially blocked electrode. There is much interest in the scanning electrochemical microscope (SECM), where no theory has been developed to date, and simulation is the rule here. The method was invented by Engstrom et al. [74], and recent papers on the subject of the last 10 years are [75–91]. A section on UMEs and their simulation is found in recent reviews [16, 92].

### 12.1.3 Some Relations

It is intuitively obvious that at longer times, when the diffusion layer thickness far exceeds the radius of a disk or hemisphere (for small $P$), or of the width of a band or the hemicylinder, currents at flat electrodes (disk, band) must resemble those at round electrodes (hemisphere, hemicylinder). Some relations between these have been established. Oldham found [93] that the steady-state currents at an ultramicrodisk and ultramicrohemisphere are the same if their diameters along the surfaces are the same and he later unified UMEs of widely different shapes in terms of the area of the interface between the electrode and electrolyte and an accessibility factor [69]. Thus for an ultramicrodisk of radius $a$, the steady-state current is the same as that at an ultramicrohemisphere of radius $2a/\pi$. At band or hemicylindrical electrodes, there is no steady state, but there still exists a relationship between them at long times. Szabo et al. [53], using the Laplace transform analytical solution for the current at an ultramicroband, and comparing it with that at an ultramicrohemicyclinder, conjectured that the current at a band of width $w$ has the same long-time current as at a hemicyclinder with radius $w/4$. This was borne out by simulated values, and analysed mathematically by Bieniasz [61].

## 12.2 Simulations

In the present context, we are interested in how best to simulate electrochemical processes at a two-dimensional electrode. The flat disk, the UMDE, is taken as an example, as the techniques that have been developed for it are the same as those for the other geometries. However, the band electrode is also briefly mentioned.

All the UMEs already mentioned have been simulated.

The UMDE evinces strong edge effects or very uneven current densities along the radius of the disk. It shares this problem with all the other flat UMEs such as ultramicrobands or -rings. The exceptions of course are the hemispherical or hemicylindrical UMEs, which have no effective edges and behave as half of a sphere or cylinder, and also deeply recessed disks or bands, which approach the shrouded types.

It was early realised (for example by Crank and Furzeland in 1977 [40]) that the singularity at such edges will degrade overall accuracy in a simulation. In fact, Gavaghan points out [44, 94] that because of this effect, the simulation error in calculated concentrations is of $O(h^{1/2})$, $h$ being the interval size in space, near the edge. This is rather poor, so that unless special techniques (see below) are used, these simulations can be very cpu-intensive. The worst methods for simulating such systems are the explicit method and equal intervals in $(r, z)$ space (for the UMDE); nevertheless, both have been used [95–98]. Motivation for using simple explicit or splitting methods is said to be that the more efficient implicit methods are not easy to implement in two dimensions. For this reason, hopscotch [99] and ADI [100] (see below) have been favourites in this area, as they obviate the need for solving largish banded systems of equations.

Flanagan and Marcoux [29] were the first to attempt a UMDE time-marching simulation, in order to find the constant in the approximation of Lingane's equation (12.1); they used the explicit method. Crank and Furzeland [40] addressed the steady state for the UMDE and described some of the problems; they also briefly mention time-marching simulations. Their work appears to have come just after that of Evans and Gourlay [101], who used hopscotch. They also found some oscillatory behaviour of the solution, which is not always mentioned. As Gourlay realised [99], hopscotch is mathematically related to ADI, which in turn approximates Crank–Nicolson, known to be oscillatory in response to initial discontinuities such as a potential jump (more on this problem below).

Heinze is usually cited as the first to do a thorough study of the UMDE simulation [28], using ADI for the potential step problem. He followed this with an LSV study [102], and used unequal intervals in the next study [103], to come to grips with the edge effect. Meanwhile, Shoup and Szabo [30] applied hopscotch to the problem, and this method has continued to be used to some extent [104–107], citing only more recent works among many other earlier ones, also with other UMEs. As mentioned in Chap. 9, hopscotch has a problem, called "propagational inadequacy" by Feldberg [108, 109]. Hopscotch becomes inaccurate for large time intervals, which are one motivation for using stable algorithms. As well, as mentioned above, hopscotch does give rise to some initial oscillations for potential step simulations. These are less severe for small time intervals, however, and perhaps for that reason are not always mentioned or considered serious. Safford and Weaver [110] addressed the nontrivial problem of uncompensated resistance and double layer capacity in these simulations. ADI, as mentioned, was used on some occasions [28, 111], and has enjoyed a lot of use in more recent years [112–123], citing only work of the last 5 years, to avoid an over-long list here. Finally, Crank–Nicolson remains an attractive method, if sparse matrix solvers are used and its oscillatory response is damped, which can be done by some rather simple expedients, as described [124, 125], such as either starting a potential jump simulation by subdividing the first step in time into sub-intervals (the Pearson method [126]) or—even simpler—using BI for the very first 1–4 steps [124, 127–131]. BI is known to have a very steady response to initial transients, and it turns out that after 1–4 BI steps, when CN is resumed, no more oscillations are seen. What is more, this was found to be especially effective with UMDE simulations in conformal space [124].

It was soon realised that unequal intervals, crowded closely around the UMDE edge, might help with accuracy, and Heinze was the first to use these in 1986 [103], as well as Bard and coworkers [132] in the same year. Taylor followed in 1990 [111]. Real Crank–Nicolson was used in 1996 [133], in a "brute force" manner, meaning that the linear system was simply solved by LU decomposition, ignoring the sparse nature of the system. More on this below. The ultimate unequal intervals technique is adaptive FEM, and this too has been tried, beginning with Nann [134] and Nann and Heinze [135, 136], and followed more recently by Gavaghan et al. [137] and a series of papers by Harriman et al. [138–144], some of which studies concern UMBEs and recessed UMDEs. One might think that FEM would make possible the use of very few sample points in the simulation space; however, as an example, Harriman et al. [143] used up to about 2000 nodes in their work. This is greater than the number of points one needs to use with conformal mapping and multi-point approximations in finite difference methods, for similar accuracy.

In general, the finding by Rudolph [145], that in one-dimensional simulations, direct discretisation on an unequally spaced grid, rather than equal spacing on a transformed grid, is best, does not appear to apply to UME simulations. Gavaghan made a very thorough study of UMDE simulations [44, 45, 94, 146] and concluded that the above-mentioned $O(h^{1/2})$ behaviour limits the convergence obtained. Much better are transformations by conformal mapping, to eliminate the edge singularity. Such conformal maps were used as early as 1966 by Newman in a mathematical study of the rotating disk [147] and by Saito [31], who worked out the steady-state current at a UMDE, used conformal mapping; in fact, he used the same formula as later applied by Michael et al. [148]. Safford et al. [149] used the same formula, and more is said about this technique below.

Other UMEs have been simulated, and are briefly mentioned here. Ultramicroband or ultramicrohemicyclinders were simulated starting in 1986 [150, 151], mostly using hopscotch. Coen et al. [57], followed by Cope et al. [58, 59] used the integral equation method (see Chap. 9) to simulate ultramicroband or -rings. Jin and coworkers used their finite analytical method (FAM) method on ultramicrobands and -rings, as well as on an ultramicro-oblate spheroidal electrode [152–157]. Varco Shea and Bard [158] used the explicit method on ultramicroband arrays, Bieniasz and Britz [159] simulated chronopotentiometry at an ultramicroband using a Rosenbrock method. A UMDE at the bottom of an insulating conical well [66] and a conical-tip electrode [160] were simulated. Reference current tables were generated by simulation for the UMDE [46], for the UMBE (band electrode) [161], and for long and capped ultramicrocylinder electrodes [63]. Simulations of interdigitated array (IDA) UMBEs are detailed in Sect. 12.5, where also generator-collector systems are treated. The SECM with its similarity to a UMDE with a close thin-layer-cell-like opposite wall and various boundary conditions there demands simulation, and considerable work along these lines has been done [76–83, 85–87, 89, 90, 95, 162–181]. There is some interest in conducting polymers embedding enzymes, covering a UMDE [182–189]. This list is probably not exhaustive.

## 12.3 Simulating the UMDE

The ways to simulate our chosen example, the UMDE, are described here. The integral equation approach, taken by Coen and coworkers over a number of years [57–59, 190–193] for UMBEs, could be used on the UMDE as well, and has been [191], see also extensive monograph on this method by Bieniasz [194], who has published extensively on the integral equation method, having improved on former solution techniques [194, 195, 195–211]. The reader is referred to these publications for the method. Also, although the adaptive FEM approach might be thought to be about the most efficient, and has been developed by a few workers (see above, references to Nann and Heinze, and Harriman et al.), it does not seem the method of choice; it is not trivial to program, and as Harriman et al. found, it appears that a rather large number of nodes were required. The reason is probably that this is a kind of discretisation in the original cylindrical $(R, Z)$ space, where convergence, as mentioned above, is of $O(h^{1/2})$ [94], and many nodes are needed to get reasonable results. This is of course also the case with finite differences, as described below. Discussion here is confined to the use of finite difference methods for UMDE simulation, since these serve as guides for the simulation of the other 2D electrodes.

One has, then, two choices: to apply finite difference discretisation either directly to a grid of points in the cylindrical $(R, Z)$ space, or to a transformed space. In one dimension, it has been found [145] that direct discretisation without transformation is better. In the case of 2D simulations where edge effects are seen, this is not the case, and transformation is better. Both approaches are described here.

## 12.3.1 Methods of Solution

Whether the simulation is on a direct discretisation of the equations in cylindrical or transformed coordinates, the discretisation process results in a (usually) linear system of ordinary differential equations, that must be solved. In two dimensions, the number of these will often be large and the equation system is banded. One approach is to ignore the sparse nature of the system and simply to solve it, using lower-upper decomposition (LUD) [212]. The method is very simple to apply and has been used [133, 213, 214]—it is especially appropriate in curvilinear coordinates and multipoint derivative approximations, where the system is of minimal size [214], and can outperform the more obvious method, using a sparse solver such as `MA28` (see later). However, many simulators tend to prefer other methods, that avoid using implicit solution in two dimensions simultaneously but still are implicit. Of these, two stand out.

### 12.3.1.1 Hopscotch

One method apparently still in some favour, is hopscotch, already mentioned in Chap. 9, extended to two-dimensional problems by Gourlay [215], as did Evans [101] in the same year. Despite the indication by Feldberg [109] and Carnahan et al. [108] that hopscotch suffers from "propagational inadequacy" (Feldberg's term), it has continued to be used by electrochemists [105–107, 148, 216–222]. Danaee and Evans [223] described a composite method using points and blocks, that may have fixed the propagation problem but this has not been tried in electrochemical work. Hopscotch is in some ways similar to Crank–Nicolson and like that method, has an oscillatory error response, as is seen in [223] and [218], among others. It has been described in Chap. 9.

### 12.3.1.2 ADI

Another, more popular, method is that of alternating direction implicit, or **ADI**. The most popular variant (there are several) is that of Peaceman and Rachford [100]. Briefly, on a given grid, a time step is divided into two half steps. In the first step, all rows are advanced by $\delta T/2$ discretising implicitly along the row, but explicitly in the other direction at each point (see, for example, Press et al. [212, p. 847]). For unknown $\mathbf{U}$, denoting the discrete second derivative operator along $X$ as $\delta_X^2$ and along $Y$ as $\delta_Y^2$, the first step at indices $i, j$ and advancing from time step $n$ to $n+1$,

$$U_{j,i}^{n+\frac{1}{2}} = U_{j,i}^{n} + \frac{1}{2}\lambda\left(\delta_X^2 U_{j,i}^{n+\frac{1}{2}} + \delta_Y^2 U_{j,i}^{n}\right) \tag{12.35}$$

followed by the same process but now implicit along $Y$ and explicit along $X$,

$$U_{j,i}^{n+1} = U_{j,i}^{n+\frac{1}{2}} + \frac{1}{2}\lambda \left( \delta_X^2 U_{j,i}^{n+\frac{1}{2}} + \delta_Y^2 U_{j,i}^{n+1} \right) . \tag{12.36}$$

This is a formulation assuming equal intervals $h$ in both directions so that we have $\lambda = \delta T/h^2$. It can easily be extended to unequal intervals. In practice, at each solution along a row or column, a simple tridiagonal equation system is solved, using the Thomas algorithm as already described in Chap. 8.

The Peaceman–Rachford ADI method is second-order with respect to time, and performs similarly to Crank–Nicolson. Indeed, Lapidus and Pinder write [224, p. 246] "... is a variation of the Crank–Nicolson approximation". It is known to be unconditionally stable [225]. As with CN, ADI may show some error oscillations, as also evidenced by the fact that some habitually use expanding time intervals when employing ADI [226–231], although some of these same workers on occasion also use equal time intervals [232, 233].

ADI was first used for electrochemistry by Heinze et al. [28, 103] to simulate the UMDE, and continues to be used by many. Naming only works published in the last 5 years (2010–2015) to avoid an overlong list we have [112–120, 122, 123, 234]. The method appears to be favoured by workers simulating the SECM and many papers have been published using ADI [163–166, 169–172, 174, 179, 235–240].

#### 12.3.1.3 Some Other Methods

There have been some attempts at using other methods regarded as more efficient. Stone [241] introduced and analysed the strongly implicit method **SIP** of solving the system of discrete equations and a book appeared on it [242] 1 year later. It has been used by electrochemists, often compared with other efficient methods [243–249], and the method compares favourably with, for example, BI or ADI. It has not to our knowledge been applied to electrochemical simulations since those years, perhaps because of the programming difficulties.

What might be called the Krylov method has been attempted and also compared. It is based on an article by Krylov [250], and is described more accessibly in the books of Saad [251, 252] and Wesseling [253]. Some electrochemists have tried it out [246, 247, 254–256] but generally in comparisons, SIP wins. As with SIP, it appears to have fallen out of favour.

Lastly we have the multigrid method, devised by Brandt in 1977 [257], which also has been compared with SIP and Krylov by some electrochemists [247, 248, 258–260] and found very efficient. It may be worth more work.

### *12.3.2 Direct Discretisation*

The discussion to follow refers to the example program `UMDE_DIRECT` (Appendix E). Consider Fig. 12.2, which is the normalised version of Fig. 12.1, the disk edge now lying at $R = 1$. When discretising directly here, we must decide on maximum values for $R$ and $Z$, as indicated, see below. Clearly, given the poor convergence, many grid points are needed, which in turn points to the use of unequal point spacing in both axes. This was done by Taylor et al. [111] and by Gavaghan in his three-part study [44, 45, 146], in which he employed a grid similar to that pictured in Fig. 12.3.

There is a slight complication in the setting of the maximum $R$ and $Z$ values. The procedure depends on whether (12.17) or (12.27) is simulated. In the former case, we have

$$Z_{max} = 6\sqrt{T_{max}} \tag{12.37}$$

where $T_{max}$ is the number of time units over which the experiment runs. These units are defined by (12.16) for this case, making the Nernst diffusion layer thickness equal to the UMDE radius for one time unit. For $R_{max}$, one should probably use the formula

$$R_{max} = 1 + 6\sqrt{T_{max}}\ , \tag{12.38}$$

taking into account that the extent is measured from the disk edge, one disk radius from the origin.

If the characteristic time is defined independently of the disk radius (as it is with LSV) and diffusion equation (12.27) results, the Nernst diffusion layer thickness is

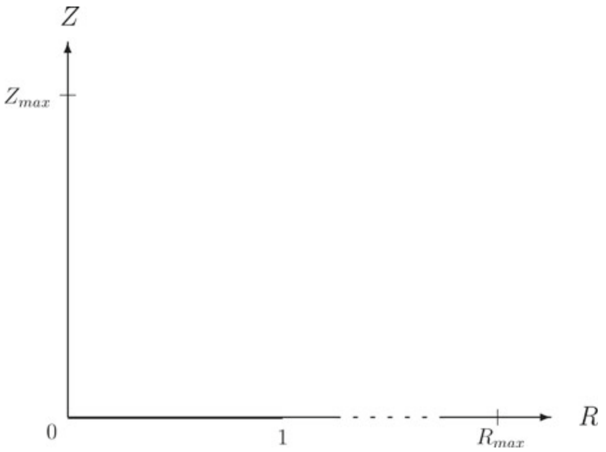


**Fig. 12.2** Coordinate system for the UMDE

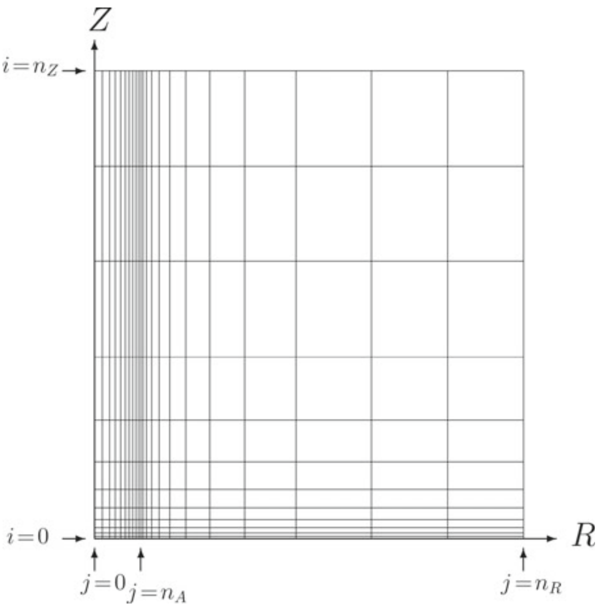


**Fig. 12.3** UMDE unequally spaced grid in $(R, Z)$ coordinates

dependent only on the number of these time units. So if the characteristic time is $\tau$ and the maximum duration of the experiment is $\tau_{max}$ (giving $T_{max} = \tau_{max}/\tau$), then the final diffusion layer thickness is $\sqrt{D\tau_{max}}$. Then, in dimensionless distance units (normalisation being division by the disk radius $a$), this becomes, after multiplying by 6 and noting (12.29),

$$Z_{max} = \frac{6}{P}\sqrt{T_{max}} \tag{12.39}$$

and for $R$,

$$R_{max} = 1 + \frac{6}{P}\sqrt{T_{max}} \ . \tag{12.40}$$

Note that $P = 1$ for the simulation of a potential step experiment, so that Eqs. (12.39) and (12.40) become identical to Eqs. (12.37) and (12.38). In the case of LSV $\tau_{max}$ is the time to scan a potential from a starting value, $E_{start}$, taking some potential units $\mathcal{RT}/n\mathcal{F}$, to the final potential $E_{stop}$ with a scan rate $v$. The characteristic time $\tau$ was previously defined as the time to sweep through one $p$-unit, see Eq. (2.92) on page 29. Therefore $T_{max}$ is equivalent to the dimensionless potential range $p_{range}$ of the LSV simulation and $Z_{max}$ becomes

$$Z_{max} = \frac{6}{P}\sqrt{p_{range}} \tag{12.41}$$

and $R$,

$$R_{max} = 1 + \frac{6}{P}\sqrt{p_{range}}\ , \tag{12.42}$$

with $P$ given by Eq. (12.28), page 259. For a CV experiment, where the potential returns to the start potential, $p_{range}$ is simply multiplied by two. This will become a little more complicated later, when the space is mapped into new coordinates, and limits in terms of these must be set. In Fig. 12.3, the number of nodes (lines) is held small, in order not to confuse the picture. The grid is chosen such that there are expanding intervals in both the $Z$ direction and in the two $R$ directions away from and on either side of the line $R = 1$. There are $n_A$ intervals between $R = 0$ and the disk edge, and a total of $n_R$ between the origin and the point at which $R = R_{max}$, the outer boundary for $R$. The positions for $Z$, indexed with $i$, begin at zero and $n_Z$ is the point at which $Z = Z_{max}$.

The unequal grid was generated with the aid of the Fortran function EE_FAC (Appendix E and described in Sect. 7.2). One needs to decide the numbers of points in each of the three ranges, and the minimum intervals, whereupon EE_FAC produces the required $\gamma$ values for the expansion. One range goes from $R = 1$ backwards to $R = 0$. In the other direction, the expansion finds the final variable point at $R_{max}$. The third range is simply $0 \le Z \le Z_{max}$.

In the third edition of this book, four-point spatial derivative approximations on an unequal grid were advocated. However, when working directly in $(R, Z)$ space, the most efficient way to solve the resulting system of discrete equations is to use a sparse solver such as MA28 [261, 262], and a recent study [214] comparing various methods of simulating transport at a UMDE concluded that in $(R, Z)$, three-point approximations, despite their lower approximation order (see Chap. 3), are more efficient in terms of computing time to achieve a given target accuracy. This rather surprising result is probably due to the internal workings of MA28. The use of three-point approximations conveniently brings with it easier programming, and therefore less danger of programming error.

We are now ready to apply the discretisations, but must decide on the vector of unknown concentrations at all the grid points in Fig. 12.3. It is convenient to include the outer boundary points, setting these to known values in the large linear system to be generated. Thus we note that the total number $N$ of unknowns (grid points) is given by

$$N = (n_R + 1)\,(n_Z + 1)\ . \tag{12.43}$$

A convenient ordering of the grid points is achieved by arranging the concentration grid one row at a time, going upwards from the bottom ($Z = 0$). The numbering of the elements in the unknowns vector is then the following. Index $k$ of the vector element corresponding to the grid value $C_{ij}$ at $(Z_i, R_j)$, $(i = 0 \ldots n_Z,\quad j = 0 \ldots n_R)$ is

$$k = i\,(n_R + 1) + j + 1\ , \tag{12.44}$$

so that the whole grid is now mapped into $N$ elements. The map is conveniently generated in the example program `UMDE_DIRECT` by the function `KMAP`. There is a corresponding function `UNMAP` which calculates $i$ and $j$ from a given $k$, needed in order to fold newly calculated concentration values back into the grid, which might be desirable for current calculations or plotting.

With $N$ elements as the unknown vector, an $N \times N$ matrix is clearly required for the solution of the linear system of discrete equations. This can be rather large. One approach, that has been tried [133, 214], is to ignore this problem and to actually generate the large matrix and let the system be solved by a suitable solver. This limits the size of $N$, however, leading to somewhat inaccurate simulations. However, when discretising, one notes a large number of zero elements in the matrix, which is banded, and this suggests a sparse matrix technique.

The program package **MA28** [261] was found to be useful. The package can be downloaded from the Harwell site [262]. It is written in `Fortran IV`, but there is no problem in adapting it to `Fortran 90`, thanks to the (so far) downward compatibility of the latest language definition. `MA28` does an LU-decomposition of a sparse matrix, allowing efficient solution by back-substitution after the initial LU-decomposition. What is more, for those cases where the matrix varies with time—as is the case in, for example, second-order homogeneous chemical reactions and in time-varying boundary conditions—`MA28` has the very convenient feature that it preserves some information from the first LU-decomposition and, as long as the sparsity pattern of the matrix does not change, subsequent LU-decompositions can be done much faster than the first. This package, then, was used in the program `UMDE_DIRECT`. Another package for the same thing is **Y12M**, available from `netlib` [263], and described in [264]. It offers the same features as MA28.

Firstly, the discretisation itself is described. We restrict the discussion to the BI time integration, in order to focus on the spatial discretisations. The program `UMDE_DIRECT` in fact uses BI as the first step, then three-point BDF, which produces second-order accuracy with respect to $\delta T$, this being the rational BDF startup described in Chap. 4, page 70. Take a point away from the boundaries, indices $i$ (for $Z$) and $j$ (for $R$). The discretisation at concentration $C_{i,j}$ of the *pde* (12.17) has three derivative terms, all to be discretised using three-point formulas. The coefficients can be precalculated. For the row along $Z$, there are, for each $0 < Z \leq Z_{max}$, that is, $0 < i \leq n_Z$, four coefficients for the approximations

$$\frac{\partial^2 C}{\partial Z^2} \approx \alpha_{Z1} C_{i-1,j} + \alpha_{Z2} C_{i,j} + \alpha_{Z3} C_{i+1,j} \tag{12.45}$$

(the coefficients can be precomputed using the routine `FORN`, described in Appendix E). These coefficients are independent of $R$ (or $j$), so there are only $3n_Z$ of these. Similarly, there are $3n_R$ coefficients $\alpha_{Rk}, k = 1 \ldots 3$, for the approximations

$$\frac{\partial^2 C}{\partial R^2} \approx \alpha_{R1} C_{i,j-1} + \alpha_{R2} C_{i,j} + \alpha_{R3} C_{i,j+1} \ . \tag{12.46}$$

This leaves the last term. For the moment, assume that we are away from the problem area $R = 0$, and we thus have the simple approximation with the last set of coefficients for the first derivative,

$$\frac{1}{R}\frac{\partial C}{\partial R} \approx \frac{1}{R_j}\left(\beta_{R1}C_{i,j-1} + \beta_{R2}C_{i,j} + \beta_{R3}C_{i,j+1}\right) \; . \tag{12.47}$$

These three formulae work for (almost) the whole field of values that undergo diffusional changes, up to the boundary lines. There is a problem area, as mentioned above, at $R = 0$, where the above approximation cannot be used, due to the singularity. This has been addressed by Crank and Furzeland [40] and again by Gavaghan [44]. The method they used is also described in detail by Smith [265]. It is the following. Expand $(\partial C/\partial R)$ at some small $R$, using Maclaurin's expansion (a special case of Taylor's expansion):

$$\frac{\partial C}{\partial R}(R) = \frac{\partial C}{\partial R}(0) + R\frac{\partial^2 C}{\partial R^2}(0) + \ldots \tag{12.48}$$

and, from boundary conditions (12.18), $\frac{\partial C}{\partial R}(0) = 0$, and letting $R \to 0$, we obtain

$$\frac{1}{R}\frac{\partial C}{\partial R}(0) \approx \frac{\partial^2 C}{\partial R^2}(0) \; . \tag{12.49}$$

Thus, we can simply add this term to the existing one, and the *pde* on the axis becomes

$$\frac{\partial C}{\partial T} = \left(2\frac{\partial^2 C}{\partial R^2} + \frac{\partial^2 C}{\partial Z^2}\right) \tag{12.50}$$

for which a discretisation already exist (12.45) and (12.46). The only (small) problem is that in this case, $j = 0$, and the discretisation thus refers to nonexistent points with index $-1$. This is easily overcome, again using the boundary condition $\frac{\partial C}{\partial R}(0) = 0$, which means that $C_{i,-1} = C_{i,1}$ and thus

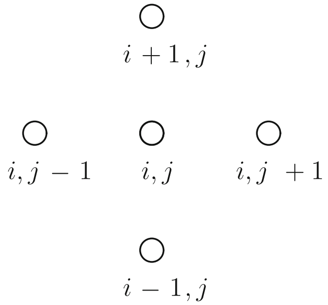
$$\frac{\partial^2 C}{\partial R^2}(R = 0) \approx \alpha_{R2}C_{i,0} + (\alpha_{R1} + \alpha_{R3})C_{i,1} \; . \tag{12.51}$$

In fact, the three $\alpha$ values are all the same, because the three points have equal intervals between them, so that all $\alpha = 1/R_1^2$.

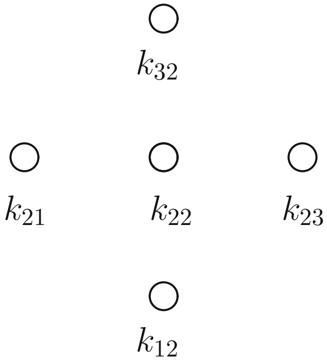
Another special area is the insulating plane outside the disk, defined by $Z = 0, R > 1$. Here, the boundary condition is usually given as in the set (12.18), zero gradient with respect to $Z$. This is expressed as a three-point first derivative, as

$$\beta_1 C_{0,j} + \beta_2 C_{1,j} + \beta_3 C_{2,j} = 0 \; . \tag{12.52}$$

To make the discretisation process more visual, consider any position $(i, j)$ in the grid. There are a total of 5 points around and including this central point, and each of them has its own $k$-value, mapped from its indices. It looks like this:



Each of the positions maps into a $k$ value, the index of the element in the unknowns vector to be solved for. These are denoted, corresponding to the above scheme, by



Thus, the central point at $(i, j)$ has map-index $k_{22}$. For its discretisation, there will be entries in row $k_{22}$, at column positions at all three $k$ values. The horizontal row (referring to the mapping formula (12.44)) are all contiguous $k$ values, while the vertical row maps into column values that are $n_R + 1$ apart from each other. So only $k_{22}$ need be computed by the mapping function KMAP, the others can then be simply set. For example,

$$\begin{aligned} k_{21} &= k_{22} - 1 \\ k_{23} &= k_{22} + 1 \\ k_{12} &= k_{22} - n_R - 1 \\ k_{32} &= k_{22} + n_R + 1 \end{aligned} \tag{12.53}$$

etc.

We can now put the discretisations together, still focussing only on the right-hand side of (12.17). Adding up the individual discretisations (12.45)–(12.47), we can express the total (semi)discretisation as

$$\frac{dC}{dT} = a_{21}C_{i,j-1} + a_{22}C_{i,j} + a_{23}C_{i,j+1} + a_{12}C_{i-1,j} + a_{32}C_{i+1,j} \ , \qquad (12.54)$$

where the $a$-coefficients are put together as follows:

$$\begin{aligned} a_{21} &= \alpha_{R1} + \beta_1/R_{j-1} \\ a_{22} &= \alpha_{R2} + \beta_2/R_i + \alpha_{Z2} \\ a_{23} &= \alpha_{R3} + \beta_3/R_{j+1} \\ a_{12} &= \alpha_{Z1} \\ a_{32} &= \alpha_{Z3} \end{aligned} \qquad (12.55)$$

with the $\alpha$ coefficients already defined above. For the axis where $R = j = 0$, the term in $C_{i,j-1}$ drops out and the coefficients follow from (12.51).

This leaves the boundary conditions. The equation for the insulating plane is given in (12.52), producing three matrix entries at each point $R > 1$. The remaining points are now those on the disk surface itself, and the points outside the diffusion space. On the disk surface, for the Cottrell-like simulation, we have zero concentrations, and at the outer points all concentrations are unity. These produce single row entries in the matrix.

We must now attend to the time integration, that is, the choice of discretisation of the left-hand side of (12.17). In the example program `UMDE_DIRECT` it was decided to use a second-order time integration, and not CN. This suggested either extrapolation or BDF, both described in Chap. 8, Sect. 8.5.2. Second-order extrapolation has the disadvantage of requiring two half-sized steps in time as well as one whole step, which means two different coefficient matrices and thus two LU-decompositions. This takes up more computer memory and cpu time. BDF, on the other hand, is done in a single step and requires, for its second-order variant, only a second concentration array, which is much smaller than the coefficient matrix. BDF requires a start-up strategy, and the rational start, [266] taking a single BI step, followed by second-order BDF is a natural choice. It produces second-order accuracy with respect to $\delta T$, and is stable.

For the BI step, the left-hand side of (12.54) is

$$\frac{dC'}{dT} \approx \frac{C'_{i,j} - C_{i,j}}{\delta T} \qquad (12.56)$$

which, when putting unknowns and knowns on opposite sides, makes (12.54)

$$a_{21}C'_{i,j-1} + (a_{22} - \frac{1}{\delta T})C'_{i,j} + a_{23}C'_{i,j+1} + a_{12}C'_{i-1,j} + a_{32}C'_{i+1,j} \\ = -\frac{C_{i,j}}{\delta T} \,. \tag{12.57}$$

For second-order BDF,

$$\frac{dC}{dT} \approx \frac{{}'C_{i,j} - 4C_{i,j} + 3C'_{i,j}}{2\delta T} \tag{12.58}$$

and the final lumped equation is

$$a_{21}C'_{i,j-1} + (a_{22} - \frac{3}{2\delta T})C'_{i,j} + a_{23}C'_{i,j+1} + a_{12}C'_{i-1,j} + a_{32}C'_{i+1,j} \\ = \frac{{}'C_{i,j}}{2\delta T} - \frac{2C_{i,j}}{\delta T} \,. \tag{12.59}$$

There is just one small difference on both sides in the two forms and in the program, there is need for only a small `IF`-statement split, to handle both in the same code stretch.

A small problem is the current integration. This is defined, in dimensionless terms in (12.19). The integration must be performed on an unevenly spaced set of $R$ values. Gavaghan has examined current integration. From some numerical experiments on an evenly spaced grid [94] he concluded that the trapezium method is the most suitable. Simpson integration did not produce better results, because the edge anomaly produced large errors that dominate the current integration process. In his 1998 paper [44], describing a UMDE simulation on an unevenly spaced grid, Gavaghan then again opted for the trapezium method, stating that it is the most economical, and for the three-point formula for evaluating the flux densities $\partial C/\partial Z$ at each $R$ and $Z = 0$. In our own numerical experiments, a Simpson-like algorithm was worked out for unevenly spaced points, and it works somewhat better than the trapezium method if the concentrations to be integrated are sufficiently accurate in the first place. Routines `U_TRAP` and `U_SIMP`, that integrate on an uneven grid, are presented in Appendix E.

The resulting code can be seen in the example program `UMDE_DIRECT` which the interested reader might study. It turns out that a rather large number of points are required, and very small intervals near the disk and around the disk edge. To get good accuracy from this program, for example, it was necessary to use the maximum possible number of points on our computer, $n_A = n_Z = 90, n_R = 180$, and set the smallest $\delta Z = 10^{-6}$ and equally, the smallest $\delta R = 10^{-5}$, similar to the values chosen by Gavaghan [45]. The expansion parameters $\gamma$ are then computed automatically to fit these numbers. With these parameters, the program produces currents with better than 0.1 % accuracy for $T < 5$, even using three-point approximations, and ran in less than 2 s, even on the somewhat outdated computer used.

### 12.3.3 Discretisation in the Mapped Space

The other major approach to simulating a UME is to map the 2D-space into another space using conformal mapping. Consider Fig. 12.4, showing two sets of equiconcentration lines for a potential jump at a UMDE at $T = 1$. The lines range from zero (along the electrode in A, or along $\Gamma = 0$ for B) in steps of 0.1 up to 0.9. Note the crowding of the lines in A around the electrode edge, but the rather even spread of the lines in B, and the fact that they are almost parallel with the base line. The mapping function used in this case is that of Amatore and Fosset [216], about which more will be said below. The figure indicates that simulation in the transformed space should be better than in the original $(R, Z)$ space, and this is indeed true, especially for larger values of $T$. However, consider now Fig. 12.5, the same situation but at $T = 0.01$. Here, the lines in normal space are, over most of the $R$-range, almost parallel with the base line except for a small area around the disk edge, while in transformed space, they are no longer parallel and somewhat crowded at the right-hand end (which corresponds to the region near the central axis, $R = 0$). This suggests that direct discretisation in $(R, Z)$ space might be favourable. In practice, however [46, 214], conformal mapping is superior over the whole time range.

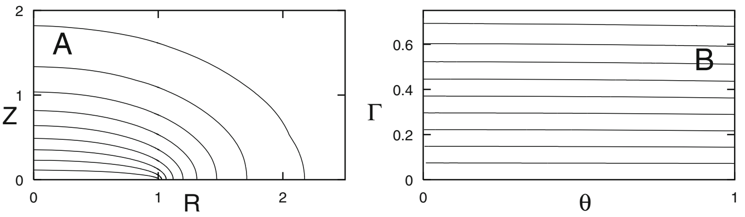


**Fig. 12.4** Equiconcentration lines at $T = 1$ at a UMDE in normal and AF-transformed space

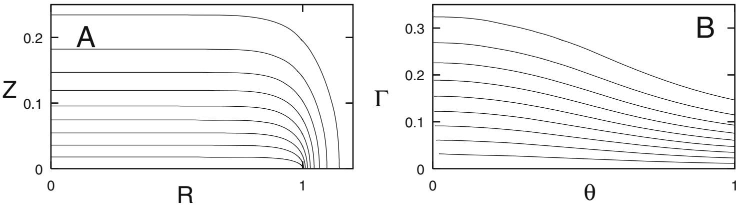


**Fig. 12.5** Equiconcentration lines at T = 0.01 at a UMDE in normal and AF-transformed space

There is thus good argument for using transformation for 2D simulations. Several transformation formulas have been suggested and used. The first to do this was Newman in 1966 [267], for his study of the resistance to current at a flat disk; he was followed by Saito [31], in his derivation of the steady-state currents at an ultramicrodisk and -band electrode. Saito used a conformal mapping function for the band electrode, that was later used again by Michael et al. [148], applying it to the UMDE (see below). Amatore describes several conformal mappings in his review of UMEs [13]. The properties of conformal maps can be found in such publications as [268], and Amatore has a good discussion of the technique [13].

#### 12.3.3.1 Some Transformations

Here, the five mapping functions for the disk electrode are presented, as well as the form that the diffusion equation for the disk electrode takes in the mapped spaces. We assume that the cylindrical coordinates, time and concentrations have all been normalised by the disk radius as in (12.14). The way to achieve these transformations is described in Appendix B.

Michael et al. [148] used the mapping function used earlier by Saito [31], transforming to elliptic coordinates [269],

$$\begin{aligned} R &= \cos\theta \cosh\Gamma \ , \\ Z &= \sin\theta \sinh\Gamma \ . \end{aligned} \tag{12.60}$$

This will be called **MWA** here. The transformation formula is in fact very old, and can be found in such sources as the texts by Sneddon [270], Morse and Feshbach [269] and Tranter [36]. The transformation is also used for band electrodes, with $R$ replaced by $X$, measured as a distance from the centre of the band, across the band [13]. It results, in the case of the disk or band electrode, in a new diffusion equation, whose form is deferred to a later place, below.

We wish to simulate by discretising on an equally spaced grid in the transformed space, and this grid should place points optimally in the original $(R, Z)$ space. That is, they should be closely spaced near the disk edge, and more widely spaced, the further away form the edge point they are. For illustration, note Fig. 12.6, where a coarse $10 \times 10$ grid is shown in the transformed $(\theta, \Gamma)$ space of the transformation (12.60). The $\Gamma$ end is open. This means that for a given simulation, one must decide on the maximum $\Gamma$ value (see page 286). This grid of points, retransformed by application of (12.60), produces the grid on the left-hand side of the figure. We note that indeed, points are closely spaced around the disk edge, and move apart away from that region. The outer, nearly circular curve corresponds to the maximum $\Gamma$ chosen in this example, 2.5. $\Gamma$ is the parameter that sets distance from the origin, in a slightly complicated way. We note that the regular grid of Fig. 12.6 is reflected in a rather regular spacing of the “radial” lines (angles) and an

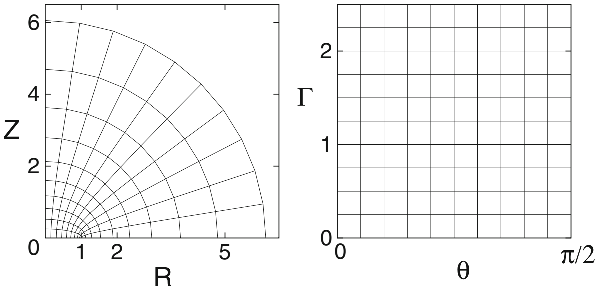


**Fig. 12.6** A 10 × 10 grid in MWA $(\theta, \Gamma)$ space and its equivalent in $(R, Z)$

expanding spacing in the distance of the circle-like lines. All this is favourable, as it will tend to space isoconcentration lines at roughly equal intervals.

The MWA map has what might be regarded as a drawback. We wish to contain the concentration field that varies during the time $T_{max}$ of the simulation, that is, to have a computational domain extending over distances of about 6 sqrt($T_{max}$) from the electrode surface. This translates, upon conformal mapping, to a certain maximum value, $\Gamma_{max}$. How this is calculated is described in Appendix D. The point is that such a calculation must be made, which however also applies to the other transformations, as will be seen below.

The next conformal map to be developed was that of Amatore and Fosset [216], here to be called **AF**:

$$\begin{aligned} R &= (1-\theta^2)^{\frac{1}{2}} / \cos\left(\tfrac{\pi}{2}\Gamma\right) \; , \\ Z &= \theta \tan\left(\tfrac{\pi}{2}\Gamma\right) \end{aligned} \tag{12.61}$$

The symbols are (here) the same as for MWA, except that the range of $\theta$ is from zero to unity, rather than to $\pi/2$. A convenient result of this transformation is that the concentration profile is very simple at steady state for the potential jump system:

$$C(\theta, \Gamma) = \Gamma \; . \tag{12.62}$$

In other words, the profile has contour lines parallel with the base line ($\Gamma = 0$). This should make simulations using this transformation very efficient. However, the above profile holds at steady state only, and when one compares the efficiency of the transformations at shorter times, VB and OAS (see below) are equally efficient, in the sense that they all take about the same amount of computer time to reach a given target accuracy in the calculated current. MWA, AF and MF transformations, however, are somewhat less efficient.

Figure 12.7 shows the equivalences for the AF map. This transformation, in the ranges for $\theta$ and $\Gamma$ shown on the right-hand figure, contains the whole semi-infinite space in $(R, Z)$, so no calculation of a maximum $\Gamma$ is supposedly needed. As explained below on page 286, however, there can be efficiency reasons for

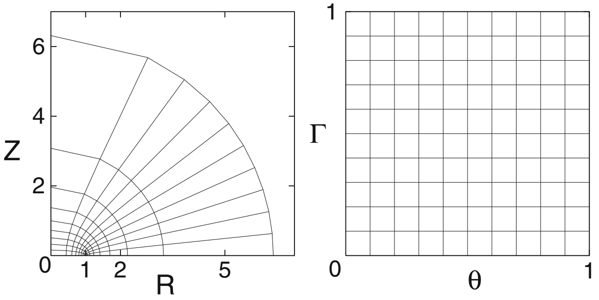


**Fig. 12.7** A 10 × 10 grid in AF $(\theta, \Gamma)$ space and its equivalent in $(R, Z)$

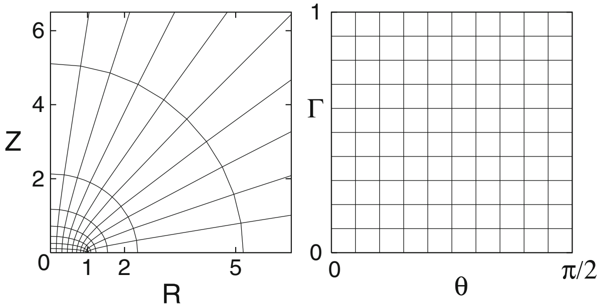


**Fig. 12.8** A 10 × 10 grid in VB $(\theta, \Gamma)$ space and its equivalent in $(R, Z)$

calculating a maximum $\Gamma$ even here. Also note that, contrary to the MWA map, equal intervals in the conformal space produce $\theta$ lines with varying (angular) spacing in $(R, Z)$, being more widely spaced near the disk axis. This is undesirable and might reduce accuracy in the discretisations around that axis. This was confirmed in a recent study comparing all five transformations [214]. A positive point is that the even spacings in the $\Gamma$ direction produces expanding distances from the electrode as we move further out. This is desirable, and better than MWA, where the distances expand to a lesser degree. The problem with the angular spacing was overcome by the fourth transformation, to be mentioned below.

Next, Verbrugge and Baker [271] changed the definition of $\Gamma$ in MWA and arrived at the new equation pair, here referred to as **VB**:

$$
\begin{aligned}
R &= \cos\theta \cosh\left(\frac{\Gamma}{1-\Gamma}\right) \\
Z &= \sin\theta \sinh\left(\frac{\Gamma}{1-\Gamma}\right) .
\end{aligned}
\qquad (12.63)
$$

This transformation produces the equivalence pair in Fig. 12.8. As with AF, the maximum $\Gamma = 1$ completely encloses the semi-infinite diffusion space. The

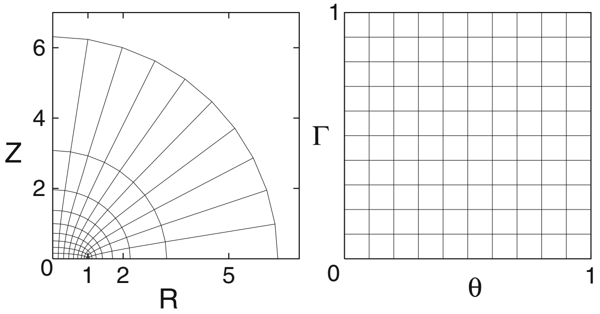


**Fig. 12.9** A 10 × 10 grid in OAS $(\theta, \Gamma)$ space and its equivalent in $(R, Z)$

parameter $\theta$ has the same limits as with MWA. We note an even spacing of the angles with changing $\theta$ and an outwardly expanding spacing for a regular increase in $\Gamma$. So this transformation has both positive features.

Oleinick et al. devised a fourth transformation, here called **OAS**. It is a variant of AF, and is defined as follows:

$$\begin{aligned} R &= \sin\left(\tfrac{\pi}{2}\theta\right) / \cos\left(\tfrac{\pi}{2}\Gamma\right) \\ Z &= \cos\left(\tfrac{\pi}{2}\theta\right) \tan\left(\tfrac{\pi}{2}\Gamma\right) \ . \end{aligned} \tag{12.64}$$

This is the AF map (now only "quasiconformal", as the authors note), with $\theta$ replaced by $\cos(\theta)$. Figure 12.9 shows the result. The change from $\theta$ to $\cos(\theta)$ eliminates the rather uneven spread of angles seen in Fig. 12.7; the angles are now spread more like those for VB, Fig. 12.8. One difference here is that the angle $\theta$ is now zero on the axis, and unity on the insulating plane, the reverse of all three earlier transformations. This transformation appeared to give rather good results, and one expects it to perform somewhat like VB, which has been confirmed [214].

A fifth transformation, found in Morse and Feshbach [269] and here called **MF** was used in an analytical solution by Newman [147, 267], and subsequently in a single electrochemical simulation study by Kottke et al. [187], dealing with a UMDE under an oblate spheroidal polymer drop containing an enzyme. It is defined as

$$\begin{aligned} R &= \sqrt{(1+\Gamma^2)(1-\theta^2)} \\ Z &= \theta\Gamma \ . \end{aligned} \tag{12.65}$$

This rather simple transformation also leads to a pleasingly simple *pde*. The grid equivalence is shown in Fig. 12.10, and we note an uneven distribution of the $\theta$ lines in the $(R, Z)$ plane, indicating that the transformation may not be too promising. MF was convenient in the Kottke et al. study, as the maximum $\Gamma$ value set there matched the surface of the polymer drop, making the simulation easier. However,

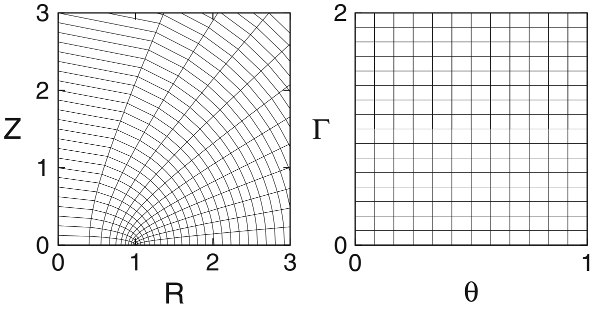


**Fig. 12.10** Grid in MF $(\theta, \Gamma)$ space and its equivalent in $(R, Z)$

applied to the general UMDE problem, the transformation turned out somewhat inefficient [214], as expected from the grid comparison.

#### 12.3.3.2 Inversion of the Transformations

It is sometimes of interest to invert the transformations, computing the pair of $(\theta, \Gamma)$ coordinates from a given $(R, Z)$ pair. All the transformations can be inverted trigonometrically. The inversion for AF has been presented by Svir and Oleinick [272] (with a small typographical problem), and that for OAS by those authors themselves [273]. For some inversions, there are alternative expressions, and there are cases where special formulae must be applied, as is described below.

The inversion equations for the MWA transformation (12.60) and $\Gamma$ are given by

$$\begin{aligned} \Gamma &= \operatorname{arcsinh} \sqrt{\frac{1}{2}\left(R^2 + Z^2 - 1 + \sqrt{(R^2+Z^2-1)^2 + 4Z^2}\right)} \\ \theta &= \arccos\left(\frac{R}{\cosh \Gamma}\right) \end{aligned} \tag{12.66}$$

for both $R > 0$ and $Z > 0$. There is an expression for $\theta$ independent of that for $\Gamma$, but the one given here is preferable, because the other has two possible solutions, and it is not immediately obvious which one is correct. The one given here comes directly from the first of the transformation pair (12.60). Special cases are

$$\begin{aligned} R > 0,\ Z = 0 : &\quad \theta = 0; \quad\ \ \Gamma = \operatorname{arccosh} R \\ R = 0,\ Z > 0 : &\quad \theta = \pi/2;\ \Gamma = \operatorname{arcsinh} Z \\ R = 0,\ Z = 0 : &\quad \theta = \pi/2;\ \Gamma = 0\,. \end{aligned} \tag{12.67}$$

Inversion of VB (12.63) is almost the same as that for MWA, except that the expression for $\Gamma$ in (12.66) is now an expression for $\Gamma'$, to be converted to the present $\Gamma$ by

$$\Gamma = \Gamma'/(1+\Gamma')\,. \tag{12.68}$$

The AF (12.61) general inversion is

$$\begin{aligned}\theta &= \sqrt{\frac{1}{2}\left(1-R^2-Z^2+\sqrt{(R^2+Z^2-1)^2+4Z^2}\right)}\\ \Gamma &= \frac{2}{\pi}\arctan\left(\frac{Z}{\theta}\right)\end{aligned} \tag{12.69}$$

with the special cases

$$\begin{aligned}&R>1,\ Z=0: \quad \theta=0;\ \Gamma=\frac{2}{\pi}\arccos\left(\frac{1}{R}\right)\\ &R=0,\ Z>0: \quad \theta=1;\ \Gamma=\frac{2}{\pi}\arctan Z\\ &R=0,\ Z=0: \quad \theta=1;\ \Gamma=0\,.\end{aligned} \tag{12.70}$$

For OAS (12.64) we have

$$\begin{aligned}\theta &= \frac{2}{\pi}\arccos\left(\sqrt{\frac{1}{2}\left(1-R^2-Z^2+\sqrt{(R^2+Z^2-1)^2+4Z^2}\right)}\right)\\ \Gamma &= \frac{2}{\pi}\arctan\left(\frac{Z}{\cos(\frac{\pi}{2}\theta)}\right)\,.\end{aligned} \tag{12.71}$$

Alternative inversion expressions are given by Oleinick et al. [273], equivalent to those above.

The special cases for the OAS inversion are

$$\begin{aligned}&R>1,\ Z=0: \quad \theta=1;\ \Gamma=\frac{2}{\pi}\arccos\left(\frac{1}{R}\right)\\ &R=0,\ Z>0: \quad \theta=0;\ \Gamma=\frac{2}{\pi}\arctan Z\\ &R=0,\ Z=0: \quad \theta=0;\ \Gamma=0\,.\end{aligned} \tag{12.72}$$

The inversion of the MF transformation is

$$\begin{aligned}\Gamma &= \sqrt{\frac{1}{2}\left(R^2+Z^2+\sqrt{(R^2+Z^2)^2+4Z^2}\right)}\\ \theta &= Z/\Gamma\,.\end{aligned} \tag{12.73}$$

#### 12.3.3.3 The Diffusion Equation in the Mapped Spaces

The transformations lead to a change in the diffusion equation and boundary conditions, in terms of the new variables. How the transformed *pde*s are found is described in Appendix B. The general form of the diffusion equation, for all cases, is

$$\frac{\partial C}{\partial T} = \frac{1}{F}\left(a_\theta \frac{\partial^2 C}{\partial \theta^2} + b_\theta \frac{\partial C}{\partial \theta} + a_\Gamma \frac{\partial^2 C}{\partial \Gamma^2} + b_\Gamma \frac{\partial C}{\partial \Gamma}\right) \tag{12.74}$$

with Tables 12.1 and 12.2 showing the parameters (they are not all constants) for MWA, AF and VB, and for OAS and MF, respectively.

Consider again Fig. 12.8 for the VB grid. It shares with the MWA and AF mappings the following simple equivalences in $(R, Z)$ space. The base line of the mapped (right-hand) grid corresponds to the electrode itself, going inward from the edge ($\theta = 0$) to the disk centre ($\theta = \pi/2$). The left-hand edge ($\theta = 0$) traces the insulating plane away from the disk edge; while the right-hand edge traces the axis itself. The top line is at maximum distance from the electrode. In the case of AF, VB and OAS, if it lies at $\Gamma = 1$, it corresponds to infinity. In the case of MWA, if suitably chosen, it lies at a distance sufficient for significant diffusional changes to be confined within that limit. More will be said about the limit below.

**Table 12.1** Parameter values for the diffusion equations in the MWA, AF and VB spaces

| Parameter | MWA | AF | VB |
|---|---|---|---|
| $F$ | $\sin^2\theta + \sinh^2\Gamma$ | $\theta^2 + \tan^2\Gamma'$ | $\sin^2\theta + \sinh^2(\frac{\Gamma}{1-\Gamma})$ |
| $a_\theta$ | $1$ | $1-\theta^2$ | $1$ |
| $b_\theta$ | $-\tan\theta$ | $-2\theta$ | $-\tan\theta$ |
| $a_\Gamma$ | $1$ | $\frac{4}{\pi^2}\cos^2\Gamma'$ | $(1-\Gamma)^4$ |
| $b_\Gamma$ | $\tanh\Gamma$ | $0$ | $(1-\Gamma)^2\tanh(\frac{\Gamma}{1-\Gamma}) - 2(1-\Gamma)^3$ |

For better readability, the symbol $\Gamma' \equiv \frac{\pi}{2}\Gamma$ is used

**Table 12.2** Parameter values for the diffusion equations in the OAS and MF spaces

| Parameter | OAS | MF |
|---|---|---|
| $F$ | $\frac{1}{\cos^2\Gamma'}\left(\cos^2\theta' + \sin^2\theta'\sin^2\Gamma'\right)$ | $\theta^2 + \Gamma^2$ |
| $a_\theta$ | $\frac{4}{\pi^2}$ | $1-\theta^2$ |
| $b_\theta$ | $\frac{2}{\pi}\cot\Gamma'$ | $2\theta$ |
| $a_\Gamma$ | $\frac{4}{\pi^2}\cot^2\Gamma'$ | $1+\Gamma^2$ |
| $b_\Gamma$ | $0$ | $2\Gamma$ |

For better readability, the symbols $\theta' \equiv \frac{\pi}{2}\theta$ and $\Gamma' \equiv \frac{\pi}{2}\Gamma$ are used

Equation (12.74) must be accompanied by boundary conditions. These are, generally, now again for the potential jump experiment:

$$\begin{aligned} T = 0,\ \text{all}\ \theta, \Gamma : &\quad C = 1 \\ T > 0, \Gamma = 0 : &\quad C = 0 \\ \Gamma = \Gamma_{max} : &\quad C = 1 \\ \theta = 0, \theta_{max} : &\quad \frac{\partial C}{\partial \theta} = 0\ . \end{aligned} \tag{12.75}$$

The second boundary condition will change appropriately if another experiment than the potential jump is simulated. Here, $\Gamma_{max}$ and $\theta_{max}$ depend on the conformal map used, and on how $\Gamma_{max}$ is chosen (see below). For MWA and VB, $\theta_{max} = \pi/2$, while for AF, it is unity.

#### 12.3.3.4 Current Integration in Conformal Coordinates

The current integration (12.19) depends on the transformation. For the three conformal mappings described above, the new expressions are as follows. For MWA [148] and VB [271],

$$I = \frac{\pi}{2} \int_0^{\pi/2} \left. \frac{\partial C}{\partial \Gamma} \right|_{\Gamma=0} \cos\theta \ \mathrm{d}\theta\ . \tag{12.76}$$

For AF it is [13]

$$I = \int_0^1 \left. \frac{\partial C}{\partial \Gamma} \right|_{\Gamma=0} \mathrm{d}\theta\ ; \tag{12.77}$$

for OAS [273],

$$I = \frac{\pi}{2} \int_0^1 \left. \frac{\partial C}{\partial \Gamma} \right|_{\Gamma=0} \sin(\frac{\pi}{2}\theta)\ \mathrm{d}\theta \tag{12.78}$$

and for MF

$$I = \frac{\pi}{2} \int_0^1 \left. \frac{\partial C}{\partial \Gamma} \right|_{\Gamma=0} \mathrm{d}\theta\ . \tag{12.79}$$

How to obtain the current equation in transformed coordinates is described in Appendix B.

### 12.3.4 Band Electrodes

Apart from the circular UMDE, there is considerable interest in the UMBE, since currents are larger. Here the *pde* is in $(X, Z)$, both dimensions normalised by the half-width of the band, which most conveniently is taken as very small with respect to the length of the band (but see Sect. 12.4.1 for bands of finite length). The dimensionless *pde* is then

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} + \frac{\partial^2 C}{\partial Z^2} \,. \tag{12.80}$$

Taking into account the symmetry of the system and thus applying the above only to the right half of the band geometry, the boundary conditions are

$$\begin{aligned}
&T = 0, \quad \text{all } X, Z: \quad C = 1 \\
&T > 0: \\
&\quad Z = 0,\ \ 0 \le X \le 1: \quad C = 0 \\
&\quad Z \to \infty,\ X \to \infty: \quad C = 1 \\
&\quad X = 0,\ \ Z > 0: \quad \partial C/\partial X = 0 \\
&\quad Z = 0,\ \ X > 1: \quad \partial C/\partial Z = 0
\end{aligned}$$

and the normalised current

$$I(T) = \frac{i(T)}{nFDc^* wL} = 2\int_0^1 \left.\frac{\partial C}{\partial Z}\right|_{Z=0} \mathrm{d}X \,. \tag{12.81}$$

The integration from $X = 0$ and the factor 2 are given by symmetry.

As for the UMDE, this is best transformed, and a formula identical with the Verbrugge–Baker one for the UMDE can be applied, as in (12.63). It leads to the transformed *pde*

$$\frac{\partial C}{\partial T} = \frac{1}{\sin^2\theta + \sinh^2\left(\frac{\Gamma}{1-\Gamma}\right)}\left(\frac{\partial^2 C}{\partial \theta^2} + (1-\Gamma)^4\frac{\partial^2 C}{\partial \Gamma^2} - 2(1-\Gamma)^3\frac{\partial C}{\partial \Gamma}\right) \,. \tag{12.82}$$

with boundary conditions

$$\begin{aligned}
&T = 0, \quad \text{all } \theta, \Gamma: \quad C = 1, \\
&T > 0: \\
&\quad \Gamma = 0, \ 0 \le \theta \le \pi/2: \quad C = 0 \\
&\quad \Gamma = \Gamma_{max}, \ 0 \le \theta \le \pi/2: \quad C = 1 \\
&\quad 0 \le \Gamma \le \Gamma_{max}, \ \theta = 0, \pi/2: \quad \partial C/\partial\theta = 0 \ .
\end{aligned}$$

The normalised current becomes

$$I = 2 \int_0^{\pi/2} \left.\frac{\partial C}{\partial \Gamma}\right|_{\Gamma=0} \mathrm{d}\theta \ . \tag{12.83}$$

Everything else in the simulation closely resembles the treatment for the UMDE.

#### 12.3.4.1 Choice of $\Gamma_{max}$

When simulating the UMDE using one of the transformations, it is often of advantage to know a maximum $\Gamma$ value, $\Gamma_{max}$. In the case of MWA, this is indeed necessary, as for that transformation, $\Gamma$ increases indefinitely with increasing distance from the electrode. All the other transformations have a limiting $\Gamma$ value of unity, corresponding to points at infinity. However, even in these cases, computing time can be saved by restricting the range of $\Gamma$ or, conversely, if using a fixed number of intervals in the $\Gamma$ direction, better resolution can be achieved and accuracy improved by the restriction [214]. It is thus of interest to find these maximum $\Gamma$ values.

How to determine the maximum $\Gamma$ is deferred to Appendix D.

#### 12.3.4.2 Discretisation

Taking one of the conformally mapped grids, such as the VB grid in Fig. 12.8, it now remains to develop the discretisations of the corresponding *pde* (12.74), with the coefficients as in Tables 12.1 and 12.2 or computed using the Fornberg algorithm [274] (see the example subroutine FORN), and boundary conditions, as in the set (12.75). One needs to choose the number of points for the derivative approximations. This has been experimented with [214, 275] and the conclusion was that three-point formulae (that is, three-point in each of the two dimensions) gave slightly better results than using a higher number of points at short times when discretising in $(R, Z)$ coordinates but five- or seven-point approximations were most efficient in transformed coordinates at longer times. The approximations for $m$ points (with $m$ odd), for some variable $u$ referred to index $i$, away from the edges, are

$$\frac{\partial u}{\partial X} \approx \sum_{k=1}^{m} \beta_k u_{i-(m-1)/2-1+k} \tag{12.84}$$

for first derivatives with respect to $X$ and

$$\frac{\partial^2 u}{\partial X^2} \approx \sum_{k=1}^{m} \alpha_k u_{i-(m-1)/2-1+k} \tag{12.85}$$

for second derivatives with respect to $X$. With equal intervals, in principle, the coefficient sets are independent of the index $i$ but if the index is within $(m-1)/2$ of an edge, asymmetrical forms must be used and then they are dependent on the index. When working in transformed coordinates, we find [214] that often as few as 13 intervals along a coordinate are sufficient, so the near-edge problem applies to relatively many index values, and one may as well compute a set of coefficients for all index values. There is a set of each of the two derivatives for both the $\theta$- and $\Gamma$ direction. The coefficients for first and second derivative approximations can be added to produce a set of weights. For example, in the *pde* (12.74), both derivatives with respect to $\theta$ are approximated by a weighted sum of $m$ points, so we can add the coefficients, and divide by $F$ (as defined in Tables 12.1 and 12.2) at the same time; and the same with respect to $\Gamma$. The diffusion equation then becomes, on the right-hand side, the discrete equation at index $i, j$

$$\left.\frac{\partial C}{\partial T}\right|_{i,j} \approx \sum_{k=j_1}^{j_2} w_\theta(j,k) C_{i,k} + \sum_{k=i_1}^{i_2} w_\Gamma(i,k) C_{k,j} \; , \tag{12.86}$$

$w_\theta$ and $w_\Gamma$ being composites of the coefficients and $j_1$ and $j_2$ being the lower and upper limits of the index around $j$, along $\theta$, and $i_1$ and $i_2$ being the lower and upper limits of the index around $i$, along $\Gamma$. These limits depend on where the reference point is; as mentioned, if it is away from the edges, they are equidistant from the reference index and if near an edge, not so. They are conveniently determined by a small subroutine at each reference point on the grid. Such a subroutine, `I1I2`, is described in Appendix E.

The bottom edge of the right-hand grid in Fig. 12.8 corresponds to the electrode, where values either are simply set (the Cottrellian case) or are computed in some other way (for example, for a reversible reaction). The top edge is defined as points with constant, bulk, concentration. Thus, the points to be treated by discretisation are those in between these two lines. As with direct discretisation on the grid in $(R, Z)$, all points are taken as unknowns and mapped into one long array. Let there be $N_\theta$ intervals of length $\delta\theta$ in the $\theta$ direction, so that the points are indexed $0 \le j \le N_\theta$, and similarly a number $N_\Gamma$ intervals of length $\delta\Gamma$, $0 \le i \le N_\Gamma$ along $\Gamma$. The total number of points is then

$$N = (N_\theta + 1)(N_\Gamma + 1) \ . \tag{12.87}$$

Then the concentration point at $(i, j)$ maps into a $k$ given by

$$k = i\,(N_\theta + 1) + j + 1 \ . \tag{12.88}$$

A stencil of $2m-1$ points is involved in the interior of the grid, and fewer at reference points falling on an edge.

Each of the positions maps into a $k$ value, the index of the element in the unknowns vector to be solved for. It seems unnecessary to depict these.

The program UMDE_VB (Appendix E) is an example of using the Verbrugge–Baker transformed grid to simulate the potential jump experiment at a disk electrode. BI is used as the first step, followed by three-point BDF. This works rather well, using $m$-point discretisations for the first and second derivatives. As mentioned, a study attempting to optimise the efficiency of these simulations [214] concluded that five- or seven-point approximations are most efficient. This yields a modest-sized system of discrete equations that can either be solved as a whole-matrix system, ignoring the majority of zeroes, or by treating it as a sparse system and using a subroutine suite such as MA28 [261], available at [262].

#### 12.3.4.3 Unequal Intervals in the Mapped Space

Although in general, equal intervals are used in the mapped $(\theta, \Gamma)$ space, with all the advantages this entails, there can be cases where unequal intervals, especially in the $\Gamma$ direction, can be of advantage. For example, in a study of an enzyme system within a hemispherical polymer drop containing the enzyme, covering a UMDE [182], there were derivative boundary conditions at the polymer drop surface and in order to render these accurately, it was desirable to limit the expansion of the grid inherent in the $\Gamma$ direction. For this reason, a mild expansion going backward from the maximum $\Gamma$ level was employed. There may be other cases where this would improve the results of a simulation.

### *12.3.5 A Remark on the Boundary Conditions*

In the set of boundary conditions above, (12.3) and (12.75), there are zero-gradient conditions. In the case of the grid in $(R, Z)$, there are two of them, at $R = 0$ and at $(Z = 0, R > 1)$. Although both are given by Crank and Furzeland [40], these authors do not in fact use both of them *as* boundary conditions; the one at $R = 0$ is used as a symmetry argument, in order to develop the form of the term that might become singular, arriving at (12.50). Thus, they allow diffusion along as well as away from the axis. Similarly, one might allow diffusion in the radial direction along the insulating plane, as well as normal to it, leaving out the (zero) normal $\partial C/\partial Z$ term, and using symmetry to construct the special form of $\partial^2 C/\partial Z^2$ there. This latter discretisation seems not to be used by anyone. What is used is simply the no-flux condition, discretised suitably. One might suspect that this makes use of less information than is available, and thus render the solution less accurate.

In the case of the mapped grid, we also have two zero gradients $\partial C/\partial\theta$, at both the left- and right-hand edges of, for example, Fig. 12.8. This is the way it is usually done [216, 271, 275]. At both edges, however, it is also possible—and might make more sense—to invoke the diffusion equation, taking symmetry into account, and leaving out the first derivative terms $\partial C/\partial\theta$ there. The program `UMDE_VB` was modified with this in mind, allowing diffusion in both directions. Some numerical experiments showed that the results were almost the same as for using the boundary condition, and convergence to an accurate value with increasing grid intervals was no faster. Therefore the choice remains a personal one.

## 12.4 Three-Dimensional Simulations

The laboratory space or electrochemical cell has of course the familiar three spatial dimensions. However, as we have seen in the previous sections, the vast majority of electrodes and their environment can be adequately described by reducing the geometry to one or two dimensions. The reduction is possible because of symmetry or zero gradients in one or two space directions. There are only a few cases where the computational space has to comprise three dimensions. This is fortunate because the computational expense increases steeply with each spatial dimension.

Three-dimensional simulations of single electrodes have been done by explicit FD [96, 276], ADI [277] and FEM (COMSOL Multiphysics®) [278]. A three dimensional dual ultramicrodisk system was simulated by Fulian and Fisher using the boundary element method (BEM) [279]. BEM [87, 176, 177, 280] and FEM (using COMSOL Multiphysics®) [281] were employed to simulate complex three dimensional tip-substrate geometries in SECM. Arrays with a small number of electrodes were simulated using ADI [282] and commercial FEM packages Fluxexpert® [283, 284] and COMSOL Multiphysics® [285, 286].

### *12.4.1 Square and Rectangular UMEs*

In Sects. 12.1.2 and 12.3.4 equations both in dimensional and dimensionless form, as well as transformed coordinates for the UMBE, are given. It was assumed that the length of the band is much greater than the width, $l \gg w$. Under these circumstances the geometric space is reduced to an $(x, z)$ plane by assuming that the concentration gradient pointing into $y$ direction, i.e. along the length of the band electrode, is zero. This implies that the concentration profile calculated for the $(x, z)$ plane is uniform along the length of the band electrode. The flux to the UMBE is then calculated from the surface flux in the $(x, z)$ plane multiplied by the length, $l$. For chronoamperometric current transients at UMBEs analytical series solutions [12, 51, 53, 61, 62, 287, 288] as well as a precise numerical simulation [161] have been published. The question arises: what will happen if the length of band electrode is shorter, so that the condition $l \gg w$ is no longer fullfilled? For the UMBE the *edge effect* is only considered along the length of the electrode, not at both ends. If $l$ becomes shorter and approaches the range of $w$, the edge effect along both ends of the electrode becomes more important for the overall flux to the electrode. While the current will decrease with decreasing length, because the area of the electrode becomes smaller, the current density will increase because of the additional edge effect along both ends of the electrode. Therefore, from a certain value of the ratio $l/w$ on, the two-dimensional approach for modelling the behaviour of such electrodes will introduce systematic errors. To examine this behaviour three spatial dimensions have to be taken into account. We will show how to simulate a potential step to the diffusion limit, i.e. the Cottrell experiment, at a rectangular and square electrode. This was done first by Strutwolf [277], followed by Cutress et al. [276] and Woodvine et al. [278]. For the notation, the term UMBE is applied only when $l \gg w$, otherwise if $l$ approaches $w$ the term rectangular UME is used, with the limit $l = w$ for a square UME.

The location of a planar rectangular electrode of width $w$ and length $l$ in a Cartesian coordinate system is shown in Fig. 12.11a. The origin of the coordinate system is at the centre of the electrode. For normalisation we make use of the same equations as for the UMBE.

$$C = c/c^* \tag{12.89a}$$

$$X = 2x/w \tag{12.89b}$$

$$Y = 2y/w \tag{12.89c}$$

$$Z = 2z/w \tag{12.89d}$$

$$L = 2l/w \tag{12.89e}$$

$$T = 4Dt/w^2\ , \tag{12.89f}$$

The right-hand side of Eq. (12.89f) contains a factor 4 which does not appear in the definition of the dimensionless time for the UMDE, Eq. (12.16). This is because space is normalized by the half width of the electrode, $w/2$, while for the UMDE the radius $a$ is used. The diffusion equation in three dimensions is then

$$\frac{\partial^2 C}{\partial T^2} = \frac{\partial^2 C}{\partial X^2} + \frac{\partial^2 C}{\partial Y^2} + \frac{\partial^2 C}{\partial Z^2}\ . \tag{12.90}$$

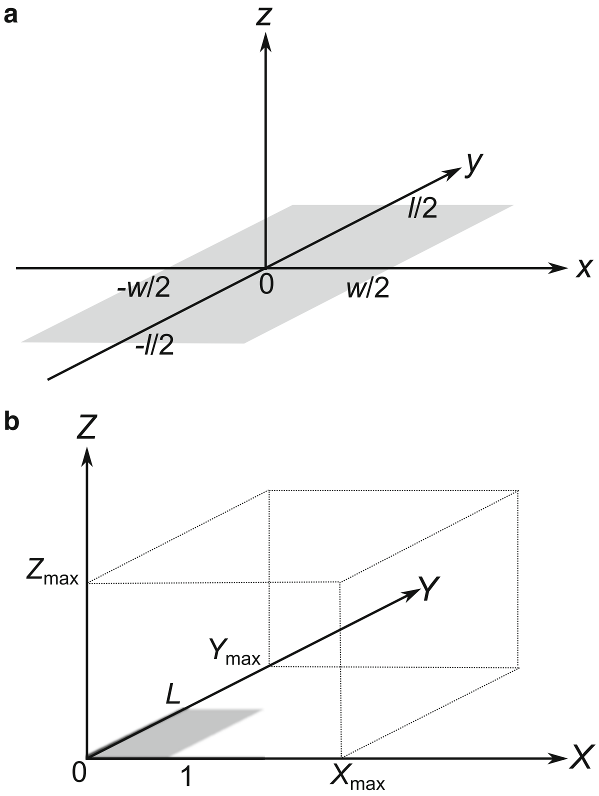


**Fig. 12.11** (**a**) Cartesian coordinates for a rectangular electrode of length $l$ and width $w$. (**b**) Computational domain for the rectangular electrode in normalised space coordinates.

Figure 12.11b presents the rectangular electrode in dimensionless space. Because of the symmetry of the rectangular UME it is sufficient to consider only a quarter of the rectangle. $X_{max}$, $Y_{max}$ and $Z_{max}$ mark the maximum values of the space variables taken into account for the computation. Analogous to Eqs. (12.37) and (12.38) for the UMDE, the limits are defined as

$$X_{\max} = 1 + 6\sqrt{T_{\max}} \tag{12.91}$$

$$Y_{\max} = L + 6\sqrt{T_{\max}} \tag{12.92}$$

$$Z_{\max} = 6\sqrt{T_{\max}}\ . \tag{12.93}$$

$T_{max}$ is the duration of the experiment in dimensionless units, see Eq. (12.89f). The parameter $L$ is the dimensionless half-length of the band and is given by the ratio $l/w$. The boundary conditions for a simple potential step experiment to the diffusion limit involving one electroactive species have been given before for an ultramicroelectrode (2.42) and for the UMDE (12.18). These conditions are adapted to the geometry shown in Fig. 12.11b.

$$T = 0 \qquad \text{for all}\, X, Y, Z, \qquad C = 1, \tag{12.94a}$$

$$T > 0$$

$$0 \le X \le 1, \quad 0 \le Y \le L, \quad Z = 0, \quad C = 0 \tag{12.94b}$$

$$1 < X < X_{\max}, \; 0 \le Y < Y_{\max}, \quad Z = 0, \quad \partial C/\partial Z = 0 \tag{12.94c}$$

$$0 \le X \le 1, \quad L < Y < Y_{\max}, \quad Z = 0, \quad \partial C/\partial Z = 0 \tag{12.94d}$$

$$0 \le X < X_{\max}, \; Y = 0, \quad 0 < Z < Z_{\max}, \; \partial C/\partial Y = 0 \tag{12.94e}$$

$$X = 0, \quad 0 < Y < Y_{\max}, \quad 0 < Z < Z_{\max}, \; \partial C/\partial X = 0 \tag{12.94f}$$

$$X = X_{\max}, \quad 0 \le Y < Y_{\max}, \quad 0 \le Z < Z_{\max}, \; C = 1 \tag{12.94g}$$

$$0 \le X \le X_{\max}, \; Y = Y_{\max}, \quad 0 \le Z \le Z_{\max}, \; C = 1 \tag{12.94h}$$

$$0 \le X \le X_{\max}, \; 0 \le Y \le Y_{\max}, \quad Z = Z_{\max}, \quad C = 1. \tag{12.94i}$$

Equation (12.94a) is the initial condition—at the beginning of the computation (or experiment) the concentration in the whole domain is equal to the bulk concentration. Equation (12.94b) is the boundary condition at the electrode, where the electroactive species reacts instantly, so that the concentration is zero. Equations (12.94c) and (12.94d) are the no-flux conditions at the insulating plane surrounding the electrode. Equations (12.94e) and (12.94f) are symmetry conditions along the *XZ* and *YZ* plane, respectively. Finally, the last three equations are the far field conditions where *C* remains at its normalised bulk value.

For the current calculation, integration has to be performed along the $x$ direction ($0 \le x \le w/2$) and along the $y$ direction ($0 \le y \le l/2$). This is expressed by the double integral

$$i = 4n\mathcal{F}D \int_0^{w/2} \int_0^{l/2} \left.\frac{\partial c}{\partial z}\right|_{z=0} \mathrm{d}y\,\mathrm{d}x \,. \tag{12.95}$$

Replacing $c$, $x$, $y$ and $z$ as well as the integration limits by Eqs. (12.89a)–(12.89d) results in

$$i = 2n\mathcal{F}Dc^* wG \tag{12.96}$$

where

$$G = \int_0^1 \int_0^L \left.\frac{\partial C}{\partial Z}\right|_{Z=0} \mathrm{d}Y\,\mathrm{d}X \tag{12.97}$$

is the normalized current.

#### 12.4.1.1 Discretisation

Some geometries, for example the UMDE, lend themselves to conformal mapping which increases the computational efficiency considerably by *folding away* the singularity at the electrode edge. However, it can be shown that conformal mapping in Euclidean space of dimension greater than two is much more rigid and restricted [289]. Therefore direct discretisation in the $X, Y, Z$ space is applied, similar to what is described for the UMDE in Sect. 12.3.2. The number of grid points for each space direction is $n_X + 1$, $n_Y + 1$ and $n_Z + 1$. For accuracy of the simulation, small intervals between grid points in the vicinity of the electrode edges are required. As for the UMDE an unequal grid is generated using the Fortran function EE_FAC (Appendix E, described in Sect. 7.2). The intervals expand from the electrode edges along $X = 1$ in both $X$ directions, towards $X = 0$ and $X = X_{max}$ and at $Y = L$ towards $Y = 0$ and $Y_{max}$. This is illustrated in Fig. 12.12a for an electrode of half-length $L = 2$. For the $Z$ direction the expansion of the intervals starts at $Z = 0$ towards $Z_{max}$, see Fig. 12.12b. Besides the target point, up to which expansion should take place, the Fortran function EE_FAC needs information about the start interval, that is, the smallest, and the number of points for the space over which expansion will take place. Let $n_{aX} + 1$ be the number of grid points in the $X$ direction on the electrode. Then $X_0, X_1, ..X_{naX}$ are points along the electrode width, where $X_{naX} = 1$ is the point at the electrode edge and $X_{naX} - X_{naX-1}$ is the smallest interval (an input parameter) which will expand towards $X_0$ with an expansion factor $\gamma$ calculated by the function EE_FAC. Starting from $X_{naX}$ towards $X_{max}$ there are then $n_X - n_{aX}$ points, where $X_{nX} = X_{max}$. The smallest interval (an input parameter) is $X_{naX+1} - X_{naX}$ which will expand until the point $X_{nZ} = X_{max}$ is reached. For the $Z$ direction expansion is only away from the plane of the electrode, starting at $Z_0 = 0$, the same as for direct discretisation of the UMDE. The function EE_FAC only needs the number of space points, $n_Z$, the maximum value $Z_{max}$ and the smallest interval, $Z_1 - Z_0$.

For the numerical solution one could in principle now use the same strategy as for the UMDE in the example program UMDE_DIRECT. Assuming the number of grid points in each of the three space dimensions, including the boundary points, is $N$. Then there are $N^3$ equations to be solved simultaneously to compute the new concentration vector $\mathbf{C}'$ implicitly. The non-zero entries of the $N^3 \times N^3$ matrix of coefficients will have a banded structure. Most of the entries will be zero and a sparse matrix solver such as **Y12M** or **MA28** (the latter was used for the UMDE, see the UMDE_DIRECT program) can be employed to decrease the required computer memory and to increase the speed of matrix computations. However, in our experience, for a time march to simulate transients, the number of space points (or unknown concentrations) should not be bigger than ca. 40,000 to keep the computation time within a reasonable limit. For the two-dimensional case this implied a maximum value of $N \approx 200$ while for three dimensions, $N$ will drop to about 35, which is not enough for an accurate simulation. For this reason an alternating direction implicit (ADI) scheme, mentioned in Sect. 12.3.1, is applied for the numerical solution of Eq. (12.90). Instead of solving simultaneously

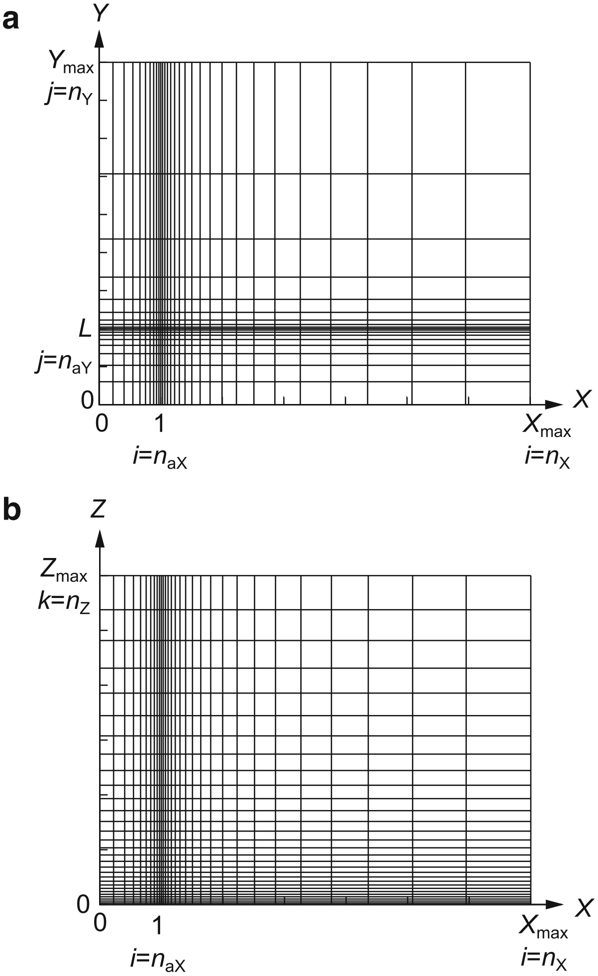


**Fig. 12.12** The unequally spaced grid for the three-dimensional simulation of a rectangular electrode with $L = 2$. (**a**) *XY* plane; (**b**) *XZ* plane

in one (time) step for all concentrations, the ADI method splits each time step into substeps in which concentrations only in one space dimension are solved for. The UMDE simulation is two dimensional and therefore two substeps are necessary. For three dimensions, each time step is divided into three substeps, in each of them the concentrations in one space direction are solved for implicitly. There are different choices of ADI schemes for three dimensions [290–295], which differ in error order and computational costs. The ADI scheme by Douglas and Gunn [293] is of second-order with respect to space and time intervals, $O(\delta T^2, \delta X^2, \delta Y^2, \delta Z^2)$, though it was noted that unequally spaced grid points degrades the order of the spatial approximations, as discussed on page 55. The Douglas–Gunn ADI scheme for three dimensions is written as [293]

$$\left(1 - \frac{\delta T}{2}\delta_X^2\right) C^{\ddagger} = \left(1 + \frac{\delta T}{2}\delta_X^2 + \delta T \delta_Y^2 + \delta T \delta_Z^2\right) C \tag{12.98a}$$

$$\left(1 - \frac{\delta T}{2}\delta_Y^2\right) C^{\ddagger\ddagger} = C^{\ddagger} - \frac{\delta T}{2}\delta_Y^2 C \tag{12.98b}$$

$$\left(1 - \frac{\delta T}{2}\delta_Z^2\right) C' = C^{\ddagger\ddagger} - \frac{\delta T}{2}\delta_Z^2 C \tag{12.98c}$$

where $C$ is the known concentration, $C^{\ddagger}$, $C^{\ddagger\ddagger}$ are interim unknowns and $\delta_X^2$, $\delta_Y^2$ and $\delta_Z^2$ are difference operators for the second derivative with respect to space. A three-point central difference formula is applied, so that $\delta_X^2$ is defined as

$$\frac{\partial^2 C}{\partial X^2} \approx \delta_X^2 C_{i,j,k} = \alpha_{X1i} C_{i-1,j,k} + \alpha_{X2i} C_{i,j,k} + \alpha_{X3i} C_{i+1,j,k}, \tag{12.99}$$

and similarly for $\delta_Y^2$ and $\delta_Z^2$. The indices $i, j, k$ characterise a location of a grid point at $X_i$, $Y_j$ and $Z_k$. The $\alpha$ coefficients for the unequal grid are calculated by the subroutine `FORN`, see Appendix E. Equation (12.98) expand to

$$\begin{aligned} -\frac{1}{2}\left(\alpha_{X1i} C^{\ddagger}_{i-1,j,k} + \left(\alpha_{X2i} - \frac{2}{\delta T}\right) C^{\ddagger}_{i,j,k} + \alpha_{X3i} C^{\ddagger}_{i+1,j,k}\right) \\ = \frac{C_{i,j,k}}{\delta T} + \frac{1}{2}\left(\alpha_{X1i} C_{i-1,j,k} + \alpha_{X2i} C_{i,j,k} + \alpha_{X3i} C_{i+1,j,k}\right) \\ + \left(\alpha_{Y1j} C_{i,j-1,k} + \alpha_{Y2j} C_{i,j,k} + \alpha_{Y3j} C_{i,j+1,k}\right) \\ + \left(\alpha_{Z1k} C_{i,j,k-1} + \alpha_{Z2k} C_{i,j,k} + \alpha_{Z3k} C_{i,j,k+1}\right), \end{aligned} \tag{12.100a}$$

$$\begin{aligned} -\frac{1}{2}\left(\alpha_{Y1j} C^{\ddagger\ddagger}_{1,j-1,k} + \left(\alpha_{Y2j} - \frac{2}{\delta T}\right) C^{\ddagger\ddagger}_{i,j,k} + \alpha_{Y3j} C^{\ddagger\ddagger}_{i,j+1,k}\right) \\ = \frac{C^{\ddagger}_{i,j,k}}{\delta T} - \frac{1}{2}\left(\alpha_{Y1j} C_{i,j-1,k} + \alpha_{Y2j} C_{i,j,k} + \alpha_{Y3j} C_{i,j+1,k}\right), \end{aligned} \tag{12.100b}$$

$$\begin{aligned} -\frac{1}{2}\left(\alpha_{Z1k} C'_{i,j,k-1} + \left(\alpha_{Z2k} - \frac{2}{\delta T}\right) C'_{i,j,k} + \alpha_{Z3k} C'_{i,j,k+1}\right) \\ = \frac{C^{\ddagger\ddagger}_{i,j,k}}{\delta T} - \frac{1}{2}\left(\alpha_{Z1k} C_{i,j,k-1} + \alpha_{Z2k} C_{i,j,k} + \alpha_{Z3k} C_{i,j,k+1}\right) \ . \end{aligned} \tag{12.100c}$$

In Eq. (12.98a) the concentrations in $X_i$ direction are treated implicitly, using known concentrations $C_{i,j,k}$. In a second step, (12.98b), the tridiagonal system of equations is solved implicitly along $Y_j$ ($0 < j < n_Y$) with the known (explicit) concentrations $C^{\ddagger}_{i,j,k}$ and $C_{i,j,k}$ on the right-hand side. In a third step, (12.98c), new concentrations $C'_{i,j,k}$ running along $Z_k$ $(0 < k < n_Z)$ are computed using known values of $C^{\ddagger\ddagger}$ and $C$. The coefficients on the left-hand side of (12.100) together with the coefficients from the discretised boundary conditions form tridiagonal matrices and the systems are solved for the unknown concentrations using the Thomas algorithm [296], as described in Sect. 8.3.

To calculate the current at the rectangular electrode, Eq. (12.97) has to be solved numerically using new concentrations $C'$. This can be done by repeated application of the trapezoidal rule in two dimensions

$$G \approx \frac{1}{4} \sum_{j=0}^{n_{aY}-1} \sum_{i=0}^{n_{aX}-1} \delta Y_j \delta X_i \left( g_{i,j} + g_{i+1,j} + g_{i,j+1} + g_{i+1,j+1} \right) \tag{12.101}$$

where $n_{aY}$ and $n_{aX}$ are the maximum numbers of grid points on the electrode along $X$ and $Y$ and $\delta X_i = X_{i+1} - X_i$ and $\delta Y_j = Y_{j+1} - Y_j$. The local surface gradients $(Z = 0)$ at the $(i, j)$th grid point are calculated, for example, by a four-point forward difference approximation

$$g_{i,j} \approx \beta_0 C'_{i,j,0} + \beta_1 C'_{i,j,1} + \beta_2 C'_{i,j,2} + \beta_3 C'_{i,j,3} \tag{12.102}$$

with $\beta_k$ being the coefficients of the forward finite difference approximation of the first derivative on the unequally spaced grid in $Z$ direction, computed using the routine FORN, see Appendix E.

### 12.4.2 The Grid

Suitable grid parameters can be evaluated by comparing current $G/L$ of the three-dimensional simulation for high values of the dimensionless length of the ultramicroband, $L$, with the values given by the formula of Aoki et al. [12, 288] for the (two-dimensional) UMBE. The following grid parameters were found to reproduce the analytical result to within an error of 0.8 % [277]: $n_X = n_Y = n_Z = 80$, $n_{aX} = n_{aY} = 40$. The first grid intervals left and right of the electrode edges and the first interval in $Z$ direction was set to $10^{-4}$.

The Douglas–Gunn scheme is unconditionally stable. But, like Crank–Nicolson or the Peaceman–Rachford ADI scheme [100], it is not $L$-stable [297], see also Sect. 15.3. For the potential step experiment, where an abrupt change in the concentration at the electrode surface occurs at zero time, the lack of $L$-stability introduces oscillations of the computed concentrations. The oscillations will be stronger the higher the value of the model diffusion coefficient $\lambda = \delta T/h^2$ is, $h$ being a spatial interval. These oscillations might be damped quickly after a few time steps, but for ultramicroelectrode simulations the model diffusion coefficient will be quite large, because of the small grid intervals around the electrode edges. Damping of error oscillations occurring in the simulation of potential step experiments has been investigated [124, 125, 298]. Strutwolf [277] applied the method of Pearson [126] where the first time step of length $\delta T$ is divided into $m$ equal subintervals. Using $\delta T = 0.01$ and a smallest space interval of $10^{-5}$ gives a maximum $\lambda$ of about $10^6$. Therefore a high value of $m \approx 5000$ has to be applied for damping oscillations [277].

Examples of transient simulations for different values of $L$, starting with a square shaped electrode ($L = 1$) are shown in [277] and [276]. The percentage deviation between transients calculated for two-dimensional diffusion from the analytical equation of Aoki et al. [12, 288] and three-dimensional simulation with decreasing value of $L$ has been presented [277]. An Aoki type equation where the coefficients were evaluated by fitting to three-dimensional numerical simulations to describe transients at square and rectangular electrodes has been published [276].

## 12.5 Ultramicroelectrode Arrays

The signal (current or potential) from a single UME is very small compared to a conventional microelectrode. An obvious idea for increasing the small signal is to run multiple UMEs in parallel in an electrochemical experiment. This led to the development of ultramicroelectrode arrays, which are defined as an ensemble of two or more UMEs on one device. The main motivation behind the development of ultramicroelectrode arrays was to achieve higher electrochemical signals while maintaining the advantageous properties of UMEs [41]. The progress in micro- and nanomachining over the last decades [299, 300] enabled the development of well-defined and reproducible UME arrays incorporating electrodes of different shapes and dimensions and various electrode-to-electrode separations; see [18, 301–312] for a few examples. Ultramicro- and nanoelectrode arrays are nowadays commercially available. Besides electroanalysis, ultramicroelectrode arrays find application in neurobiology and electrophysiology to stimulate and record neuron activities [313–315]. Partially blocked electrodes can be regarded as electrode arrays, where disk shaped openings in a blocking layer form active sites for electrochemical reactions. Earlier theoretical and experimental works were concerned with partially blocked stationary [316] and RDEs [317–320]. Scheller et al. prepared such blocked electrodes with disk shaped openings of radii down to 6 μm by a photoresist technique [318]. A modification of the Levich equation (i.e. the equation for the RDE limiting current) was proposed taking into account nonlinear diffusion to the active sites. The theory made use of what is called the *diffusion domain approach* (see later in this section). The behaviour of microscopic active sites of partially blocked electrodes has been numerically simulated [71–73, 316, 321–323].

A UME array operating as a working electrode in an electrochemical experiment can be run in two principal modes. (a) In parallel mode all electrodes are connected by a single common wire to a potentiostat, so that identical currents or potentials are applied to all electrodes in the array; (b) in addressable mode, electrodes or group of electrodes can be addressed individually by a bi-potentiostat or multiplexer. A typical example is the generator-collector mode, where a species is electrochemically generated on one set of electrodes while the generated species is then consumed at the second set of electrodes (the collector electrodes). A controlled potential or current is applied to the generator, and a controlled potential at the collector, different to that at the generator. Examples of theory and numerical simulation of generator-collector systems are given in [112, 217, 234, 258, 324–336]. More generally addressable UME arrays have been developed [286, 337–340], which are of special interest for monitoring binding events that occur between the members of a molecular library and a biological receptor [341].

UMEs are usually arranged in a regular pattern to form an array. Multitudes of parallel UMBEs are present in an IDA electrode. Disks or square shaped UMEs can be arranged in a square or hexagonal lattice. Randomly distributed UMDEs—if they are not too closely packed—can be modelled by estimating the nearest-neighbour distribution [342] and statistical considerations, as has been done by Scharifker [343] and later by Compton et al. [344–348] and others [77, 349]. The next section looks into the simulation approach for regular arrays of UMDEs.

### 12.5.1 Regular Arrays of UMDEs

As mentioned before, a motivation for operating a UMDE array in parallel mode is to multiply the electrochemical response of a single UMDE in the array, for example to decrease the detection limit of an analyte. However, the behaviour of the array response depends on the design of the array and four parameters are crucial for this: (1) the size of the UMDEs, given by their radius $a$; (2) the centre-to-centre distance $l$ between the nearest neighbour electrodes; (3) the total number $N_e$ of UMEs in the array; (4) the duration of the experiment or observation time. For the regular array we assume that both $l$ and $a$ have the same values for all electrodes. Points (2) and (3) also determine the overall size of the array. In practice this size is often restricted by experimental conditions, e.g. the dimension of the electrochemical cell.

If the duration of the experiment is such that the extension of the diffusion layers $\delta$ evolving around the single electrodes of the array is smaller than the half-distances to the closest neighbours $l/2$, then the problem is reduced to that of a single UMDE. If $\delta > l/2$, then the diffusion zones will overlap and vicinal UMDEs deplete the same region in the solution, leading to a decrease of flux to an individual electrode in comparison with an isolated electrode, where radial diffusion can evolve unhindered. This effect is taken into account by the concept of active and inactive regions or the diffusion domain approach, which was first used for partially blocked RDEs by Scheller et al. [318] and later applied by Matsuda and co-workers for modelling chronoamperometry and chronopotentiometry [350], linear and cyclic sweep voltammetry [351] and impedance measurements [352] at hexagonal arrays of co-planar UMDEs. The same concept was then applied to the calculation of CVs for hexagonal and square ultramicroelectrode arrays [316].

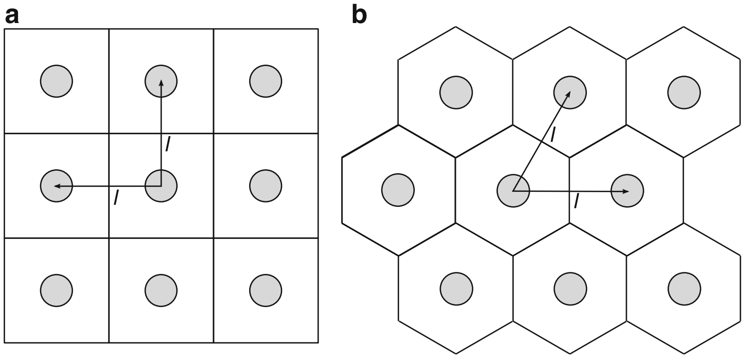


**Fig. 12.13** Top view of sections of regular arrays of UMDEs arranged in (**a**) a square and (**b**) hexagonal lattice

The diffusion domain approach allows the mapping of the three-dimensional multi-electrode array geometry onto a two-dimensional unit cell involving just one active area [350]. To illustrate how this is achieved, consider Fig. 12.13a which shows the top view of a section of an array of UMDEs in a square packed geometry, forming a square lattice. The active areas, i.e. the UMDEs, are each at the centre of a square with a side length $l$, which is identical with the centre-to-centre separation between adjacent electrodes. It is assumed that the array consists of a large number of electrodes, all at the same potential or current, so that each electrode is represented by an inner square cell and that the different environment of the outer electrodes, i.e. the UMDEs along the edges of the array, can be neglected. Because all square unit cells are identical, the boundary condition at all four sides of the square is a zero flux condition. To simulate a single square unit cell, three space dimensions $(x, y, z)$ have to be considered, with the $z$ axis being perpendicular to the centre of the disk electrode. The $z$ axis falls together with the $C_4$ symmetry axis. The computational domain would be a rectangular prism with a certain height $z_{max}$ and a square base area of side length $l$. Figure 12.13b shows UMDEs arranged in a hexagonal pattern and the base side of the unit cell has the shape of a regular hexagon with the UMDE at its centre. Here the $z$-axis falls together with a $C_6$ symmetry axis. Though the domain is reduced to a unit cell with just one electrode, the problem is again three dimensional in space and therefore computationally demanding. It is desirable to have the $z$ axis assigned to a $C_\infty$ symmetry axis because that results in a cylinder and as we have seen from UMDE simulation (Sect. 12.3) the problem is reduced to two space dimensions, $r$ and $z$. The reduction of dimensions is achieved by the diffusion domain approach.

#### 12.5.1.1 The Diffusion Domain Approach

Figure 12.14a shows a circle concentric with the UMDE and with the same surface area as the square base of the unit cell. Since the area of the square is $l^2$, the radius $d$ of the circle is found from $\pi d^2 = l^2$ or

$$d = \frac{l}{\sqrt{\pi}} = 0.5642\, l\,. \tag{12.103}$$

For a hexagonal configuration, illustrated in Fig. 12.14b, the equivalence of the areas of the hexagon and the circle concentric with the UMDE results in

$$d = \frac{\sqrt[4]{3}}{\sqrt{2\pi}} l = 0.5250 l\,. \tag{12.104}$$

Using these approximations the simulation task becomes essentially the same as already described for the simulation of a UMDE undisturbed by other electrodes. The *pde* to solve is given by (12.2). Normalisation is done as already described for a single UMDE , resulting in Eq. (12.17) for potential step or (12.27) for LSV conditions. The difference is that the UMDE is now in effect embedded in a walled area of base radius $d$, i.e. the radius of the diffusion domain, so that $r_{max} = d$ and the boundary condition changes from

$$r \to \infty : \quad c = c^* \tag{12.105}$$

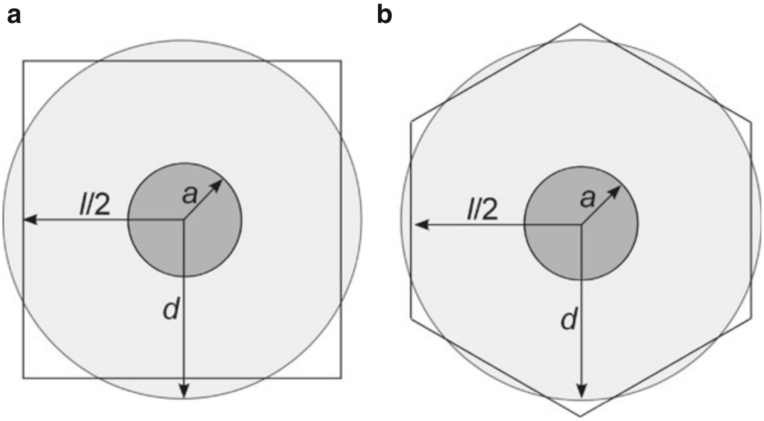


**Fig. 12.14** The diffusion domain approach to (**a**) a square lattice and (**b**) a hexagonal lattice of UMDEs of radius $a$ and a centre-to-centre separation of $l$

(see Eq. (12.3)) to

$$r = d : \quad \frac{\partial c}{\partial r} = 0 \,. \tag{12.106}$$

In dimensionless form this reads

$$R = R_d : \quad \frac{\partial C}{\partial R} = 0 \,, \tag{12.107}$$

where $R_d = d/a$. The constant (bulk) concentration condition is thus replaced by a no-flux or symmetry condition.

Equation (12.19) is applied to calculate the current at a single electrode. To calculate the current of the array, (12.19) is multiplied by $N_e$, the number of electrodes of the array.

The validity of the diffusion zone approach, even in the case where the electrode separation becomes very small has been shown by three-dimensional Brownian motion simulation [349]. However, the approximate nature and limits of the diffusion zone approach must be pointed out. It is obvious that the representation of a square or hexagon by a circle (or vice versa) is an approximation. The closer the number in front of $l$ in Eqs. (12.103) and (12.104) is to 0.5, the better the approximation. The diffusion domain approach is better suited to a hexagonal array than for a square array, though the difference is rather small [349]. A more serious limitation arises from the assumption that the UMDE which is actually simulated is representative of all the electrodes of the array. The simulated electrode is an *inner* electrode surrounded by six (hexagonal lattice) or four (square lattice) nearest neighbour electrodes, unlike the electrodes at the edge where the number of nearest neighbour electrodes is smaller. Therefore the diffusion zone approach requires that the number of inner electrodes is much higher than the number of edge electrodes. In other words, the larger $N_e$, and the larger the area of the array, the better the approximation. In a situation where $N_e$ is small, but the array size is large, so that $l > \delta$, the diffusion fields around the UMDEs do not disturb each other and the simulation is equivalent to an isolated disk electrode. In such a case, the far field boundary condition, if constant bulk concentration or no-flux condition, should have no influence, because the concentration at this limit is not affected by diffusion. However, if the array dimension is in the μm range, then the diffusion zone approach might introduce considerable errors, because then the UMDEs are not equivalent, assuming a small value of $N_e$. Indeed the overall array might then behave similarly to an ultramicroelectrode in an electrochemical experiment. And as for an ultramicroelectrode, where the current varies across the electrode surface—highest close to the edge, because of the *edge effect*—the flux to UMDEs along the edge of the array will be different from the flux towards UMDEs closer to the centre of the array. The ultramicroelectrode-like behaviour of μm sized arrays has been observed experimentally. For example, a 40 × 40 $\mu\text{m}^2$ sized array with 390 disks with $a = 0.115\,\mu\text{m}$ and $l = 2.475\,\mu\text{m}$ showed no current peak on the time scale of a slow voltammetric scan experiment [310]. Using the diffusion zone approach in this situation predicts overlapping diffusion zones and a high contribution of linear diffusion and consequently a current peak. For a µm sized array, a simulation model has to take all electrodes into account and has to consider three space dimensions, which is expensive from a computational point of view and the number of publications is small. Kolev et al. simulated the three-dimensional diffusion process at an array consisting of nine rectangular UMEs, using an ADI algorithm adapted to three space dimensions [282] and a mixed uniform/nonuniform grid. Three-dimensional simulations for UMDE arrays were performed using commercial FEM software packages Flux-Expert® [283, 284] and COMSOL Multiphysics® [285, 286, 353]. The latter software was also employed in three-dimensional (diffusion) simulation of voltammograms and SECM approach curves of nanotip arrays (modelled as nanodisk arrays) [354].

2D simulation of UMDE arrays using the diffusion zone approach has been done by finite differences [344, 346, 355–359] and FEM [283, 360]. By fitting simulated transients Shoup and Szabo derived an equation for the current at a hexagonal UMDE array as a function of time and coverage. This equation reproduces their simulations (using the hopscotch algorithm) with an error of less than 2 % for a coverage of $0.368 < \theta < 0.841$ ($\theta = 1 - (a/d)^2$) [356]. The diffusion zone approach was also applied in FEM simulations of ion transfer across liquid/liquid interfaces confined in ultramicropores of regular solid-state membranes [361–364].

#### 12.5.1.2 An Example

Consider an array of size $2 \times 2\,\text{mm}^2$, so that the diffusion zone approach can be applied. The UMDEs in the array have a radius of 10 µm and are arranged on a square packed lattice. The centre-to-centre distance $l$ determines how many electrodes fit on the array. Table 12.3 lists five different array designs together with their parameters. The current response towards an LSV experiment should be simulated for the different arrays. A simple redox reaction

$$A + e^- \rightleftharpoons B \tag{12.108}$$

with complete reversibility of the electron transfer at a given potential $E$ occurs at the electrode. Normalisation of space and concentration is done as given by Eq. (12.14).

**Table 12.3** Five designs of a 2 mm × 2 mm array of disk electrodes with radius of 10 µm in a square packed geometry

| Array | $l$ (µm) | $N_e$ | $d$ (µm) | $R_d$ |
|---|---|---|---|---|
| $A_1$ | 500 | 25 | 282.1 | 28.21 |
| $A_2$ | 200 | 121 | 112.8 | 11.28 |
| $A_3$ | 100 | 441 | 56.4 | 5.64 |
| $A_4$ | 50 | 1681 | 28.2 | 2.82 |
| $A_5$ | 25 | 6561 | 14.1 | 1.41 |

Time is made dimensionless following Eq. (12.16) and more specific for the LSV experiment by Eq. (12.26)

$$\begin{aligned} C &= \frac{c}{c*} \\ R &= \frac{r}{a} \\ Z &= \frac{z}{a} \\ T &= \frac{n\mathcal{F}v}{\mathcal{RT}}t\,. \end{aligned} \tag{12.109}$$

The *pde* to be solved is given by Eq. (12.27), where $P$ is defined by (12.28). $P^2$ is proportional to the potential scan rate $v$ and can be regarded as a dimensionless equivalent to the scan rate. For a given electrode radius $a$ a high $P$ implies a high voltage scan rate. The dimensionless boundary conditions for the LSV experiment are

$$T = 0 \qquad \text{all } R, Z: \qquad C = 1 \tag{12.110a}$$

$$T = 0: p = p_1 \tag{12.110b}$$

$$T > 0$$

$$p = p_1 - T \tag{12.110c}$$

$$0 \leq R \leq 1, \qquad Z = 0: \qquad C = \exp(p)/(1 + \exp(p)) \tag{12.110d}$$

$$1 < R < R_d, \qquad Z = 0: \qquad \partial C/\partial Z = 0 \tag{12.110e}$$

$$R = R_d, \qquad 0 < Z < Z_{max}: \partial C/\partial R = 0 \tag{12.110f}$$

$$0 \leq R \leq R_d, \qquad Z = Z_{max}: \qquad C = 1 \tag{12.110g}$$

$$R = 0, \qquad 0 < Z < Z_{max}: \partial C/\partial R = 0 \tag{12.110h}$$

with $p_1$ being the dimensionless start potential. The Dirichlet boundary condition $C = 1$ for $R \to \infty$ is replaced by Eq. (12.110f), a no-flux condition at $R = R_d$, where $R_d$ is the dimensionless radius $d/a$ of the equivalent circle, see Fig. 12.14a. Boundary condition (12.110d) expresses the Nernst equation (2.32), where the dimensionless potential $p$ is given by Eq. (2.29), assuming $E^0 = 0\ V$. For a scan towards negative potentials with $p_1$ as the starting potential, $p$ at time $T$ is given by $p_1 - T$. Using the potential range as an input parameter, the starting potential is $p_{range}/2$.

Furthermore identical diffusion coefficients for the reduced and oxidised form B and A are assumed, so that $C_B = 1 - C_A$ for all $Z,\ R$. Therefore the problem can be simplified by using just one concentration, $C$. The value of $Z_{max}$ in (12.110) depends on the dimensional potential range and on the parameter $P$ and is given by (12.41). The dimensionless current is calculated from Eq. (12.19).

For the simulation direct discretisation in $(R, Z)$ space is applied, as described previously for the UMDE. It is assumed that the reader is familiar with the approach introduced in Sect. 12.3.2. The generation of the grid with high grid point density close to the electrode surface and at the electrode edge is accomplished with the help of the Fortran function `EE_FAC`. The grid consists of $N = (n_Z + 1) \times (n_R + 1)$ grid points $Z_i$ and $R_j$ with $i = 0, 1, \dots n_Z$ and $j = 0, 1, \dots n_R$. $R_{nR} = R_d$ is the dimensionless radius calculated from (12.103). The diffusion equation (12.17) was discretised by-nine point stencils, with five-point central difference formulas for the first and second derivatives with respect to $R$ and the second derivative with respect to $Z$. The coefficients were calculated by the Fortran subroutine `FORN`. For the points next to the boundaries asymmetric five-point formulas are created with the help of the subroutine `I1I2`, described in Appendix E, to avoid points lying outside the computational domain. For the array there is a no-flux condition along the boundary of the diffusion zone, $R = R_d$, $0 < Z < Z_{max}$, Eq. (12.110f). Again, the one-sided five-point formula $u'_5(5)$ was applied for discretisation. Asymmetric formulas $u'_5(5)$ and $u''_4(5)$ are also used for the discretisation of the $R$ derivatives of the diffusion equation for the grid points next to $R_d, j = n_R - 1$. The code has been implemented in the example program `UMDE_ARRAY` in Appendix E.

Simulations were performed for the arrays listed in Table 12.3, using $v = 10\,\mathrm{mV\,s^{-1}}$ and a diffusion coefficient of $D = 9 \times 10^{-10}\,\mathrm{m^2\,s^{-1}}$, so that $P = 0.2078$. For an isolated UMDE such a $P$ value will result in a sigmoidal shaped voltammogram with a normalised limiting current of $G = 1$. The potential range is $p_{range} = 36$, equivalent to 0.92 V. $Z_{max}$ was calculated from Eq. (12.110f) giving $Z_{max} = 173$. The number of grid points in the $Z$ direction were $n_Z = 100$. In the $R$ direction $n_R$ was set to 70, with $n_a = 30$ points along the electrode, resulting in a total of $101 \times 71$ grid points. The smallest intervals in both space directions were set to $\delta Z = \delta R = 10^{-5}$. Starting at $Z = 0$, the intervals expand towards $Z_{max}$ with an expanding factor $\gamma$, calculated by the Fortran subroutine `EE_FAC`. In the $R$ direction, the smallest interval is located at the electrode edge around $R = 1$, from which expansion takes place towards $R = 0$ and towards $R_{max}$.

Figure 12.15 shows the voltammograms for the different arrays. In Fig. 12.15a the responses of single electrodes of the arrays are presented. For the largest distance between the electrodes, $A_1$, a sigmoidal shape with a limiting current of unity is observed, indicating negligible or no interaction of diffusion zones in the limiting region, a situation characterized by a spherical diffusion field, as outlined in Fig. 12.16a. The current decay at lower potentials resulting in the appearance of a current peak for array $A_2$ indicates the influence of linear diffusion, a consequence of overlapping diffusion zones. A sketch of this is shown in Fig. 12.16b. Diffusion is purely linear after the total overlap of the diffusion zones, as shown in Fig. 12.16c. The LSV of the array will then behave as a microelectrode with area 4 mm$^2$, i.e. the size of the array. The design with the smallest electrode separation, $A_5$ might show the diffusion behaviour shown in Fig. 12.16c. The peak current for a micro electrode can be taken from the Randles–Ševčík function (see Sect. 9.8) [365, 366]

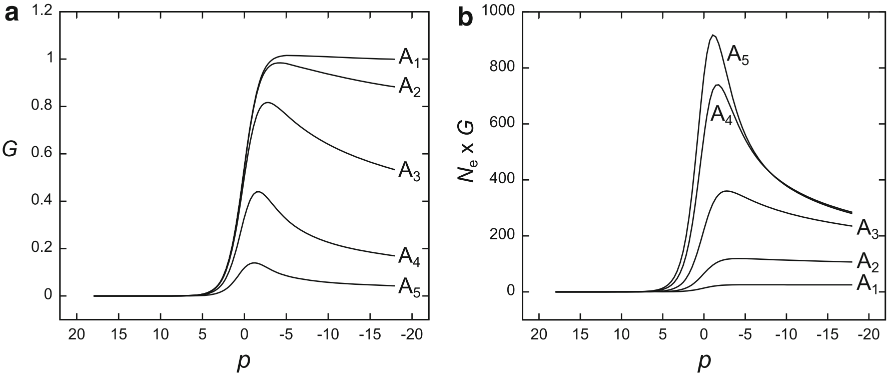


**Fig. 12.15** Simulated linear scan voltammograms for the arrays listed in Table 12.3. (**a**) Response of a single electrode within the array; (**b**) response of the array consisting of $N_e$ electrodes

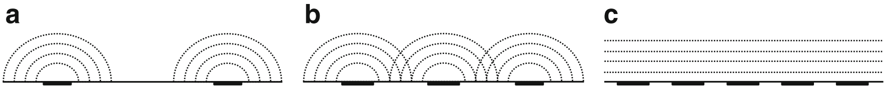


**Fig. 12.16** Sketch of equiconcentration lines showing the diffusional behaviour of UMDE arrays. (**a**) Non-interacting diffusion zones; (**b**) partially overlapping diffusion zones; (**c**) strong overlapping, resulting in purely linear diffusion

$$i_p = 0.4463 \left( \frac{\mathcal{F}^3}{\mathcal{RT}} \right)^{1/2} n^{3/2} A c^* (Dv)^{1/2}. \tag{12.111}$$

Using $A = 4\,\mathrm{mm}^2$, $v = 10\,\mathrm{mV\,s}^{-1}$, $D = 9 \times 10^{-10}\,\mathrm{m}^2\,\mathrm{s}^{-1}$, $c^* = 0.1\,\mathrm{mol\,m}^{-3}$, the peak current of the ultramicroelectrode is $3.2 \times 10^{-4}$ A. This value has to be compared with the current peak for array $A_5$ which is $G_p = 0.14$ for a single electrode (Fig. 12.15a) or 918 for the whole array (Fig. 12.15b). The latter $G_p$ value has to be multiplied by Eq. (12.5) to obtain the current in physical dimensions. The result is a peak current of $3.2 \times 10^{-4}$ A, indicating the absence of radial diffusion. The active area of array $A_5$ is only ca. 50 % of that of the equivalent $\mathrm{mm}^2$ sized micro electrode. Because the current in both cases is the same, the current density at the array electrodes will be higher than that of the micro electrode and the capacity is expected to be lower. Ohmic and kinetic polarization will increase and in case of a kinetically controlled electrode reaction, an apparent rate constant will depend on surface coverage of the array, $k_a^0 = k^0 \Theta$ [316, 351, 357], where the surface coverage $\Theta$ is the ratio of electroactive area to the total area of the array. For array $A_5$ the surface coverage is $\Theta = N_e \pi a^2 / A^2 \approx 0.5$. Because in the example full reversibility of the electron transfer is assumed by boundary condition (12.110d) this has no effect.

A special case is diffusion to a μm sized electrode array, where electrodes radii are in the nm range, with total overlapping diffusion zones. In this case, the current response will exhibit features of a UME, e.g. a steady state limiting current of the same value as a UME with the same area as the array. This has been observed experimentally [367].

With decreasing distances between the electrodes of the array, transport to the individual electrodes decreases and so does the current, as shown in Fig. 12.15a. However, with smaller centre-to-centre distances the number of electrodes $N_e$ on the array increases and the overall current is the highest for the array with the smallest $l$, $A_5$, c.f. Fig. 12.15b. Only array $A_1$ exhibits typical UMDE behaviour, which might be desirable in some experiments. All this has to be taken into account when designing an electrode array.

### 12.5.2 Arrays of UMBEs

Arrays of multiple UMBEs parallel to each other have been widely used in electroanalytical chemistry [368]. Fabrication methods include photolithography [369], chemical and thermal vapor deposition [370, 371] and screen printing [372]. The references serve as examples for the above-mentioned method and are not complete by any means.

Figure 12.17a presents a section of a UMBE array in the form of a two terminal *IDA* electrode. The length and width of the band and the width of the gap between the electrodes are denoted by $l$, $w$ and $g$. The IDA consists of two sets of UMBEs that mesh with each other. The two sets can be connected in parallel to a potentiostat, so that the same potential or current is applied to all electrode fingers and consequently the same electrochemical reaction occurs at all UMBEs, as schematically presented in Fig. 12.17b.

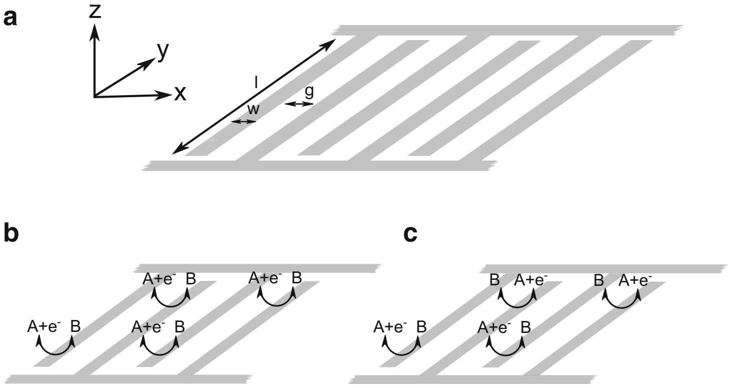


**Fig. 12.17** Arrays of ultramicroband electrodes. (**a**) An interdigitated array electrode (IDA); (**b**) an IDA operated in parallel mode; (**c**) IDA operated in generator-collector mode

The two sets of ultramicrobands can also be addressed differently by using a bipotentiostat to enable a generator-collector mode. One set of electrodes is set to a potential (or current) where a species is generated at the electrode, e.g. by reduction, and the other set of electrodes is set to a potential (or current) where the generated species is consumed, e.g. by oxidation. The generator-collector mode is illustrated in Fig. 12.17c.

One of the first publications on electroanalytical application of IDAs described a photolithography process to produce arrays consisting of 40 ultramicroband pairs with $w = 3.5\,\mu\text{m}$, $l$ between 1 and 2 mm and $g = 2.5\,\mu\text{m}$ [369]. An early numerical solution for the steady state current at generator-collector IDA electrodes using Schwarz–Christoffel transformation was presented by Aoki et al. [373]. The three-dimensional computational domain for simulating an array of UMBEs can be considerably simplified if four conditions are fulfilled: (a) the width of the band electrodes is much smaller than the length (the same condition as for the single band simulations introduced in Sect. 12.3.4); (b) all electrodes in the array are of the same width and length; (c) the number of bands is high; (d) the overall array size is on a micro scale that is, in the mm range). Conditions (c) and (d) are mutually dependent and arise for the same reason that justifies the diffusion domain approach for arrays of UMDEs: given a large number of band electrodes, the different diffusion fields established at the outermost electrodes compared to the inner ones can be neglected and the diffusion field at an inner electrode is representative for all electrodes. However, this is only the case if the overall array size is in the larger micro range, condition (d). If, for example, $w$ and $g$ are in the nano range, the expansion of the array in the $x$ direction, $N_e(w + g) - g$, can be in the μm range, even if the number of bands, $N_e$, is high. On the time scale of an electrochemical experiment, typically in the range of seconds for a voltammetric scan, the IDA might behave like a microelectrode, that is the bands are prone to different flux densities and cannot be treated equally. The case where condition (a) is not satisfied and $l$ approaches $w$ was discussed in Sect. 12.4.1. If the conditions (a)–(d) are fullfilled, diffusion is reduced to two dimensions, and the *pde* to solve is the same as for a single UMBE, (12.33). The computational domain for the simulation of an ultramicroband array in parallel and generator-collector mode is shown on the left side of Fig. 12.18. The simulation space for the parallel-operated IDA is similar to the one for a single UMBE (see Sect. 12.3.4), except along the boundary line $x = (w + g)/2,\ 0 < z < z_{max}$. The symmetry of the diffusion field along the line midway between the electrodes is described by a no-flux condition,

$$t > 0,\ x = (w + g/2),\ 0 < z < z_{max},\ \partial c/\partial x = 0\ . \tag{12.112}$$

In both the parallel and generator-collector mode, the space variables are made dimensionless by dividing by the half width of the electrodes, $w/2$.

$$X = 2x/w \tag{12.113a}$$

$$Z = 2z/w. \tag{12.113b}$$

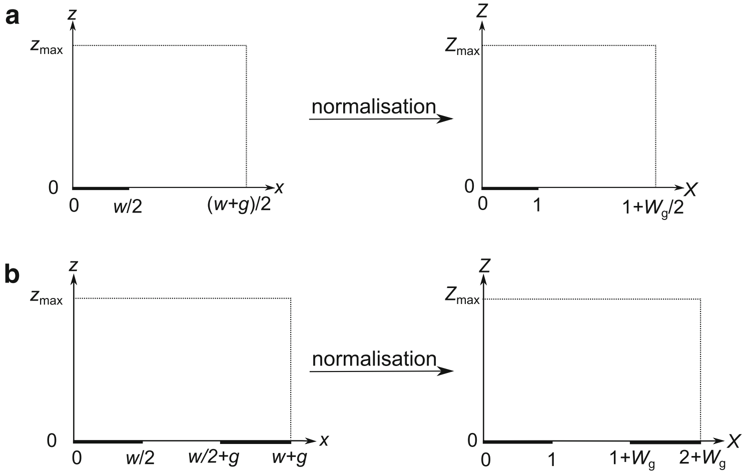


**Fig. 12.18** Computational domain for an ultramicroband IDA in parallel (**a**) and generator-collector mode (**b**). The domains are presented in physical (*left*) and, after normalisation by multiplication with $2/w$, in dimensionless space (*right side*)

The normalised gap width is

$$W_g = 2g/w. \tag{12.114}$$

The figure on the right of Fig. 12.18a presents the computational domain in dimensionless space for the IDAs in parallel mode. The symmetry condition for the parallel-operated UMBE array, (12.112), becomes

$$T > 0,\ X = 1 + W_g/2,\ 0 < Z < Z_{max},\ \ \partial C/\partial X = 0, \tag{12.115}$$

where the concentration $C$ is made dimensionless by dividing with a reference (bulk) concentration. $T$ denotes the dimensionless time, as given by Eq. (12.89f) for a potential step experiment. The remaining boundary conditions are identical to those for the simulation of a single UMBE. The current of the IDA is calculated from Eq. (12.34) and multiplying by $N_e$, the number of electrode fingers in the array.

The computational domain for the IDA in generator-collector mode is shown in Fig. 12.18b in physical space and after normalisation by (12.113). Adjacent generator and collector electrodes have to be taken into account because different electrochemical processes are taking place at these electrodes and therefore there will be no symmetry line at the centre of the gap, as is the case for the parallel mode. One set of the electrodes of the IDA are set to a potential where reduction of a species A, present in the in the bulk, occurs, to generate species B

$$\mathrm{A} + \mathrm{e}^- \rightleftharpoons \mathrm{B}. \tag{12.116}$$

On the collector set of UMBEs, the generated species B is recycled back to A

$$\mathrm{B} \rightleftharpoons \mathrm{A} + \mathrm{e}^-, \tag{12.117}$$

c.f. Fig. 12.17c.

The IDA generator-collector system has been simulated using the Laasonen (BI) scheme [374], explicit FD [375, 376], hopscotch and conformal mapping [220], the finite analytic numerical method [152], extrapolation using expanding space intervals [332] and ADI with expanding space intervals [348, 377]. Commercial FEM software packages Flux-Expert® [324] and COMSOL Multiphysics® [378] have also been employed.

For the boundary conditions of the computational domain (12.18b, right-hand side) the left electrode serves as the generator and the right electrode as the collector. Both electrode potentials are set so that the redox reactions are at the diffusion limit and the surface concentration of the reacting species at the electrode is zero.

$$\begin{aligned}
&T = 0 && \text{all } R, Z: && C_A = 1, C_B = 0 && (12.118\text{a})\\
&T > 0 \\
&0 \le X \le 1, && Z = 0: && C_A = 0,\ C_B = 1 && (12.118\text{b})\\
&1 < X < (1 + W_g), && Z = 0: && \partial C_A/\partial Z = \partial C_B/\partial Z = 0 && (12.118\text{c})\\
&(1 + W_g) \le X \le (2 + W_g), && Z = 0: && C_A = 1,\ C_B = 0 && (12.118\text{d})\\
&X = 0, && 0 < Z < Z_{max}: && \partial C_A/\partial X = \partial C_B/\partial X = 0 && (12.118\text{e})\\
&X = 2 + W_g, && 0 < Z < Z_{max}: && \partial C_A/\partial X = \partial C_B/\partial Z = 0 && (12.118\text{f})\\
&0 \le X \le (2 + W_g), && Z = Z_{max}: && C_A = 1,\ C_B = 0 && (12.118\text{g})
\end{aligned}$$

The generator current (in Amperes) is calculated by solving the integral

$$i_g = 2n\mathcal{F}Dc^*l \int_o^{w/2} \left.\frac{\partial c_A}{\partial z}\right|_{z=0} \mathrm{d}x \tag{12.119}$$

and the collector current is given by

$$i_c = -2n\mathcal{F}Dc^*l \int_o^{w/2} \left.\frac{\partial c_B}{\partial z}\right|_{z=0} \mathrm{d}x, \tag{12.120}$$

where $w$, $g$ and $l$ are illustrated in Fig. 12.17a. Identical diffusion coefficients $D_A = D_B = D$ are assumed and $c^*$ is the initial or bulk concentration of species A. The computational domain contains only half of each electrode and this is taken into account by the factor 2. The current as calculated by Eqs. (12.119) and (12.120) is only for one generator and collector electrode. If the IDA has $N_e$ band electrodes, half of them operated as generators and half as collectors, a factor of $N_e/2$ is applied to the equations for $i_g$ and $i_c$. With the help of Eqs. (12.113) and (12.114) the currents are made dimensionless

$$G_g = 2\int_0^1 \left.\frac{\partial C_A}{\partial Z}\right|_{Z=0} \mathrm{d}X, \tag{12.121}$$

$$G_c = -2\int_{1+W_g}^{2+W_g} \left.\frac{\partial C_B}{\partial Z}\right|_{Z=0} \mathrm{d}X. \tag{12.122}$$

For the simulation of current transients the two-dimensional diffusion equation together with the boundary conditions (12.118) has to be solved. The implicit finite difference approach described for the direct discretisation of the UMDE, Sect. 12.3.2 can be readily applied. For example, one can use three points in each spatial direction, resulting in two-dimensional five point stencils. Three-point BDF, which is second-order accurate with respect to $\delta T$, can be used for time integration with a first BI step to start the simulation, as described in Sect. 12.3.2 for the UMDE or second-order accurate extrapolation, described in Sect. 4.9. The sparse matrix can be solved with the `MA28` package [261]. As for other UMEs, high numbers of grid points are required along the electrode surfaces and especially at the electrode edges $(X, Y) = (1, 0)$ and $(0, 1 + W_g)$, where the boundary conditions change abruptly. An example of the grid for the IDA in generator-collector mode (with a reduced number of grid points for clarity) is presented in Fig. 12.20a. The grid in the $X$ direction is created from the smallest space interval $\Delta X_1$ at both electrode edges and from the number of grid points along the electrode, $n_a$ and along the gap, $n_b$. In the $Z$ direction $n_z$ grid points expand from the smallest interval $\Delta Z_1$ at $Z = 0$ to $Z = Z_{max}$, where $Z_{max}$ is defined by Eq. (12.37). For simulation the following grid parameters were typically chosen: $\Delta X_1 = \Delta Z_1 = 10^{-4}$, $n_a = 40$, $n_b = 80$ (so $n_x = 160$) and $n_z = 100$. The constant time intervals were usually set to $\delta T = 0.01$.

### *12.5.3 Elevated UMBEs*

Photolithography, a common process for IDA electrodes, dual band and single band electrode production, leads to elevated bands, as shown by SEM and AFM [379]. Indeed, elevated IDA electrodes can be produced with various heights. IDA of elevated UMBEs have been simulated by the finite analytical numerical method [152], by the strongly implicit technique [243], by explicit finite differences [376], by a finite difference method with extrapolation [332] and using the commercial FEM program COMSOL Multiphysics® [378]. Elevated IDA structures with the side walls being conductive while the top of the elevated bands were non-conductive (comb IDAs) were fabricated by Kim et al. [380] and steady state currents were simulated using the commercial FEM program ANSYS®.

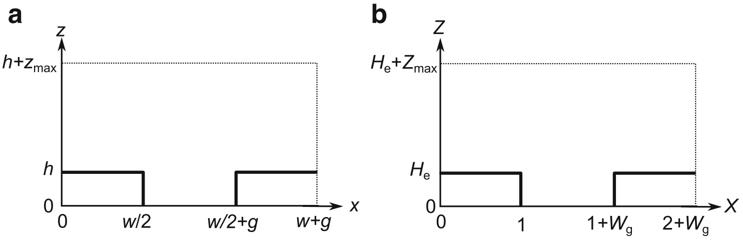


**Fig. 12.19** The computational domain for the simulation of an IDA of elevated band electrodes. (**a**) Physical and (**b**) dimensionless space

For simulation the same dimensionless variables as for IDA of co-planar band electrodes are used. The computational domain for an IDA of elevated band electrodes in physical and dimensionless space is shown in Fig. 12.19. The electrode height in physical dimensions is given by $h$. Then $H_e = 2h/w$ defines the dimensional electrode height. The electrode reaction occurs not only on the top of the electrodes, but also on their sides. The boundary conditions of the region shown in Fig. 12.19b for a potential step to the diffusion limit at both, the generator and collector are

$$T = 0 \qquad \text{all } R, Z: \qquad C_A = 1, C_B = 0 \tag{12.123a}$$

$$T > 0$$

$$0 \leq X \leq 1, \qquad Z = H_e: \qquad C_A = 0,\ C_B = 1 \tag{12.123b}$$

$$X = 1, \qquad 0 \leq Z < H_e: \qquad C_A = 0,\ C_B = 1 \tag{12.123c}$$

$$1 < X < (1 + W_g), \qquad Z = 0: \qquad \partial C_A/\partial Z = \partial C_B/\partial Z = 0 \tag{12.123d}$$

$$X = 1 + W_g, \qquad 0 \leq Z < H_e: \qquad C_A = 1,\ C_B = 0 \tag{12.123e}$$

$$(1 + W_g) \leq X \leq (2 + W_g), \quad Z = H_e: \qquad C_A = 1,\ C_B = 0 \tag{12.123f}$$

$$X = 0, \qquad H_e < Z < Z_{max}: \partial C_A/\partial X = \partial C_B/\partial X = 0 \tag{12.123g}$$

$$X = 2 + W_g, \qquad H_e < Z < Z_{max}: \partial C_A/\partial X = \partial C_B/\partial Z = 0 \tag{12.123h}$$

$$0 \leq X \leq (2 + W_g), \qquad Z = Z_{max}: \qquad C_A = 1,\ C_B = 0 \tag{12.123i}$$

The current not only flows on the top of the electrodes but also on the sides. This is taken into account by an integral of the surface gradients along the sides of the electrodes. The dimensionless current for a single generator electrode is

$$G_g = 2\int_0^1 \left.\frac{\partial C_A}{\partial Z}\right|_{Z=H_e} \mathrm{d}X + 2\int_0^{H_e} \left.\frac{\partial C_A}{\partial X}\right|_{X=1} \mathrm{d}Z \tag{12.124}$$

and for a single collector electrode

$$G_c = -2\int_{1+W_g}^{2+W_g} \left.\frac{\partial C_B}{\partial Z}\right|_{Z=H_e} \mathrm{d}X - 2\int_0^{H_e} \left.\frac{\partial C_B}{\partial X}\right|_{X=1+W_g} \mathrm{d}Z\,. \tag{12.125}$$

Each electrode has two sides and therefore there is a factor two in the second term of the left-hand side of Eqs. (12.124) and (12.125). These equations describe the dimensional current to a cross section in the $(X, Z)$ plane of the band electrodes. To obtain the dimensionless current over the whole bands, Eqs. (12.124) and (12.125) have to be multiplied by $2L$. However, it is preferable to consider the electrode length in the final step, when converting the dimensionless currents into currents with physical units. If the IDA consists of $N_e$ electrodes of length $l$ (unit meters), half of them running in generator modes, the other half in collector mode, then $G_g$ and $G_c$ are multiplied by $n\mathcal{F}Dc^*lN_e/2$ to obtain currents in physical units. A typical grid for simulating an IDA of elevated electrodes (with a reduced number of grid points for clarity) is shown in Fig. 12.20b. The elevation of the band electrode has consequences not only for the current but also for the *time of flight* experiments in generator-collector mode. The time of flight is a measure of how fast the communication by diffusion between the generator and collector electrode is, i.e. how quickly the generated species appears at the collector electrode [325]. The travel time $T_{0.5}$ is defined as the time at which the collector current reaches half of

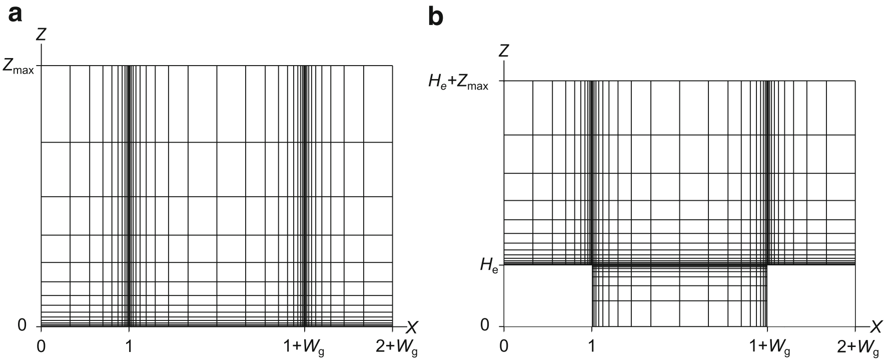


**Fig. 12.20** Grids for the simulation of IDA electrodes operated in generator-collector mode. (**a**) Co-planar electrodes; (**b**) elevated electrodes

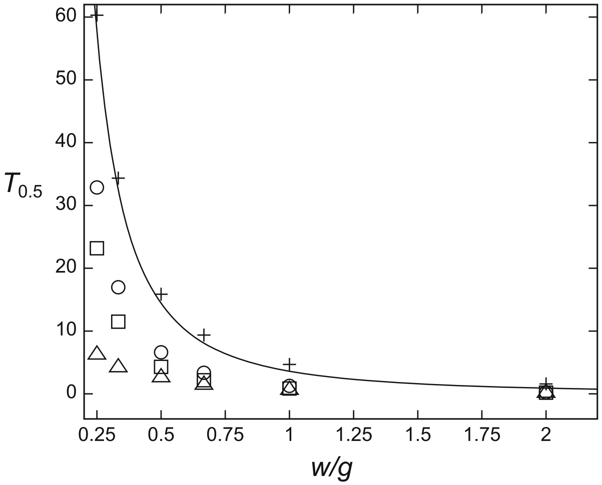


**Fig. 12.21** Travel time between band electrodes of an IDA as a function of the electrode width to gap width ratio. The line is calculated from Eq. (12.126), the symbols are simulation results for flat electrodes (*plus sign*), and electrodes with a height of $H_e = 1$ (*open circle*), 2 (*open square*) and 4 (*open triangle*).

its steady state value. Niwa et al. [381] introduced an empirical equation relating the electrode and gap width to the time of flight. An improved equation was presented by Aoki and Tanaka[382], which, modified for dimensionless variables, is

$$T_{0.5} = 3.6\left(\left(\frac{g}{w}\right)^2 + \frac{g}{w} + \frac{1}{36}\right). \tag{12.126}$$

Figure 12.21 shows a plot of the travel time $T_{0.5}$ between adjacent bands of an IDA as a function of the ratio $w/g$. The line in Fig. 12.21 was calculated from Eq. (12.126) and is shown together with simulated values for the flat electrodes (symbol $+$). Agreement of the simulations with (12.126) is good. As the gap width $g$ for a given electrode width increases (and thus $w/g$ decreases), $T_{0.5}$ steeply increases, an effect of the increasing length of path between the generator and collector electrodes. Also included in Fig. 12.21 are results of simulations for elevated bands with heights of $H_e = 1$ (the electrode height is half the electrode width), $H_e = 2$ (the electrode height is equal to the electrode width) and $H_e = 4$ (the electrode height is twice the electrode width). At larger $w/g$ ratios ($w/g \gtrapprox 2$) the travel times converge to the same value, independent of the electrode elevation. However, for smaller $w/g$ ratios the travel times become shorter the more elevated the electrodes are. For example at $w/g = 0.25$ the travel time of the coplanar IDA is reduced by factors of 1.8, 2.6 and 9.7 for $H_e$ values of 1, 2 and 4, respectively. The enhancement in current response and decrease in response time for electrodes operated in generator-collector mode for the case of $g < w$ has led to the development of *ultramicro gap electrodes*, such as the dual plate ultramicrotrench electrode [383–386]. Liu et al. [387] set the time of flight equivalent to the time delay $t_{onset}$ of the current onset at the collector electrode. For the potential sweep experiment, they calculated the time of flight from

$$t_{onset} = \frac{\Delta E_{onset}}{v} \tag{12.127}$$

where $v$ is the potential scan rate and $\Delta E_{onset}$ is the potential interval beginning at a point of the potential axis where the generator current started to rise to a point on the potential axis where the collector current started to flow. For a potential step experiment, the time delay $t_{onset}$ would be simply the period of time passed after the potential was applied at the generator electrode to the time where the collector current started to rise. In any case, the definition of $t_{onset}$ suffers from some arbitrariness with respect to the threshold of the current onset.

### *12.5.4 Dual Electrode Systems*

A device with two closely spaced electrodes both serving as working electrodes that might be individually addressable constitutes a dual electrode system, sometimes also termed a system of paired electrodes. Like the IDA electrodes treated in the previous section, we consider a dual electrode system in stagnant solution and operated in generator-collector mode. As IDA electrodes the dual electrode system can be used as an alternative to the rotating (macro) ring-disk electrode. For a review on dual electrode systems see [388]. Different electrode geometries have been explored, such as dual UMDEs [330, 335, 389, 390], dual hemispherical UMEs [391], dual bands [105, 112, 328, 392–395], ring-disk electrodes [396–399], dual cylinder [400, 401] and dual hemicylinder [331, 402] electrodes.

The lack of axial symmetry makes the simulation of dual disk and hemisphere systems computationally expensive because three space dimensions have to be taken into account. A few attempts have been made to numerically simulate such systems. The BEM in three space dimensions was used for steady-state simulations of dual hemispheres [279] and for time-dependent simulations of dual hemispheres and disk systems in generator-collector mode [403]. Current transient behaviour at dual disk electrodes has been simulated using explicit finite differences on an expanding mesh utilizing a graphics processor unit (GPU) together with the CPU to increase the performance of the computer [334]. However, each computation using the explicit method took between 1 and 3 days [334]. Diffusion can be regarded as a continuous random walk of particles and a three-dimensional random walk model [404] has been applied to model the generator-collector process at dual disk electrodes [405]. An improved model has been introduced [406], though in this case the disk electrodes are not co-planar but facing each other. Phillips and Stone [330] used an integral equation method to calculate steady state collection efficiencies for two geometries, the dual disk system and the ring disk system; for the latter, either the disk or the ring served as the generator. A first order homogeneous reaction was also considered. A simplification was made by assuming a constant current density across the generator electrodes.

The axial symmetry of the ring-disk electrode and the assumption that the length of the dual bands and dual cylinder electrodes are much bigger than their width and radius, respectively, allow reduction to two space dimensions, as was the case for single UMDE and UMBE simulations described in previous sections. Liljeroth et al. [397, 407] simulated the ring-disk electrode where a species was generated at the disk and collected at the ring using a commercial FEM toolbox FEMLAB® (predecessor of COMSOL Multiphysics®) connected to MATLAB®. For the simulation of dual band electrodes in generator-collector mode hopscotch ([99], see Sect. 9.2.5) with conformal mapping of the space [105, 217, 408], ADI with conformal mapping [326] and FEMLAB® monitored by MATLAB® [395] have been employed. Rajantie et al. [328] used spatial splitting of the fully implicit scheme, which overcomes error oscillation issues of the classical Peaceman and Rachford ADI scheme [100] while maintaining high accuracy by extrapolation. Conformal mapping of the space as described by Fosset and Amatore [217] was applied. Dual hemicylinders were simulated by Amatore et al. [331, 402] using ADI and conformal mapping. Dual band electrodes were used in electrochemical time-of-flight experiments to estimate diffusion coefficients by direct comparison with simulations [325]. A constant current mode instead of the more usual potentiostatic mode was employed to generate a species and the diffusion to the collector electrode was monitored potentiometrically. An approximate model of a dual band electrode was developed, using a hemicylindrical shape of the generator electrode to avoid unequal current distribution tied to the edge effect at planar (flat) UMBEs [325]. The diffusion equation was solved by a fully implicit FD scheme.

In the following, two ultramicrobands of the same width $w$ and length $l$ and separated by a gap $g$ are considered, both $w$ and $g$ in the µm range. Again, $l$ is much greater than $w$; so diffusion is essentially two-dimensional. The computational domain in $(x, z)$ coordinates is shown in Fig. 12.22a. The origin of the coordinates lies at the midpoint of the gap between the electrodes. Multiplication with $2/g$ renders the space variables dimensionless. The resulting domain in $(X, Z)$ is shown in Fig. 12.22b, where the dimensionless electrode width is now $W_e = 2w/g$. Dimensionless time and concentration are given by $T = 4Dt/g^2$ and $C_k = c_k/c^*$ with $D$ and $c^*$ being the diffusion coefficient and bulk concentration of a reference species. The diffusion equation for a species $k$ has the same form as for the single band and for the IDA system

$$\frac{\partial C_k}{\partial T} = d_k \left( \frac{\partial C_k^2}{\partial X^2} + \frac{\partial C_k^2}{\partial Z^2} \right) . \tag{12.128}$$

Assuming the redox reactions (12.116) and (12.117) occurring at the generator and collector electrode, respectively, the index $k$ presents $A$ and $B$. The dimensionless diffusion coefficient $d_k = D_k/D$ becomes unity if identical $D$'s are assumed for all

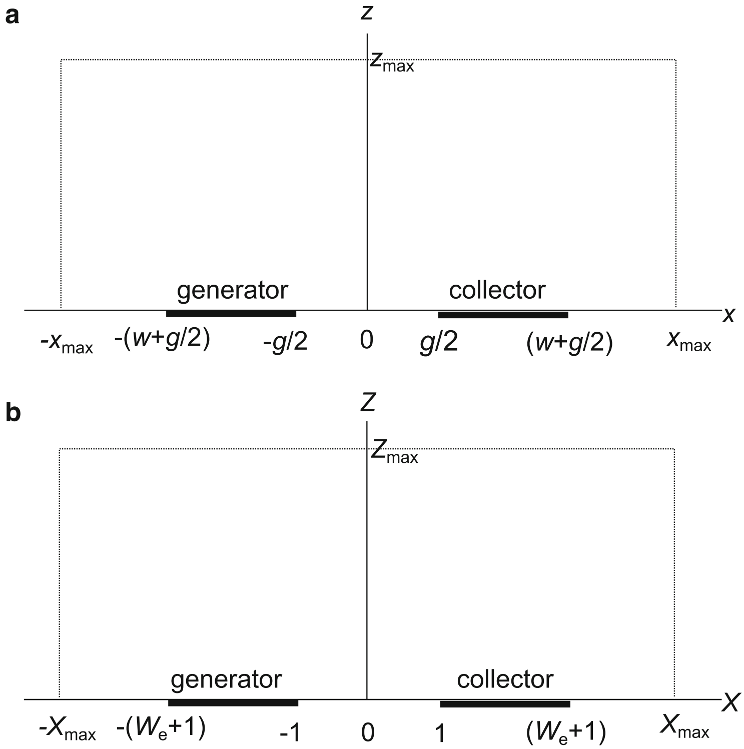


**Fig. 12.22** Computational domain in Cartesian coordinates for a dual ultramicroband system in generator-collector mode. (**a**) Physical space, (**b**) dimensionless space

species. If the diffusion coefficients are different and $D_A$ is used as a reference, then $d_A = 1$ and $d_B = D_B/D_A$.

The computational domains shown in Fig. 12.22b have three far field boundary conditions at $\pm X_{max}$ and $Z_{max}$. These are calculated similarly to Eqs. (12.37) and (12.37). Depending on $T_{max}$ the computational domain might become quite large, resulting in long computation times. As shown before, a more closed region can be accomplished by conformal mapping, thus avoiding the large values of space limits. Other advantages have been mentioned in Sect. 12.3.3. Here we show briefly a conformal map introduced by Amatore et al. for dual UMBEs [105, 217, 408]. His conformal map was also used to simulate electrochemical titration experiments [328]. The transformation equations from the $(X, Z)$ space to conformal coordinates $(\theta, \Gamma)$ are [217]

$$X = \cos\theta \cosh\Gamma \tag{12.129a}$$

$$Z = \sin\theta \sinh\Gamma \; . \tag{12.129b}$$

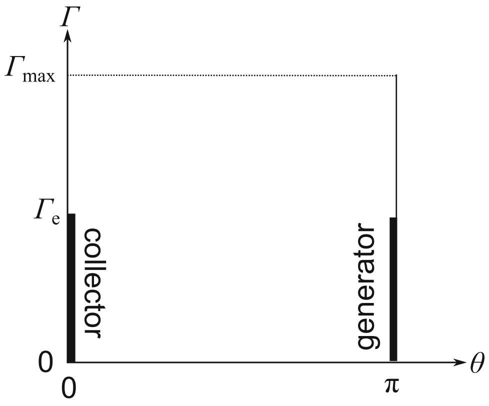


**Fig. 12.23** Computational domain in the conformal space as defined by (12.129) for a dual ultramicroband electrode in generator-collector mode

This is the MWA transformation mentioned in Sect. 12.3.3.1, but applied to bands. Inserting into Eq. (12.128) gives the time-dependent diffusion equation in conformal coordinates for species $k$,

$$\frac{\partial C_k}{\partial T} = \frac{d_k}{\sinh^2 \Gamma + \sin^2 \theta} \left( \frac{\partial C_k^2}{\partial \Gamma^2} + \frac{\partial C_k^2}{\partial \theta^2} \right) . \tag{12.130}$$

The computational domain in $(\theta, \Gamma)$ coordinates is presented in Fig. 12.23. For diffusion limited redox reactions at the generator and collector, the boundary conditions in conformal coordinates are

$$\begin{aligned}
&T = 0 && \text{all } \Gamma, \theta : && C_A = 1, C_B = 0 && \text{(12.131a)}\\
&T > 0 && && && \\
&\theta = 0, && \Gamma_e < \Gamma \leq \Gamma_{max} : && \partial C_A/\partial\theta = d_B \partial C_B/\partial\theta = 0, && \text{(12.131b)}\\
&\theta = 0, && 0 \leq \Gamma \leq \Gamma_e : && C_A = 1,\ C_B = 0 && \text{(12.131c)}\\
&0 < \theta < \pi, && \Gamma = 0 : && \partial C_A/\partial\Gamma = d_B \partial C_B/\partial\Gamma = 0 && \text{(12.131d)}\\
&\theta = \pi, && 0 \leq \Gamma \leq \Gamma_e : && C_A = 0,\ C_B = 1 && \text{(12.131e)}\\
&\theta = \pi, && \Gamma_e < \Gamma \leq \Gamma_{max} : && \partial C_A/\partial\theta = d_B \partial C_B/\partial\theta = 0, && \text{(12.131f)}\\
&0 < \theta < \pi, && \Gamma = \Gamma_{max} : && C_A = 1,\ C_B = 0. && \text{(12.131g)}
\end{aligned}$$

where Eqs. (12.131c) and (12.131e) describe the boundary conditions for the collector and generator electrode, respectively. In conformal space, the $\Gamma$ coordinate of the outer edges of the electrodes is given by

$$\Gamma_e = \operatorname{arccosh}(1 + W_e) , \tag{12.132}$$

which is also the width of the band electrodes, see Fig. 12.23. While the domain in the $\theta$ direction is closed, $0 \leq \theta \leq \pi$, the domain is open in the positive $\varGamma$ direction. A maximum value of $\varGamma$ has to be estimated from the expansion of the diffusion layer during the experiment. A value of $6\sqrt{T_{max}}$ is sufficient, see (5.5) in Sect. 5.2, where $T_{max}$ is the duration of the experiment to be simulated. Taking into account the position of the outer electrode edges in $(X, Z)$ space, Fig. 12.22b, the maximum values of $X$ are

$$X_{max} = \pm\left(1 + W_e + 6\sqrt{T_{max}}\right) , \tag{12.133}$$

from which the maximum value of $\varGamma$ is calculated to

$$\varGamma_{max} = \operatorname{arccosh}(X_{max}) , \tag{12.134}$$

as also shown in Appendix D. Currents of the generator and collector electrodes are calculated from

$$G_g = \int_0^{\varGamma_e} \left.\frac{\partial C_A}{\partial \theta}\right|_{\theta=\pi} \mathrm{d}\varGamma \tag{12.135}$$

and

$$G_c = -\int_0^{\varGamma_e} \left.\frac{\partial C_A}{\partial \theta}\right|_{\theta=0} \mathrm{d}\varGamma . \tag{12.136}$$

The current in units of Ampere is obtained by multiplication of these equations with $n\mathcal{F}D_A c^* l$, where $l$ is the length of the UMBE.

Simulation of electrochemical diffusion layer titration using dual band UMEs has been reported [326, 328], where homogeneous reactions were involved. The dual band was operated in generator-collector mode, so that the redox reactions (12.116) and (12.117) occurred at the generator and collector band, respectively. However, there was also an analyte C present in solution, that does not react at the electrodes but undergoes a homogeneous reaction with the generated species to convert B back to A.

$$\mathrm{B} + \mathrm{C} \rightarrow \mathrm{A} + \mathrm{P} . \tag{12.137}$$

B served as a titrant for C and the change of the collector current compared to the collector current in the absence of C was an indicator of (1) the amount of C present in solution and (2) the rate constant of reaction (12.137). Rajantie et al. [328] used ferrocyanide as a titrant which was generated from ferricyanide galvanostatically and detected at the collector band amperometrically. The analyte was ascorbic acid. Good agreement was found between simulation and experiment.

# References

1. Amatore C, Pebay C, Thouin L, Wang A, Warkocz JS (2010) Difference between ultramicroelectrodes and microelectrodes: influence of natural convection. Anal Chem 82:6933–6939
2. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
3. Laitinen HA, Kolthoff IM (1939) A study of diffusion processes by electrolysis with microelectrodes. J Am Chem Soc 61:3344–3349
4. Galus Z (1994) Fundamentals of electrochemical analysis, 2nd edn. Ellis Horwood, New York (trans: Chalmers RA, Bryce WAJ (eds))
5. Vetter K (1961) Elektrochemische kinetik. Springer, Berlin
6. Levich VG (1962) Physicochemical hydrodynamics. Prentice-Hall, New Jersey
7. Bond AM, Oldham KB, Zoski CG (1988) Theory of electrochemical processes at an inlaid disc microelectrode under steady-state conditions. J Electroanal Chem 245:71–104
8. Engstrom RC, Pharr CM, Koppang MD (1987) Visualization of the edge effect with electrogenerated chemiluminescence. J Electroanal Chem 221:251–255
9. Zeiri L, Younes O, Efrima S, Deutsch M (1997) Interfacial electrodeposition of silver. J Phys Chem B 101:9299–9308
10. Fleischmann M, Pons S, Rolison DR, Schmidt PP (1997) Ultramicroelectrodes. Dataetch Systems, Morganton, NC
11. Montenegro I, Queirós MA, Daschbach JL (eds) (1991) Microelectrodes: theory and applications. Nato science series E, vol 197. Springer, Heidelberg
12. Aoki K (1993) Theory of ultramicroelectrodes. Electroanalysis 5:627–639
13. Amatore C (1995) Electrochemistry at ultramicroelectrodes. In: Rubinstein I (ed) Physical electrochemistry. Marcel Dekker, New York, pp 131–208
14. Speiser B (1996) Numerical simulation of electroanalytical experiments: recent advances in methodology. In: Bard AJ, Rubinstein I (eds) Electroanalytical chemistry, vol 19. Marcel Dekker, New York, pp 1–108
15. Heinze J (1993) Elektrochemie mit Ultramikroelektroden. Angew Chem 105:1327–1349
16. Bieniasz LK, Britz D (2004) Recent developments in digital simulation of electroanalytical experiments. Pol J Chem 78:1195–1219
17. Chen S, Liu Y (2014) Electrochemistry at nanometer-sized electrodes. Phys Chem Chem Phys 16:635–652
18. Feeney R, Kounaves SP (2000) Microfabricated ultramicroelectrode arrays: developments, advances, and applications in environmental analysis. Electroanalysis 12:677–684
19. Forster R (2014) Micro- and nanoelectrodes. In: Kreysa G, i Ota K, Savinell RF (eds) Encyclopedia of applied electrochemistry. Springer, Heidelberg, pp 1248–1256 (Online)
20. Kranz C (2014) Recent advancements in nanoelectrodes and nanopipettes used in combined scanning electrochemical microscopy techniques. Analyst 139:336–352
21. Oja SM, Wood M, Zhang B (2013) Nanoscale electrochemistry. Anal Chem 85:473–486
22. Wang Y, Velmurugan J, Mirkin MV (2010) Kinetics of charge-transfer reactions at nanoscopic electrochemical interfaces. Israel J Chem 50:291–305
23. Oldham KB (1981) Edge effects in semiinfinite diffusion. J Electroanal Chem 122:1–17
24. Lingane PJ (1964) Chronopotentiometry and chronoamperometry with unshielded planar electrodes. Anal Chem 36:1723–1726
25. Soos ZG, Lingane PJ (1964) Derivation of the chronoamperometric constant for unshielded, circular, planar electrodes. J Phys Chem 68:3821–3828
26. Kakihana M, Ikeuchi H, Satô GP, Tokuda K (1980) Chronoamperometric measurement of diffusion coefficients with a microdisk electrode. J Electroanal Chem 108:381–383
27. Kakihana M, Ikeuchi H, Satô GP, Tokuda K (1981) Diffusion current at microdisk electrodes - application to accurate measurement of diffusion coefficients. J Electroanal Chem 117: 201–211
28. Heinze J (1981) Diffusion processes at finite (micro) disk electrodes solved by digital simulation. J Electroanal Chem 124:73–86

29. Flanagan JB, Marcoux L (1973) Digital simulation of edge effects at planar disc electrodes. J Phys Chem 77:1051–1055
30. Shoup D, Szabo A (1982) Chronoamperometric current at finite disk electrodes. J Electroanal Chem 140:237–245
31. Saito Y (1968) A theoretical study on the diffusion current at the stationary electrodes of circular and narrow band types. Rev Polarog (Jpn) 15:177–187
32. Zoski CG (1990) A survey of steady-state microelectrodes and experimental approaches to a voltammetric steady state. J Electroanal Chem 296:317–333
33. Liu X, Lu J, Cha C (1990) Square root law for the diffusion current at a microelectrode. J Electroanal Chem 294:289–292
34. Grigull U (1963) Die Grundgesetze der Wärmeübertragung, 3rd edn. Springer, Heidelberg
35. Gröber H (1921) Die Grundgesetze der Wärmeleitung und des Wärmeübergangs. Springer, Heidelberg. Often only Grigull is cited as the author. Reprint of the 3rd Ed. of 1955
36. Tranter CJ (1951) Integral transforms in mathematical physics. Methuen/Wiley, London/New York
37. Weber H (1873) Ueber die Besselschen Functionen und ihre Anwendung auf die Theorie der elektrischen Ströme. J Reine Angew Math 75:75–105
38. Sarangapani S, de Levie R (1979) On some problems of diffusion towards a circular disk. J Electroanal Chem 102:165–174
39. Carslaw HS, Jaeger JC (1959) Conduction of heat in solids, 2nd edn. Oxford Clarendon Press, Oxford
40. Crank J, Furzeland RM (1977) The treatment of boundary singularities in axially symmetric problems containing discs. J Inst Math Appl 20:355–370
41. Aoki K, Osteryoung J (1981) Diffusion-controlled current at the stationary finite disk electrode. Theory. J Electroanal Chem 122:19–35
42. Aoki K, Osteryoung J (1984) Formulation of the diffusion-controlled current at very small stationary disk electrodes. J Electroanal Chem 160:335–339
43. Phillips CG, Jansons KM (1990) The short-time transient of diffusion outside a conducting body. Proc R Soc Lond A 428:431–439
44. Gavaghan DJ (1998) An exponentially expanding mesh ideally suited to the fast and efficient simulation of diffusion processes at microdisc electrodes. 1. Derivation of the mesh. J Electroanal Chem 456:1–12
45. Gavaghan DJ (1998) An exponentially expanding mesh ideally suited to the fast and efficient simulation of diffusion processes at microdisc electrodes. 2. Application to chronoamperometry. J Electroanal Chem 456:13–23
46. Britz D, Poulsen K, Strutwolf J (2004) Reference values of the diffusion-limited current at a disk electrode. Electrochim Acta 50:107–113. See Erratum, ibid. 53:8101 (2008)
47. Mahon PJ, Oldham KB (2004) The transient current at the disk electrode under diffusion control: a new determination by the Cope-Tallman method. Electrochim Acta 49:5041–5048
48. Mahon PJ, Oldham KB (2005) Diffusion-controlled chronoamperometry at a disk electrode. Anal Chem 77:6100–6101
49. Heinze J (1981) Theory of cyclic voltammetry at microdisk electrodes. Ber Bunsenges Phys Chem 85:1096–1103
50. Aoki K, Akimoto K, Tokuda K, Matsuda H, Osteryoung J (1984) Linear sweep voltammetry at very small stationary disk electrodes. J Electroanal Chem 171:219–230
51. Aoki K, Tokuda K, Matsuda H (1986) Theory of chronoamperometric curves for a short time at microband electrodes. Denki Kagaku 54:1010–1017
52. Abramowitz M, Stegun IA (eds) (1969) Handbook of mathematical functions. Dover Publications, New York
53. Szabo A, Cope DK, Tallman DE, Kovach PM, Wightman RM (1987) Chronoamperometric current at hemicylinder and band microelectrodes: theory and experiment. J Electroanal Chem 217:417–423
54. Senthamarai R, Rajendran L (2008) Analytical expression for transient chronoamperometric current at ultramicroband electrode. Russ J Electrochem 44:1156–1161

55. Amatore C, Pebay C, Sella C, Thouin L (2012) Mass transport at microband electrodes: transient, quasi-steady-state, and convective regimes. Chem Phys Chem 13:1562–1568
56. Amatore CA, Fosset B, Deakin MR, Wightman RM (1987) Electrochemical kinetics and microelectrodes. Part III. Equivalency between band and hemicylinder electrodes. J Electroanal Chem 225:33–48
57. Coen S, Cope DK, Tallman DE (1986) Diffusion current at a band electrode by an integral equation method. J Electroanal Chem 215:29–48
58. Cope DK, Scott CH, Tallman DE (1990) Transient behavior at planar microelectrodes. Diffusion current at ring electrodes by the integral equation method. J Electroanal Chem 285:49–69
59. Cope DK, Tallman DE (1999) Transit behavior at planar microelectrodes. High efficiency algorithm for an integral equation method at a band electrode. Electrochem Soc Proc 99-5:82–89
60. Aoki K, Tokuda K (1987) Linear sweep voltammetry at microband electrodes. J Electroanal Chem 237:163–170
61. Bieniasz LK (2015) Theory of potential step chronoamperometry at a microband electrode: complete explicit semi-analytical formulae for the Faradaic current density and the Faradaic current. Electrochim Acta 178:25–33
62. Bieniasz LK (2016) Highly accurate, inexpensive procedures for computing theoretical chronoamperometric currents at single straight electrode edges and at single microband electrodes. J Electroanal Chem 760:71–79
63. Britz D, Østerby O, Strutwolf J (2010) Reference values of the chronoamperometric response at cylindrical and capped cylindrical electrodes. Electrochim Acta 55:5629–5635
64. Zoski CG, Mirkin MV (2002) Steady-state limiting currents at finite conical microelectrodes. Anal Chem 74:1986–1992
65. Selzer Y, Mandler D (1999) Comprehensive treatment of sphere-cap microelectrodes (SCMs) using digital simulations. Electrochem Commun 1:569–575
66. Britz D, Strutwolf J (2006) Electroanalytical response of an ultramicroelectrode at the bottom of an insulating conical well: digital simulation. Electrochim Acta 52:33–41
67. Birke R (1989) Steady state concentrations and currents on an oblate spheroid microelectrode. J Electroanal Chem 274:297–304
68. Diao GW, Li L, Zhang ZX (1996) The equation of chronoamperometric current on an oblate hemispheroid microelectrode and its verification. Chin J Chem 14:331–337
69. Oldham KB (1992) Steady-state shape voltammetry at microelectrodes of arbitrary shape. J Electroanal Chem 323:53–76
70. Amatore C, Oleinick A, Svir I (2004) Simulation of diffusion at microring electrodes through conformal mapping. J Electroanal Chem 564:245–260
71. Brookes BA, Davies TJ, Fisher AC, Evans RG, Wilkins SJ, Yunus K, Wadhawan JD, Compton RG (2003) Computational and experimental study of the cyclic voltammetry response of partially blocked electrodes. Part 1. Nonoverlapping, uniformly distributed blocking systems. J Phys Chem B 107:1616–1627
72. Davies TJ, Brookes BA, Fisher AC, Yunus K, Wilkins SJ, Greene PR, Wadhawan JD, Compton RG (2003) Computational and experimental study of the cyclic voltammetry response of partially blocked electrodes. Part II: randomly distributed and overlapping blocking systems. J Phys Chem B 107:6431–6449
73. Davies TJ, Brookes BA, Compton RG (2004) A computational and experimental study of the cyclic voltammetry response of partially blocked electrodes, part III: interfacial liquid–liquid kinetics of aqueous vitamin $B_{12s}$ with random arrays of femtolitre microdroplets of dibromocyclohexane. J Electroanal Chem 566:193–216
74. Engstrom RC, Weber M, Wunder DJ, Burgess R, Winquist S (1986) Measurements within the diffusion layer using a microelectrode probe. Anal Chem 58:844–848
75. Cornut R, Griveau S, Lefrou C (2010) Accuracy study on fitting procedure of kinetics SECM feedback experiments. J Electroanal Chem 650:55–61

76. Deng H, Peljo P, Momotenko D, Cortés-Salazar F, Stockmann TJ, Kontturi K, Opallo M, Girault HH (2014) Kinetic differentiation of bulk/interfacial oxygen reduction mechanisms at/near liquid/liquid interfaces using scanning electrochemical microscopy. J Electroanal Chem 732:101–109
77. Fernández JL, Wijesinghe M, Zoski CG (2015) Theory and experiments for voltammetric and SECM investigations and application to ORR electrocatalysis at nanoelectrode ensembles of ultramicroelectrode dimensions. Anal Chem 87:1066–1074
78. Holder MN, Gardner CE, Macpherson JV, Unwin PR (2005) Combined scanning electrochemical-atomic force microscopy (SECM-AFM): simulation and experiment for flux-generation at un-insulated metal-coated probes. J Electroanal Chem 585:8–18
79. Jensen MB, Tallman DE (2013) A LabVIEW-based virtual instrument for simulation and analysis of SECM approach curves. J Solid State Electrochem 17:2999–3003
80. Kiss A, Nagy G (2015) Deconvolution in potentiometric SECM. Electroanalysis 27:587–590
81. Lefrou C (2006) A unified new analytical approximation for positive feedback currents with a microdisk SECM tip. J Electroanal Chem 592:103–112
82. Leonhardt K, Avdic A, Lugstein A, Pobelov I, Wandlowski T, Gollas B, Denuault G (2013) Scanning electrochemical microscopy: diffusion controlled approach curves for conical AFM-SECM tips. Electrochem Commun 27:29–33
83. Li F, Unwin PR (2015) Scanning electrochemical microscopy (SECM) of photoinduced electron transfer kinetics at liquid/liquid interfaces. J Phys Chem C 119:4031–4043
84. Lindsey G, Abercrombie S, Denuault G, Daniele S, De Faveri E (2007) Scanning electrochemical microscopy: approach curves for sphere-cap scanning electrochemical microscopy tips. Anal Chem 79:2952–2956
85. Lucas M, Stafiej J, Slim C, Delpech S, di Caprio D (2014) Cellular automata modeling of scanning electrochemical microscopy (SECM) experiments. Electrochim Acta 145:314–318
86. Mahé É (2007) Electrochemistry in confined microsystems. Part 1. CV-SECM of reversible system at a confined disk ultramicroelectrode. Electrochim Acta 52:5018–5029
87. Sklyar O, Kueng A, Kranz C, Mizaikoff B, Lugstein A, Bertagnolli E, Wittstock G (2005) Numerical simulation of scanning electrochemical microscopy experiments with frame-shaped integrated atomic force microscopy - SECM probes using the boundary element method. Anal Chem 77:764–771
88. Sun P, Mirkin MV (2007) Scanning electrochemical microscopy with slightly recessed nanotips. Anal Chem 79:5809–5816
89. Träuble M (2004) Modellierung und Simulation elektrochemischer Prozesse mit Randelementmethoden. Ph.D. thesis, Universität Oldenburg, Oldenburg (in German)
90. Wu ZQ, Zhou T, Wang K, Zhang JR, Xia XH (2010) Current distribution at electrode surfaces as simulated by finite element method. Electrochim Acta 55:4870–4875
91. Yao D, Chen J, Liu X, Lu X (2010) Progress in the theory of electron transfer across liquid/liquid interfaces based on thin layer. Chem Bull/Huaxue Tongbao 73:3–9 (in Chinese with some English content)
92. Britz D, Strutwolf J (2014) Digital simulation of electrochemistry at microelectrodes. In: Lei KF (ed) Microelectrodes. Nova Science Publishers, New York, pp 1–85
93. Oldham KB, Zoski CG (1988) Comparison of voltammetric steady states at hemispherical and disc microelectrodes. J Electroanal Chem 256:11–19
94. Gavaghan DJ (1997) How accurate is your two-dimensional numerical simulation? Part 1. An introduction. J Electroanal Chem 420:147–158
95. Fernández JL, Bard AJ (2004) Scanning electrochemical microscopy 50. Kinetic study of electrode reactions by the tip generation-substrate collection mode. Anal Chem 76:2281–2289
96. Myland JC, Oldham KB (1996) The three-dimensional simulation of short-time diffusion-controlled voltammetry at the walls of a hollow cube. J Electroanal Chem 405:39–50
97. Myland JC, Oldham KB (2005) Modelling diffusion to a disk electrode by fully explicit simulation. J Electroanal Chem 576:353–362

98. Myland J, Oldham KB (2014) The excess current in cyclic voltammetry arising from the presence of an electrode edge. J Solid State Electrochem 18:3259–3269
99. Gourlay AR (1970) Hopscotch: a fast second-order partial differential equation solver. J Inst Math Appl 6:375–390
100. Peaceman DW, Rachford HH (1955) The numerical solution of parabolic and elliptic differential equations. J Soc Ind Appl Math 3:28–41
101. Evans NTS, Gourlay AR (1977) The solution of a two-dimensional time-dependent diffusion problem concerned with oxygen metabolism in tissues. J Inst Math Appl 19:239–251
102. Heinze J (1984) Theory of cyclic voltammetry at microdisk electrodes. Ber Bunsenges Phys Chem 85:1096–1103
103. Heinze J, Störzbach M (1986) Electrochemistry at ultramicroelectrodes - simulation of heterogeneous and homogeneous kinetics by an improved ADI-technique. Ber Bunsenges Phys Chem 90:1043–1048
104. Amatore C, Svir I (2003) A new and powerful approach for simulation of diffusion at microelectrodes based on overlapping sub-domains: application to chronoamperometry at the microdisk. J Electroanal Chem 557:75–90
105. Arkoub IA, Amatore C, Sella C, Thouin L, Warkocz JS (2001) Diffusion at double microband electrodes operated within a thin film coating. Theory and experimental illustration. J Phys Chem B 105:8694–8703
106. Gaidamauskaite E, Baronas R (2007) A comparison of finite difference schemes for computational models of biosensors. Nonlinear Anal Modell Control 12:359–369
107. Ikeuchi H, Kanakubo M (2000) Determination of diffusion coefficients of the electrode reaction products by the double potential step chronoamperometry at small disk electrodes. J Electroanal Chem 493:93–99
108. Carnahan B, Luther HA, Wilkes JO (1969) Applied numerical methods. Wiley, New York
109. Feldberg SW (1987) Propagational inadequacy of the hopscotch finite difference algorithm: the enhancement of performance when used with an exponentially expanding grid for simulation of electrochemical diffusion problems. J Electroanal Chem 222:101–106
110. Safford LK, Weaver MJ (1989) The combined influences of solution resistance and charge-transfer kinetics on microelectrode cyclic voltammetry. J Electroanal Chem 261:241–247
111. Taylor G, Girault HH, McAleer J (1990) Digital simulation of charge transfer to an ultramicrodisk interface. J Electroanal Chem 293:19–44
112. Barnes EO, Lewis GEM, Dale SEC, Marken F, Compton RG (2013) Dual band electrodes in generator-collector mode: simultaneous measurement of two species. J Electroanal Chem 703:38–44
113. Ellison J, Eloul S, Batchelor-McAuley C, Tschulik K, Salter C, Compton RG (2015) The effect of insulator nano-sheath thickness on the steady state current at a micro-disc electrode. J Electroanal Chem 745:66–71
114. Eloul S, Compton RG (2014) Voltammetric sensitivity enhancement by using preconcentration adjacent to the electrode: simulation, critical evaluation, and insights. J Phys Chem C 118:24520–24532
115. Eloul S, Compton RG (2014) Shielding of a microdisc electrode surrounded by an adsorbing surface. Chem Electrochem 1:917–924
116. Henstridge MC, Ward KR, Compton RG (2014) The Marcus-Hush model of electrode kinetics at a single nanoparticle. J Electroanal Chem 712:14–18
117. Lin C, Compton RG (2015) Voltammetric mechanistic characterisation of electrode reactions: distinguishing between chemical instability and fast product diffusion. J Electroanal Chem 743:86–92
118. Molina A, Gonzalez J, Barnes EO, Compton RG (2014) Simple analytical equations for the current-potential curves at microelectrodes: a universal approach. J Phys Chem C 118:346–356
119. Molina A, Olmos J, Laborda E (2015) Reverse pulse voltammetry at spherical and disc microelectrodes: characterization of homogeneous chemical equilibria and their impact on the species diffusivities. Electrochim Acta 169:300–309

120. Ngamchuea K, Eloul S, Tschulik K, Compton RG (2014) Planar diffusion to macro disc electrodes - what electrode size is required for the Cottrell and Randles-Sevcik equations to apply quantitatively? J Solid State Electrochem 18:3251–3257
121. Wang H, Du N (2014) Fast alternating-direction finite difference methods for three-dimensional space-fractional diffusion equations. J Comput Phys 258:305–318
122. Ward KR, Lawrence NS, Hartshorne RS, Compton RG (2012) Modelling the steady state voltammetry of a single spherical nanoparticle on a surface. J Electroanal Chem 683:37–42
123. Ward KR, Xiong L, Lawrence NS, Hartshorne RS, Compton RG (2013) Thin-layer vs. semi-infinite diffusion in cylindrical pores: a basis for delineating Fickian transport to identify nano-confinement effects in voltammetry. J Electroanal Chem 702:15–24
124. Britz D, Østerby O, Strutwolf J (2003) Damping of Crank-Nicolson error oscillations. Comput Biol Chem 27:253–263
125. Østerby O (2003) Five ways of reducing the Crank-Nicolson oscillations. BIT Numer Math 43:811–822
126. Pearson CE (1965) Impulsive end condition for diffusion equation. Math Comput 19:570–576
127. Khaliq AQM, Wade BA (2001) On smoothing of the Crank-Nicolson scheme for nonhomogeneous parabolic problems. J Comput Methods Sci Eng 1:107–124
128. Luskin M, Rannacher R (1982) On the smoothing properties of the Crank-Nicolson scheme. Appl Anal 14:117–135
129. Rannacher R (1982) Discretisation of the heat equation with singular initial data. Z Angew Math Mech 62:T346–T348
130. Rannacher R (1984) Finite element solution of diffusion problems with irregular data. Numer Math 43:309–327
131. Østerby O (2002) Five ways of reducing the Crank-Nicolson oscillations. Technical Report Daimi PB-558, Department of Computer Science, Aarhus University
132. Bard AJ, Crayston JA, Kittlesen GP, Shea TV, Wrighton MS (1986) Digital simulation of the measured electrochemical response of reversible redox couples at microelectrode arrays: consequences arising from closely spaced ultramicroelectrodes. Anal Chem 58:2321–2331
133. Britz D (1996) Brute force digital simulation. J Electroanal Chem 406:15–21
134. Nann T (1997) Digitale Simulation in der Elektrochemie mit der Methode der Finiten Elementen. Ph.D. thesis, Albert-Ludwigs-Universität zu Freiburg im Breisgau. Publ. by Shaker Verlag, Aachen
135. Nann T, Heinze J (1999) Simulation in electrochemistry using the finite element method. Part 1. The algorithm. Electrochem Commun 1:289–294
136. Nann T, Heinze J (2003) Simulation in electrochemistry using the finite element part 2: scanning electrochemical microscopy. Electrochim Acta 48:3975–3880
137. Gavaghan DJ, Gillow K, Süli E (2006) Adaptive finite element methods in electrochemistry. Langmuir 22:10666–10682
138. Harriman K, Gavaghan DJ, Houston P, Kay D, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. *ECE* and *EC*$_2$*E* mechanisms at channel microband electrodes. Electrochem Commun 2:576–585
139. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. An E reaction at a channel microband electrode. Electrochem Commun 2:567–575
140. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Application to a simple model problem. Electrochem Commun 2:150–156
141. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. First-order EC' mechanism at inlaid and recessed discs. Electrochem Commun 2:163–170
142. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Theory. Electrochem Commun 2:157–162

143. Harriman K, Gavaghan DJ, Süli E (2003) Adaptive finite element simulation of chronoamperometry at microdisc electrodes. Electrochem Commun 5:519–529
144. Harriman K, Gavaghan DJ, Süli E (2004) Time-dependent EC', ECE and $EC_2E$ mechanisms at microdisc electrodes: simulation using adaptive finite element methods. J Electroanal Chem 569:35–46
145. Rudolph M (2002) Digital simulation on unequally spaced grids. Part 1. Critical remarks on using the point method by discretisation on a transformed grid. J Electroanal Chem 529: 97–108
146. Gavaghan DJ (1998) An exponentially expanding mesh ideally suited to the fast and efficient simulation of diffusion processes at microdisc electrodes. 3. Application to voltammetry. J Electroanal Chem 456:25–35
147. Newman J (1966) Current distribution on a rotating disk below the limiting current. J Electrochem Soc 113:1235–1241
148. Michael AC, Wightman RM, Amatore CA (1989) Microdisk electrodes. Part 1. Digital simulation with a conformal map. J Electroanal Chem 267:33–45
149. Safford LK, Weaver MJ (1991) Cyclic voltammetric wave-shapes for microdisk-electrodes: coupled effects of solution resistance, double-layer capacitance, and finite electrochemical kinetics. J Electroanal Chem 312:69–96
150. Deakin MR, Wightman RM, Amatore CA (1986) Electrochemical kinetics at microelectrodes. Part II. Cyclic voltammetry at band electrodes. J Electroanal Chem 215:49–61
151. Amatore CA, Deakin MR, Wightman RM (1986) Electrochemical kinetics at microelectrodes. Part I. Quasi-reversible electron transfer at cylinders. J Electroanal Chem 206:23–36
152. Jin B, Qian W, Zhang Z, Shi H (1996) Application of the finite analytic numerical method. Part 1. Diffusion problems on coplanar and elevated interdigitated microarray band electrodes. J Electroanal Chem 411:29–36
153. Jin B, Qian W, Zhang Z, Shi H (1996) Application of the finite analytic numerical method. Part 3. Digital simulation of charge transfer to a micro-ring electrode interface. J Electroanal Chem 417:45–51
154. Jin BK, Shi HS, Zhang ZX (1996) Equation of voltammetric waves controlled by diffusion, charge transfer and chemical reactions at ultamicroband electrode. Chem J Chin Univ 17:1052–1055 [in Chinese, Eng. abstract]
155. Qian W, Jin B, Diao G, Zhang Z, Shi H (1996) Application of a finite analytic numerical method. Part 2. Digital simulation of charge transfer to an oblate hemispheroid microelectrode and experiment verification. J Electranal Chem 414:1–10
156. Qian W, Jin B, Shi H, Zhang Z (1997) Finite analytic solution and finite analytic numerical method for solving two-dimensional diffusion problems on microelectrodes. J Electroanal Chem 439:29–36
157. Qian W, Jin BK, Shi HS, Yu JS, Zhang ZX (1997) Digital simulation of chronoamperometric current at microdisk electrode and its verification. Acta Chim Sin 55:1108–1115
158. Varco Shea T, Bard AJ (1987) Digital simulation of homogeneous chemical reactions coupled to heterogeneous electron transfer and applications at platinum/mica/platinum ultramicroband electrodes. Anal Chem 59:2101–2111
159. Bieniasz LK, Britz D (2001) Chronopotentiometry at a microband electrode: simulation study using a Rosenbrock time integration scheme for differential-algebraic equations and a direct sparse solver. J Electroanal Chem 503:141–152
160. Britz D, Chandra S, Strutwolf J, Wong DKY (2010) Diffusion-limited chronoamperometry at conical-tip microelectrodes. Electrochim Acta 55:1272–1277
161. Britz D, Poulsen K, Strutwolf J (2005) Reference values of the diffusion-limited chronoamperometric current at a microband electrode. Electrochim Acta 51:333–339. See Erratum, ibid. 53:7805 (2008)
162. Amphlett JL, Denuault G (1998) Scanning electrochemical microscopy (SECM): an investigation of the effects of tip geometry on amperometric tip response. J Phys Chem B 102:9946–9951

163. Barker AL, Macpherson JV, Slevin CJ, Unwin PR (1998) Scanning electrochemical microscopy (SECM) as a probe of transfer processes in two-phase systems: theory and experimental applications of SECM-induced transfer with arbitrary partition coefficients, diffusion coefficients, and interfacial kinetics. J Phys Chem B 102:1586–1598
164. Barker AL, Unwin PR, Amemiya S, Zhou J, Bard AJ (1999) Scanning electrochemical microscopy (SECM) in the study of electron transfer kinetics at liquid/liquid interfaces: beyond the constant composition approach. J Phys Chem B 103:7260–7269
165. Barker AL, Unwin PR, Zhang J (2001) Measurement of the forward and back rate constants for electron transfer at the interface between two immiscible electrolyte solutions using scanning electrochemical microscopy (SECM): theory and experiment. Electrochem Commun 3:372–378
166. Barker AL, Unwin PR (2001) Measurement of solute partitioning across liquid/liquid interfaces using scanning electrochemical microscopy - double potential step chronoamperometry (SECM-DPSC): principles, theory, and application to ferrocenium ion transfer across the 1,2-dichloroethane/aqueous interface. J Phys Chem B 105:12019–12031
167. Cornut R, Lefrou C (2007) New analytical approximations for negative feedback currents with a microdisk SECM tip. J Electroanal Chem 604:91–100
168. Demaille C, Unwin PR, Bard AJ (1996) Scanning electrochemical microscopy. 33. Application to the study of ECE/DISP Reactions. J Phys Chem 100:14137–14143
169. Georganopoulou DG, Caruana DJ, Strutwolf J, Williams DE (2000) Electron transfer mediated by glucose oxidase at the liquid/liquid interface. Faraday Discuss 116:109–118
170. Macpherson JV, Unwin PR (1996) Scanning electrochemical microscope-induced dissolution: theory and experiment for silver chloride dissolution kinetics in aqueous solution without supporting electrolyte. J Phys Chem 100:19475–19483
171. Martin RD, Unwin PR (1997) Scanning electrochemical microscopy: theory and experiment for the positive feedback mode with unequal diffusion coefficients of the redox mediator couple. J Electroanal Chem 439:123–136
172. Martin RD, Unwin PR (1998) Theory and experiment for the substrate generation/tip collection mode of the scanning electrochemical microscope: application as an approach for measuring the diffusion coefficient ratio of a redox couple. Anal Chem 70:276–284
173. Mauzeroll J, Hueske EA, Bard AJ (2003) Scanning electrochemical microscopy. 48. Hg/Pt hemispherical ultramicroelectrodes: fabrication and characterization. Anal Chem 75:3880–3889
174. Selzer Y, Mandler D (2000) Scanning electrochemical microscopy. Theory of the feedback mode for hemispherical ultramicroelectrodes: steady-state and transient behavior. Anal Chem 72:2383–2390
175. Shao Y, Mirkin MV (1998) Probing ion transfer at the liquid/liquid interface by scanning electrochemical microscopy (SECM). J Phys Chem B 102:9915–9921
176. Sklyar O, Wittstock G (2002) Numerical simulations of complex nonsymmetrical 3D systems for scanning electrochemical microscopy using the boundary element method. J Phys Chem B 106:7499–7508
177. Sklyar O, Ufheil J, Heinze J, Wittstock G (2003) Application of the boundary element method numerical simulations for characterization of heptode ultramicroelectrodes in SECM experiments. Electrochim Acta 49:117–128
178. Sklyar O (2004) Modelling scanning electrochemical microscopy (SECM) experiments on microstructured functionalised surfaces. Ph.D. thesis, Universität Oldenburg, Oldenburg
179. Slevin CJ, Macpherson JV, Unwin PR (1997) Measurement of local reactivity at liquid/solid, liquid/liquid, and liquid/gas interfaces with the scanning electrochemical microscope: principles, theory, and applications of the double potential step chronoamperometric mode. J Phys Chem B 101:10851–10859
180. Strutwolf J, Barker AL, Gonsalves M, Caruana DJ, Unwin PR, Williams DE, Webster JPR (2000) Probing liquid | liquid interfaces using neutron reflection measurements and scanning electrochemical microscopy. J Electroanal Chem 483:163–173

181. Unwin PR, Bard AJ (1991) Scanning electrochemical microscopy. 9. Theory and application of the feedback mode to the measurement of following chemical reaction rates in electrode processes. J Phys Chem 95:7814–7824
182. Britz D, Strutwolf J (2013) Digital simulation of chronoamperometry at an electrode within a hemispherical polymer drop containing an enzyme: comparison of a hemispherical with a flat disk electrode. Biosens Bioelectron 50:269–277
183. Britz D, Strutwolf J (2015) Digital simulation of chronoamperometry at a disk electrode under a flat polymer film containing an enzyme. Electrochim Acta 152:302–307
184. Che G, Dong S (1992) Application of ultramicroelectrodes in studies of homogeneous catalytic reactions– Part II. A theory of quasi-first and second-order homogeneous catalytic reactions. Electrochim Acta 37:2695–2699
185. Che G, Dong S (1992) Application of ultramicroelectrodes in studies of homogeneous catalytic reactions– Part III. The condition for quasi-first and second order homogeneous catalytic reactions at ultramicrodisk electrodes in the steady state. Electrochim Acta 37:2701–2705
186. Dong S, Che G (1992) The application of an ultramicroelectrode in homogeneous catalytic reaction – Part I. General characteristic of a homogeneous catalytic reaction at an ultramicroelectrode with arbitrary geometry under steady state. Electrochim Acta 37:2587–2589
187. Kottke PA, Kranz C, Kwon YK, Masson JF, Mizaikoff B, Fedorov AG (2008) Theory of polymer entrapped enzyme ultramicroelectrodes: fundamentals. J Electroanal Chem 612:208–218
188. Meena A, Rajendran L (2010) Analytical solution of system of coupled nonlinear reaction diffusion equations. Part II: direct reaction of substrate at underlying microdisc surface. J Electroanal Chem 650:143–151
189. Phanthong C, Somasundrum M (2003) The steady state current at a microdisk biosensor. J Electroanal Chem 558:1–8
190. Cope DK, Scott CH, Kalapathy U, Tallman DE (1990) Transient behavior at planar microelectrodes. Diffusion current at a band electrode by an integral equation method. Part II. J Electroanal Chem 280:27–35
191. Cope DK, Tallman DE (1990) Transient behavior at planar microelectrodes. Diffusion current at the disk electrode by the integral equation method. J Electroanal Chem 285:79–84
192. Cope DK, Tallman DE (1990) Transient behavior at planar microelectrodes. A comparison of diffusion current at ring, band and disk electrodes. J Electroanal Chem 285:85–92
193. Engblom SO, Cope DK, Tallman DE (1996) Diffusion current at the tubular band electrode by the integral equation method. J Electroanal Chem 406:23–31
194. Bieniasz LK (2015) Modelling electroanalytical experiments by the integral equation approach. Springer, Heidelberg
195. Bieniasz LK (1992) An efficient numerical method of solving the Abel integral equation for cyclic voltammetry. Comput Chem 16:311–317
196. Bieniasz LK (2008) An adaptive Huber method with local error control, for the numerical solution of the first kind Abel integral equations. Computing 83:25–39
197. Bieniasz LK (2008) Cyclic voltammetric current functions determined with a prescribed accuracy by the adaptive Huber method for Abel integral equations. Anal Chem 80:9659–9665
198. Bieniasz LK (2010) Automatic simulation of cyclic voltammograms by the adaptive Huber method for weakly singular second kind Volterra integral equations. Electrochim Acta 55:721–728
199. Bieniasz LK (2010) Automatic simulation of cyclic voltammograms by the adaptive Huber method for systems of weakly singular Volterra integral equations. J Electroanal Chem 642:127–134
200. Bieniasz LK (2010) An adaptive Huber method for weakly singular second kind Volterra integral equations with nonlinear dependencies between unknowns and their integrals. Computing 87:35–54

201. Bieniasz LK (2011) An adaptive Huber method for nonlinear systems of weakly singular second kind Volterra integral equations. Appl Math Comput 217:5622–5631
202. Bieniasz LK (2011) Analysis of the applicability of the integral equation method in the theory of transient electroanalytical experiments for homogeneous reaction-diffusion systems: the case of planar electrodes. J Electroanal Chem 657:91–97
203. Bieniasz LK (2011) Extension of the adaptive Huber method for solving integral equations occurring in electroanalysis, onto Kernel function representing fractional diffusion. Electroanalysis 23:1506–1511
204. Bieniasz LK (2011) Automatic simulation of electrochemical transients at cylindrical wire electrodes, by the adaptive Huber method for Volterra integral equations. J Electroanal Chem 662:371–378
205. Bieniasz LK (2011) Extension of the adaptive Huber method for Volterra integral equations arising in electroanalytical chemistry, to convolution kernels $\exp[-\alpha(t-\tau)]$ erex $\{[\beta(t-\tau)]^{1/2}\}$ and $\exp[-\alpha(t-\tau)]$ daw $\{[\beta(t-\tau)]^{1/2}\}$. J Comput Methods Sci Eng 11:323–338
206. Bieniasz LK (2012) Automatic simulation of electrochemical transients by the adaptive Huber method for Volterra integral equations involving kernel terms $\exp[-\alpha(t-\tau)]\mathrm{erex}\{[\beta(t-\tau)]\}$ and $\exp[-\alpha(t-\tau)]\mathrm{daw}\{[\beta(t-\tau)]\}$. J Math Chem 50:765–781
207. Bieniasz LK (2012) Automatic solution of integral equations pertinent to diffusion with first order homogeneous reactions at cylindrical wire electrodes. J Electroanal Chem 674:38–47
208. Bieniasz LK (2012) Automatic simulation of electrochemical transients assuming finite diffusion space at planar interfaces, by the adaptive Huber method for Volterra integral equations. J Electroanal Chem 684:20–31
209. Bieniasz LK (2013) Automatic solution of the Singh and Dutt integral equations for channel or tubular electrodes, by the adaptive Huber method. J Electroanal Chem 693:95–104
210. Bieniasz LK (2013) Automatic solution of integral equations describing electrochemical transients under conditions of internal spherical diffusion. J Electroanal Chem 694:104–113
211. Bieniasz LK (2013) Automatic solution of integral equations describing electrochemical transients at dropping mercury electrodes. J Electroanal Chem 705:44–51
212. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical recipes in Fortran. The art of scientific computing, 2nd edn. Cambridge University Press, Cambridge
213. Balslev H, Britz D (1992) Direct digital simulation of the steady-state limiting current at a rotating disk electrode for a complex mechanism. Acta Chem Scand 46:949–955
214. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376
215. Gourlay AR, McKee S (1977) The construction of hopscotch methods for parabolic and elliptic equations in two space dimensions with a mixed derivative. J Comput Appl Math 3:201–206
216. Amatore CA, Fosset B (1992) Space variables well fitted for the study of steady state and near-steady-state diffusion at a microdisk. J Electroanal Chem 328:21–32
217. Fosset B, Amatore CA, Bartelt JE, Michael AC, Wightman RM (1991) Use of conformal maps to model the voltammetric response of collector-generator double-band electrodes. Anal Chem 63:306–314
218. Lavagnini I, Pastore P, Magno F, Amatore CA (1991) Performance of a numerical method based on the hopscotch algorithm and on an oblate spheroidal space coordinate- expanding time grid for simulation of voltammetric curves at an inlaid disk microelectrode. J Electroanal Chem 316:37–47
219. Pastore P, Magno F, Lavagnini I, Amatore C (1991) Digital simulation via the hopscotch algorithm of a microelectrode-based channel flow-through amperometric detector. J Electroanal Chem 301:1–13
220. Postlethwaite TA, Hutchinson JE, Murray R, Fosset B, Amatore C (1996) Interdigitated array electrodes as an alternative to the rotating ring-disk electrode for determination of the reaction products of dioxygen reduction. Anal Chem 68:2951–2958
221. Shoup D, Szabo A (1984) Hopscotch: an algorithm for the numerical solution of electrochemical problems. J Electroanal Chem 160:1–17

222. Shoup D, Szabo A (1986) Explicit hopscotch and implicit finite-difference algorithms for the Cottrell problem: exact analytical results. J Electroanal Chem 199:437–441
223. Danaee A, Evans DJ (1988) A composite hopscotch method of increased accuracy. Int J Comput Math 24:181–198
224. Lapidus L, Pinder GF (1982) Numerical solution of partial differential equations in science and engineering. Wiley, New York
225. Hunsdorfer WH, Verwer JG (1989) Stability and convergence of the Peaceman-Rachford ADI method for initial-boundary value problems. Math Comput 53:81–101
226. Amatore C, Oleinick A, Svir I (2005) Diffusion within nanometric and micrometric spherical-type domains by nanometric ring or pore active interfaces. Part 1: conformal mapping approach. J Electroanal Chem 575:103–123
227. Klymenko OV, Evans RG, Hardacre C, Svir IB, Compton RG (2004) Double potential step chronoamperometry at microdisk electrodes: simulating the case of unequal diffusion coefficients. J Electroanal Chem 571:211–221
228. Klymenko OV, Oleinick AI, Amatore C, Svir I (2007) Reconstruction of hydrodynamic flow profiles in a rectangular channel using electrochemical methods of analysis. Electrochim Acta 53:1100–1106
229. Svir IB, Klimenko AV, Compton RG (2000) Two approaches for digital simulation of the channel flow cell problem. Radioelek Informatika 2:29–33
230. Svir IB, Golovenko VM (2001) Simulation of the microdisc problem in spherical coordinates. Application to electrogenerated chemiluminescence. Electrochem Commun 3:11–15
231. Svir I, Oleinick A, Yunus K, Fisher AC, Wadhawan JD, Davies TJ, Compton RG (2005) Theoretical and experimental study of the ECE mechanism at microring electrodes. J Electroanal Chem 578:289–299
232. Amatore C, Oleinick A, Svir I (2008) Theoretical analysis of microscopic ohmic drop effects on steady-state and transient voltammetry at the disk microelectrode: a quasi-conformal mapping modeling and simulation. Anal Chem 80:7947–7956
233. Svir IB, Klimenko AV, Compton RG (2001) The simulation of convective diffusion transport of matter to a channel double microband electrode and its application to electrogenerated chemiluminescence. Radiotekhnika 118:92–101
234. Menshykau D, Cortina-Puig M, del Campo FJ, Muñoz FX, Compton RG (2010) Plane-recessed disk electrodes and their arrays in transient generator-collector mode: the measurement of the rate of the chemical reaction of electrochemically generated species. J Electroanal Chem 648:28–35
235. Bard AJ, Mirkin MV, Unwin PR, Wipf DO (1992) Scanning electrochemical microscopy. 12. Theory and experiment of the feedback mode with finite heterogeneous electron-transfer kinetics and arbitrary substrate size. J Phys Chem 96:1861–1868
236. Fan FRF, Kwak J, Bard AJ (1996) Single molecule electrochemistry. J Am Chem Soc 118:9669–9675
237. Macpherson JV, Unwin PR (1994) A novel approach to the study of dissolution kinetics using the scanning electrochemical microscope: theory and application to the dissolution of CuSO4.5H2O in aqueous sulfuric acid solutions. J Phys Chem 98:1704–1713
238. Macpherson JV, Unwin PR (1997) Determination of the diffusion coefficient of hydrogen in aqueous solution using single and double potential step chronoamperometry at a disk ultramicroelectrode. Anal Chem 69:2063–2069
239. Martin RD, Unwin PR (1998) Scanning electrochemical microscopy. Kinetics of chemical reactions following electron-transfer measured with the substrate-generation-tip-collection mode. J Chem Soc Faraday Trans 94:753–759
240. Pierce DT, Unwin PR, Bard AJ (1992) Scanning electrochemical microscopy. 17. Studies of enzyme-mediator kinetics for membrane- and surface-immobilized glucose-oxidase. Anal Chem 64:1795–1804
241. Stone HL (1968) Iterative solution of implicit approximations of multidimensional partial differential equations. SIAM J Numer Anal 5:530–558

242. Weinstein FG, Stone HL, Kwan TV (1969) Iterative procedure for solution of systems of parabolic and elliptic equations in three dimensions. Ind Eng Chem Fundam 8:281–287
243. Alden JA, Booth J, Compton RG, Dryfe RAW, Sanders GHW (1995) Diffusional mass transport to microband electrodes of practical geometries: a simulation study using the strongly implicit procedure. J Electroanal Chem 389:45–54
244. Alden JA, Compton RG (1996) A comparison of finite difference algorithms for the simulation of microband electrode problems with and without convective flow. J Electroanal Chem 402:1–10
245. Alden JA, Hutchinson F, Compton RG (1997) Can cyclic voltammetry at microdisc electrodes be approximately described by one-dimensional diffusion? J Phys Chem B 101:949–958
246. Alden JA, Compton RG (1997) A general method for electrochemical simulations. 1. Formulation of the strategy for two-dimensional simulations. J Phys Chem B 101:8941–8954
247. Alden JA, Feldman MA, Hill E, Prieto F, Oyama M, Coles BA, Compton RG (1998) Channel microband electrode arrays for mechanistic electrochemistry. Two -dimensional voltammetry: transport-limited currents. Anal Chem 70:1707–1720
248. Bidwell MJ, Alden JA, Compton RG (1996) Hydrodynamic voltammetry with channel microband electrodes: the simulation of voltammetric waveshapes. J Electroanal Chem 417:119–128
249. Prieto F, Oyama M, Coles BA, Alden JA, Compton RG, Okazaki S (1998) Mechanistic determination using arrays of variable sized channel microband electrodes. The oxidation of 2,3,7,8 -tetra-methoxythianthrene in the presence of pyridine in acetonitrile solution. Electroanalysis 10:685–690
250. Krylov A (1931) On the numerical solution of equations by which the frequencies of small vibrations of material systems in technical problems are determined. Izv Akad Nauk SSSR VII:491–539 [in Russian]
251. Saad Y (1996) Iterative methods for sparse linear systems. PWS Publishing Company, Boston
252. Saad Y (2003) Iterative methods for solving sparse linear systems. SIAM Publications, Philadelphia
253. Wesseling P (1992) An introduction to multigrid methods. Wiley, New York
254. Alden JA, Compton RG (1997) A general method for electrochemical simulations. 2. Application to the simulation of steady-state currents at microdisk electrodes: homogeneous and heterogeneous kinetics. J Phys Chem B 101:9606–9616
255. Bard AJ, Denuault G, Friesner RA, Dornblaser BC, Tuckerman LS (1991) Scanning electrochemical microscopy: theory and application of the transient (chronoamperometric) SECM response. Anal Chem 63:1282–1288
256. Welford PJ, Brookes BA, Climent V, Compton RG (2001) The hanging meniscus contact: geometry induced diffusional overpotential. The reduction of oxygen in dimethylsulphoxide at Au(111). J Electroanal Chem 513:8–15
257. Brandt A (1977) Multi-level adaptive solutions to boundary-value problems. Math Comput 31:333–390
258. Alden JA, Compton RG (1996) The multigrid method, MGD1: an efficient and stable approach to electrochemical modelling. The simulaton of double electrode problems. J Electroanal Chem 415:1–12
259. Zhang J (1997) Accelerated multigrid high accuracy solution of the convection-diffusion equation with high Reynolds number. Numer Methods PDEs 13:77–92
260. Zhang J (2000) A note on an accelerated high-accuracy multigrid solution of the -diffusion equation with high Reynolds number. Numer Methods PDEs 16:1–10
261. Duff IS, Reid JK (1979) Some design features of a sparse matrix code. ACM Trans Math Soft 5:18–35
262. http://hsl.rl.ac.uk/archive/hslarchive.html
263. http://www.netlib.org
264. Zlatev Z, Wasniewski J, Schaumburg K (1981) Y12M. Solution of large and sparse systems of linear algebraic equations. Documentation and subroutines. In: Goos G, Hartmanis J (eds) Lecture notes in computer science, vol 121. Springer, Berlin

265. Smith GD (1985) Numerical solution of partial differential equations, 3rd edn. Oxford University Press, Oxford
266. Britz D, Strutwolf J, Thøgersen L (2001) Investigation of some starting protocols for BDF (FIRM) in electrochemical digital simulation. J Electroanal Chem 512:119–123
267. Newman J (1966) Resistance for flow of current to a disk. J Electrochem Soc 113:501–502
268. Spiegel MR (1963) Advanced calculus. McGraw-Hill, New York
269. Morse PM, Feshbach H (1953) Methods in theoretical physics, 2 vols. McGraw-Hill, New York.
270. Sneddon IN (1966) Mixed boundary value problems in potential theory. North-Holland Publishing Co., Amsterdam
271. Verbrugge MW, Baker DR (1992) Transient diffusion and migration to a disk electrode. J Phys Chem 96:4572–4580
272. Svir IB, Oleinick AI (2001) The electrogenerated chemiluminescence kinetics at a microdisk electrode. J Electroanal Chem 499:30–38
273. Oleinick A, Amatore C, Svir I (2004) Efficient quasi-conformal map for simulation of diffusion at disk microelectrodes. Electrochem Commun 6:588–594
274. Fornberg B (1988) Generation of finite difference formulas on arbitrarily spaced grids. Math Comput 51:699–706
275. Strutwolf J, Britz D (2004) Higher-order discretisations in electrochemical digital simulation. Part 5. Application to stationary ultramicrodisk electrode simulation. J Electroanal Chem 566:15–23
276. Cutress IJ, Compton RG (2010) Theory of square, rectangular, and microband electrodes through explicit GPU simulation. J Electroanal Chem 645:159–166
277. Strutwolf J (2005) Computational study of chronoamperometry at rectangular microelectrodes. Electroanalysis 17:1547–1554
278. Woodvine HL, Terry JG, Walton AJ, Mount AR (2010) The development and characterisation of square microfabricated electrode systems. Analyst 135:1058–1065
279. Fulian Q, Williams NA, Fisher AC (1999) Computational electrochemistry: three-dimensional boundary element simulations of double electrode geometries. Electrochem Commun 1:124–127
280. Sklyar O, Treutler TH, Vlachopoulos N, Wittstock G (2005) The geometry of nanometer-sized electrodes and its influence on electrolytic currents and metal deposition processes in scanning tunnelling and scanning electrochemical microscopy. Surf Sci 597:181–195
281. Filice FP, Li MSM, Henderson JD, Ding Z (2015) Three-dimensional electrochemical functionality of an interdigitated array electrode by scanning electrochemical microscopy. J Phys Chem C 119:21473–21482
282. Kolev SD, Simons JHM, van der Linden W (1993) Mathematical modelling of the chronoamperometric response of an array of rectangular microelectrodes. Anal Chim Acta 273:71–80
283. Beriet C, Ferrigno R, Girault HH (2000) Simulation of the chronoamperometric response of a regular array of micro-disc electrodes. J Electroanal Chem 486:56–64
284. Lee HJ, Beriet C, Ferrigno R, Girault HH (2001) Cyclic voltammetry at a regular microdisc electrode array. J Electroanal Chem 502:138–145
285. Godino N, Borrisé X, Muñoz FX, del Campo FJ, Compton RG (2009) Mass transport to nanoelectrode arrays and limitations of the diffusion domain approach: theory and experiment. J Phys Chem C 113:11119–11125
286. Zoski CG, Wijesinghe M (2010) Electrochemistry at ultramicroelectrode arrays and nanoelectrode ensembles of macro- and ultramicroelectrode dimensions. Israel J Chem 50:347–359
287. Aoki K, Tokuda K, Matsuda HJ (1987) Theory of chronoamperometric curves at microband electrodes. J Electroanal Chem 225:19–32
288. Aoki K, Tokuda K, Matsuda HJ (1987) Derivation of an approximate equation for chronoamperometric curves at microband electrodes and its experimental verification. J Electroanal Chem 230:61–67
289. Kühnel W, Rademacher H-B (2007) Liouville's theorem in conformal geometry. J Math Pure Appl 88:251–260

290. Douglas J Jr, Rachford H Jr (1956) On the numerical solution of heat conduction problems in two and three space dimensions. Trans Am Math Soc 82:421–439
291. Brian PLT (1961) A finite-difference method of high-order accuracy for the solution of three-dimensional transient heat conduction problems. AIChE J 7:367–370
292. Douglas J Jr (1962) Alternating direction methods for three space variables. Numer Math 4:41–63
293. Douglas J Jr, Gunn JE (1964) A general formulation of alternating direction methods. Numer Math 6:428–453
294. Fairweather G, Mitchell AR (1965) A new alternating direction method for parabolic equations in three space variables. J Soc Ind Appl Math 13:957–965
295. Dai W, Nassar R (1998) A second-order ADI scheme for three-dimensional parabolic differential equations. Numer Methods PDEs 14:159–168
296. Thomas LH (1949) Elliptic problems in linear difference equations over a network. Watson Scientific Computing Laboratory, Columbia University, New York
297. Hairer E, Wanner G (1991) Solving ordinary differential equations II. Stiff and differential-algebraic problems. Springer, Berlin
298. Britz D, Oldham KB, Østerby O (2009) Strategies for damping the oscillations of the alternating direction implicit method of simulation of diffusion-limited chronoamperometry at disk electrodes. Electrochim Acta 54:4822–4828
299. Schultze J, Tsakova V (1999) Electrochemical microsystem technologies: from fundamental research to technical systems. Electrochim Acta 44:3605–3627
300. Suzuki H (2000) Advances in the microfabrication of electrochemical sensors and systems. Electroanalysis 12:703–715
301. Belmont C, Tercier ML, Buffle J, Fiaccabrino GC, Kondelka-Hep M (1996) Mercury-plated iridium-based microelectrode arrays for trace metals detection by voltammetry: optimum conditions and reliability. Anal Chim Acta 329:203–214
302. Arrigan DWM (2004) Nanoelectrodes, nanoelectrode arrays and their applications. Analyst 129:1157–1165
303. Aguiar FA, Gallant AJ, Rosamond MC, Rhodes A, Wood D, Kataky R (2007) Conical recessed gold microelectrode arrays produced during photolithographic methods: characterisation and causes. Electrochem Commun 9:879–885
304. Berduque A, Lanyon YH, Beni V, Herzog G, Watson YE, Rodgers K, Stam F, Alderman J, Arrigan DWM (2007) Voltammetric characterisation of silicon-based microelectrode arrays and their application to mercury-free stripping voltammetry of copper ions. Talanta 71:1022–1030
305. Lanyon YH, De Marzi G, Watson YE, Quinn AJ, Gleeson JP, Redmond G, Arrigan DWM (2007) Fabrication of nanopore array electrodes by focussed ion beam milling. Anal Chem 79:3048–3055
306. Ordeig O, del Campo J, Muñoz FX, Banks CE, Compton RG (2007) Electroanalysis utilizing amperometric microdisk electrode arrays. Electroanalysis 19:1973–1986
307. Huang XJ, O'Mahony AM, Compton RG (2009) Microelectrode arrays for electrochemistry: approaches to fabrication. Small 5:776–788
308. Freeman NJ, Sultana R, Reza N, Woodvine H, Terry JG, Walton AJ, Brady CL, Schmueser I, Mount AR (2013) Comparison of the performance of an array of nanoband electrodes with a macro electrode with similar overall area. Phys Chem Chem Phys 15:8112–8118
309. Scanlon MD, Berduque A, Strutwolf J, Arrigan DWM (2010) Flow-injection amperometry at microfabricated silicon-based $\mu$-liquid-liquid interface arrays. Electrochim Acta 55:4234–4239
310. Scanlon M, Strutwolf J, Blake A, Iacopino D, Quinn AJ, Arrigan DWM (2010) Ion-transfer electrochemistry at arrays of nanointerfaces between immiscible electrolyte solutions confined within silicon nitride nanopore membranes. Anal Chem 82:6115–6123
311. Zachek MK, Takmakov P, Park J, Wightman RM, McCarty GS (2010) Simultaneous monitoring of dopamine concentration at spatially different brain locations in vivo. Biosens Bioelectron 25:1179–1185

312. Said NAM, Twomey K, Ogurstov VI, Arrigan DWM, Herzog G (2011) Fabrication and electrochemical characterization of micro- and nanoelectrode arrays for sensor applications. J Phys: Conf Ser 307:012052
313. Fejtl M, Stett A, Nisch W, Boven KH, Möller A (2006) On micro-electrode array revival: its development, sophistication of recording, and stimulation. In: Taketani M, Baudry M (eds) Advances in network electrophysiology. Springer, New York, pp 24–37
314. Ghane-Motlagh B, Sawan M (2013) Design and implementation challenges of microelectrode arrays: a review. Mater Sci Appl 4:483–495
315. Grygoryev K, Herzog G, Jackson N, Strutwolf J, Arrigan DWM, McDermott K, Galvin P (2014) Reversible integration of microfluidic devices with microelectrode arrays for neurobiological applications. BioNanoScience 4:263–275
316. Amatore CA, Savéant JM, Tessier D (1983) Charge transfer at partially blocked surfaces: a model for the case of microscopic active and inactive sites. J Electroanal Chem 147:39–51
317. Landsberg R, Thiele R (1966) Über den Einfluss inaktiver Oberflachenbereiche auf den Diffusionsgrenzstrom an rotierenden Scheibenelektroden und die Transitionszeit be1 galvanostatischen Messungen. Electrochim Acta 11:1243–1259
318. Scheller F, Müller S, Landsberg R, Spitzer HJ (1968) Gesetzmässigkeit für den Diffusionsgrenzstrom an teilweise blockierten Modellelektroden. J Electroanal Chem 19:187–198
319. Scheller F, Landsberg R, Müller S (1969) Zur Rührabhängigkeit des Grenzstromes an teilweise bedeckten rotierenden Scheibenelektroden bei relativ grossen Umdrehungszahlen. J Electroanal Chem 20:375–381
320. Scheller F, Landsberg R, Wolf H (1970) Zur Bestimmung des Blockierungsgrades an teilweise blockierten Festelektroden. Electrochim Acta 15:525–531
321. Juozėnas A, Šidlauskas V, Jurevičius D (1993) Chronopotentiometry on partially blocked electrodes. Digital simulation. Chemija :13–17
322. Baronas R, Ivanauskas F, Survila A (2000) Simulation of electrochemical behavior of partially blocked electrodes under linear potential sweep conditions. J Math Chem 27:267–278
323. Chevallier FG, Davies TJ, Klymenko OV, Jiang L, Jones TGJ, Compton RG (2005) Numerical simulation of partially blocked electrodes under cyclic voltammetry conditions: influence of the block unit geometry on the global electrochemical properties. J Electroanal Chem 577:211–221
324. Ferrigno R, Josserand J, Brevet PF, Girault HH (1998) Coplanar interdigitated band electrodes for electrosynthesis. Part 5: Finite element simulation of paired reactions. Electrochim Acta 44:587–595
325. Slowinska K, Feldberg SW, Majda M (2003) An electrochemical time-of-flight technique with galvanostatic generation and potentiometric sensing. J Electroanal Chem 554–555:61–69
326. Svir IB, Oleinick AI, Compton RG (2003) Dual microband electrodes: current distributions and diffusion layer 'titrations'. Implications for electroanalytical measurements. J Electroanal Chem 560:117–126
327. Fosset B, Amatore CA, Bartelt JE, Wightman RM (1991) Theory and experiment for the collector-generator triple-band electrode. Anal Chem 63:1403–1408
328. Rajantie H, Strutwolf J, Williams DE (2001) Theory and practice of electrochemical titrations with dual microband electrodes. J Electroanal Chem 500:108–120
329. Seddon BJ, Girault HH, Eddowes MJ (1989) Interdigitated microband electrodes: chronoamperometry and steady state currents. J Electroanal Chem 266:227–238
330. Phillips CG, Stone HA (1997) Theoretical calculation of collection efficiencies for collector-generator microelectrode systems. J Electroanal Chem 437:157–165
331. Amatore C, Oleinick AI, Svir IB (2003) Simulation of the double hemicylinder generator-collector assembly through conformal mapping technique. J Electroanal Chem 553:49–61
332. Strutwolf J, Williams DE (2005) Electrochemical sensor design using coplanar and elevated interdigitated array electrodes. A computational study. Electroanalysis 17:169–177
333. Menshykau D, O'Mahoney AM, del Campo FJ, Munõz FX, Compton RG (2009) Microarrays of ring-recessed disk electrodes in transient generator-collector mode: theory and experiment. Anal Chem 81:9372–9382

334. Cutress IJ, Wang Y, Limon-Petersen JG, Dale SEC, Rassaei L, Compton RG (2011) Dual-microdisk electrodes in transient generator-collector mode: experiment and theory. J Electroanal Chem 655:147–153
335. Bell CG, Howell PD, Stone HA (2013) Time-dependent chronoamperometric response of dual inlaid disk electrodes. J Electroanal Chem 689:303–313
336. Paixão TRLC, Richter EM, Brito-Neto JGA, Bertotti M (2006) Fabrication of a new generator-collector electrochemical micro-device: characterization and applications. Electrochem Commun 8:9–14
337. Fiaccabrino GC, Koudelka-Hep M, Jeanneret S, van den Berg A, de Rooij NF (1994) Array of individually addressable microelectrodes. Sens Actuators B 18–19:675–677
338. Barker AL, Unwin PR, Gardner JW, Rieley H (2004) A multi-electrode probe for parallel imaging in scanning electrochemical microscopy. Electrochem Commun 6:91–97
339. Zoski CG, Yang N, He P, Berdondini L, Kouldelka-Hep M (2007) Addressable nanoelectrode membrane arrays: fabrication and steady-state behavior. Anal Chem 79:1474–1484
340. Dimaki M, Vergani M, Heiskanen A, Kwasny D, Sasso L, Carminati M, Gerrard JA, Emneus J, Svendsen WE (2014) A compact microelectrode array chip with multiple measuring sites for electrochemical applications. Sensors 14:9505–9521
341. Graaf MD, Moeller KD (2015) Introduction to microelectrode arrays, the site-selective functionalization of electrode surfaces, and the real-time detection of binding events. Langmuir 31:7697–7706
342. Chandrasekhar S (1943) Stochastic problems in physics and astronomy. Rev Mod Phys 15:1–89
343. Scharifker BR (1988) Diffusion to ensembles of microelectrodes. J Electroanal Chem 240:61–76
344. Davies TJ, Ward-Jones S, Banks CE, del Campo J, Mas R, Muñoz FX, Compton RG (2005) The cyclic and linear sweep voltammetry of regular arrays of microdisc electrodes: fitting of experimental data. J Electroanal Chem 585:51–62
345. Davies TJ, Compton RG (2005) The cyclic and linear sweep voltammetry of regular and random arrays of microdisc electrodes: theory. J Electroanal Chem 585:63–82
346. Chevallier FG, Jiang L, Jones TGJ, Compton RG (2006) Mathematical modelling and numerical simulation of cyclic voltammetry at an electrode covered with an insulating film containing cylindrical micropores. J Electroanal Chem 587:254–262
347. Belding SR, Dickinson EJF, Compton RG (2009) Diffusional cyclic voltammetry at electrodes modified with random distributions of electrocatalytic nanoparticles: theory. J Phys Chem C 113:11149–11156
348. Streeter I, Compton RG (2007) Linear sweep voltammetry at randomly distributed arrays of microband electrodes. J Phys Chem C 111:15053–15058
349. Sliusarenko O, Oleinick A, Svir I, Amatore C (2015) Development and validation of an analytical model for predicting chronoamperometric responses of random arrays of micro- and nanodisk electrodes. Chem Electrochem 2:1279–1291
350. Gueshi T, Tokuda K, Matsuda H (1978) Voltammetry at partially covered electrodes. Part I. Chronopotentiometry and chronoamperometry at model electrodes. J Electroanal Chem 89:247–260
351. Gueshi T, Tokuda K, Matsuda H (1979) Voltammetry at partially covered electrodes. Part II. Linear potential sweep and cyclic voltammetry. J Electroanal Chem 101:29–38
352. Tokuda K, Gueshi T, Matsuda H (1979) Voltammetry at partially covered electrodes. Part III. Faradaic impedance measurements at model electrodes. J Electroanal Chem 102:41–48
353. Fernández JL, Wijesinghe M, Zoski CG (2015) Theory and experiments for voltammetric and SECM investigations and application to ORR electrocatalysis at nanoelectrode ensembles of ultramicroelectrode dimensions. Anal Chem 87:1066–1074
354. Adam C, Kanoufi F, Sojic N, Etienne M (2015) Shearforce positioning of nanoprobe electrode arrays for scanning electrochemical microscopy experiments. Electrochim Acta 179:45–56
355. Reller H, Kirowa-Eisner E, Gileadi E (1982) Ensembles of microelectrodes. A digital simulation. J Electroanal Chem 138:65–77

356. Shoup D, Szabo A (1984) Chronoamperometry at an ensemble of microdisk electrodes. J Electroanal Chem 160:19–26
357. Davies TJ, Banks CE, Compton RG (2005) Voltammetry at spatially heterogeneous electrodes. J Solid State Electrochem 9:797–808
358. Ordeig O, Banks CE, Davies TJ, del Campo J, Mas R, Muñoz FX, Compton RG (2006) Regular arrays of microdisk electrodes: simulation quantifies the fraction of 'dead' electrodes. Analyst 131:440–445
359. Cutress IJ, Compton RG (2009) Stripping voltammetry at microdisk electrode arrays: theory. Electroanalysis 21:2617–2625
360. Morf WE, Koudelka-Hep M, de Rooij NF (2006) Theoretical treatment and computer simulation of microelectrode arrays. J Electroanal Chem 590:47–56
361. Strutwolf J, Scanlon MD, Arrigan DWM (2009) Electrochemical ion transfer across liquid/liquid interfaces confined within solid-state micropore arrays - simulations and experiments. Analyst 134:148–158
362. Strutwolf J, Scanlon MD, Arrigan DW (2010) The performance of differential pulse stripping voltammetry at micro-liquid-liquid interface arrays. J Electroanal Chem 641:7–13
363. Strutwolf J, Arrigan WM (2010) Optimisation of the conditions for stripping voltammetric analysis at liquid-liquid interfaces supported at micropore arrays: a computational simulation. Anal Bioanal Chem 398:1625–1631
364. Ellis JS, Strutwolf J, Arrigan DWM (2012) Finite-element simulations of the influence of pore wall adsorption on cyclic voltammetry of ion transfer across a liquid-liquid interface formed at a micropore. Phys Chem Chem Phys 14:2494–2500
365. Randles JEB (1947) Kinetics of rapid electrode processes. Discuss Faraday Soc 1:11–19
366. Ševčík A (1948) Oscillographic polarography with periodical triangular voltage. Coll Czech Chem Commun 13:349–377
367. Liu Y, Strutwolf J, Arrigan DWM (2015) Ion-transfer voltammetric behavior of propranolol at nanoscale liquid-liquid interface arrays. Anal Chem 87:4487–4494
368. Niwa O (1995) Electroanalysis with interdigitated array microelectrodes. Electroanalysis 7:606–613
369. Chidsey CE, Feldman BJ, Lundgren C, Murray RW (1986) Micrometer-spaced platinum interdigitated array electrode: fabrication, theory, and initial use. Anal Chem 58:601–607
370. Nagale MP, Fritsch I (1998) Individually addressable, submicrometer band electrode arrays. 1. Fabrication from multilayered materials. Anal Chem 70:2902–2907
371. Nagale MP, Fritsch I (1998) Individually addressable, submicrometer band electrode arrays. 2. Electrochemical characterization. Anal Chem 70:2908–2913
372. Vagin MY, Sekretaryova AN, Sanchez Reategui R, Lundstrom I, Winquist F, Eriksson M (2014) Arrays of screen-printed graphite microband electrodes as a versatile electroanalysis platform. Chem Electrochem 1:755–762
373. Aoki K, Morita M, Niwa O, Tabei H (1988) Quantitative analysis of reversible diffusion-controlled currents of redox soluble species at interdigitated array electrodes under steady-state conditions. J Electroanal Chem 256:269–282
374. Ou TY, Moldoveanu S, Anderson JL (1988) Hydrodynamic voltammetry at an interdigitated electrode array in a flow channel. Part II. Chemical reaction succeeding electron transfer. J Electroanal Chem 247:1–16
375. Seddon BJ, Girault HH, Eddowes MJ (1989) Interdigitated microband electrodes: chronoamperometry and steady state currents. J Electroanal Chem 266:227–238
376. Liu F, Kolesov G, Parkinson BA (2014) Preparation, applications, and digital simulation of carbon interdigitated array electrodes. Anal Chem 86:7391–7398
377. Streeter I, Fietkau N, del Campo J, Mas R, Mũnoz FX, Compton RG (2007) Voltammetry at regular microband electrode arrays: theory and experiment. J Phys Chem C 111:12058–12066
378. Odijk M, Olthuis W, Dam VAT, van den Berg A (2008) Simulation of redox-recycling phenomena at interdigitated array (IDA) electrodes: amplification and selectivity. Electroanalysis 20:463–468

379. Wollenberger U, Paeschke M, Hintsche R (1994) Interdigitated array microelectrodes for the determination of enzyme activities. Analyst 119:1245–1249
380. Kim SK, Hesketh PJ, Li C, Thomas JH, Halsall HB, Heineman WR (2004) Fabrication of comb interdigitated electrodes array (IDA) for a microbead-based electrochemical assay system. Biosens Bioelectron 20:887–894
381. Niwa O, Morita M, Tabei H (1990) Electrochemical behavior of reversible redox species at interdigitated array electrodes with different geometries: consideration of redox cycling and collection efficiency. Anal Chem 62:447–452
382. Aoki K, Tanaka M (1989) Time-dependence of diffusion-controlled currents at a soluble redox couple at interdigitated microarray electrodes. J Electroanal Chem 266:11–20
383. Lewis GEM, Dale SEC, Kasprzyk-Hordern B, Lubben AT, Barnes EO, Compton RG, Marken F (2014) Cavity transport effects in generator-collector electrochemical analysis of nitrobenzene. Phys Chem Chem Phys 16:18966–18973
384. Hammond JL, Gross AJ, Estrela P, Iniesta J, Green SJ, Winlove CP, Winyard PG, Benjamin N, Marken F (2014) Cysteine-cystine redox cycling in a gold-gold dual-plate generator-collector microtrench sensor. Anal Chem 86:6748–6752
385. Gross AJ, Marken F (2015) ITO-ITO dual-plate microgap electrodes: E and EC' generator-collector processes. Electroanalysis 27:1035–1042
386. Zafarani HR, Mathwig K, Sudhölter EJR, Rassaei L (2016) Electrochemical redox cycling in a new nanogap sensor: design and simulation. J Electroanal Chem 760:42–47
387. Liu F, Kolesov G, Parkinson BA (2014) Time of flight electrochemistry: diffusion coefficient measurements using interdigitated array (IDA) electrodes. J Electrochem Soc 161:H3015–H3019
388. Barnes EO, Lewis GEM, Dale SEC, Marken F, Compton RG (2012) Generator-collector double electrode systems: a review. Analyst 137:1068–1081
389. Zhang W, Stone HA, Sherwood JD (1996) Mass transfer at a microelectrode in channel flow. J Phys Chem 100:9462–9464
390. Matysik FM (1997) Voltammetric characterization of a dual-disc microelectrode in stationary solution. Electrochim Acta 42:3113–3116
391. Lewis GEM, Dale SEC, Kasprzyk-Hordern B, Barnes EO, Compton RG, Marken F (2012) Square wave electroanalysis at double hemisphere junctions. Electroanalysis 24:1726–1731
392. Rajantie H, Williams DE (2001) Electrochemical titrations of thiosulfate, sulfite, dichromate and permanganate using dual microband electrodes. Analyst 126:86–90
393. Rajantie H, Williams DE (2001) Potentiometric titrations using dual microband electrodes. Analyst 126:1882–1887
394. Ueno K, Kim HB, Kitamura N (2003) Characteristic electrochemical responses of polymer microchannel-microelectrode chips. Anal Chem 75:2086–2091
395. Amatore C, Sella C, Thouin L (2006) Electrochemical time-of-flight responses at double-band generator-collector devices under pulsed conditions. J Electroanal Chem 593:194–202
396. Kovalcik KD, Kirchhoff JR, Giolando DM, Bozon JP (2004) Copper ring-disk microelectrodes: fabrication, characterization, and application as an amperometric detector for capillary columns. Electrochim Acta 507:237–245
397. Liljeroth P, Johans C, Slevin CJ, Quinn BM, Kontturi K (2002) Disk-generation/ring-collection scanning electrochemical microscopy: theory and application. Anal Chem 74:1972–1978
398. Harvey SLR, Parker KH, O'Hare D (2007) Theoretical evaluation of the collection efficiency at ring-disc microelectrodes. J Electroanal Chem 610:122–130
399. Harvey SLR, Coxon P, Bates D, Parker KH, O'Hare D (2008) Metallic ring-disc microelectrode fabrication using inverted hollow cylindrical sputter coater. Sens Actuators B 129:659–665
400. Seddon BJ, Shao Y, Girault HH (1994) Printed microelectrode array and amperometric sensor for environmental monitoring. Electrochim Acta 39:2377–2386
401. Seddon BJ, Wang CF, Peng W, Zhang X (1995) Preparation and amperometric response of carbon and platinum dual-cylinder microelectrodes. Electrochim Acta 40:455–465

402. Amatore C, Oleinick A, Svir I (2003) Theory of transient and steady-state ECL generation at double-hemicylinder assemblies using conformal mapping and simulations. Electrochem Commun 5:989–994
403. Qiu FL, Fisher AC (2003) The boundary element method: chronoamperometric simulations at microelectrodes. Electrochem Commun 5:87–93
404. Nagy G, Sugimoto Y, Denuault G (1997) Three-dimensional random walk simulation of diffusion controlled electrode processes: (I) a hemisphere, disc and growing hemisphere. J Electroanal Chem 433:167–173
405. Baur JE, Motsegood PN (2004) Diffusional interactions at dual disk microelectrodes: comparison of experiment with three-dimensional random walk simulations. J Electroanal Chem 572:29–40
406. Cutress IJ, Dickinson EJF, Compton RG (2011) Electrochemical random walk theory. Probing voltammetry with small numbers of molecules: stochastic versus statistical (Fickian) diffusion. J Electroanal Chem 655:1–8
407. Liljeroth P, Johans C, Slevin CJ, Quinn BM, Kontturi K (2002) Micro ring-disk electrode probes for scanning electrochemical microscopy. Electrochem Commun 4:67–71
408. Amatore C, Sella C, Thouin L (2002) Diffusional cross-talk between paired microband electrodes operating within a thin film: theory for redox couples with unequal diffusion coefficients. J Phys Chem B 106:11565–11571

# Chapter 13
# Migrational Effects

Migrational, or potential field, effects are those that affect the transport of ions under the influence of an electric field in electrolytic solutions. The field can be applied externally by means of electrodes, and can also be the result of the drift of different ions with different mobilities in the solution. Thus, concentration and potential gradients are coupled [1, 2]. In electroanalysis migration is regarded as a nuisance and is usually eliminated by using excess inert electrolyte—the consensus is to use inert electrolyte at about 100 times the concentration of the electroactive substance. This ensures such weak potential effects that they can safely be ignored, and mostly are. However, if it is not possible or desirable for some reason to add the excess electrolyte, or if we are concerned with very small volumes or liquid junctions, potential fields must be taken into account. Oldham and Feldberg [3] provide six reasons why it is sometimes desirable or necessary to work with dilute or absent supporting electrolyte, possibly chief among which are electrochemistry in nonaqueous solution where the solubility of added electrolyte is low, and electroanalysis in natural waters.

Concerns with liquid junctions—that is, electrolytes with different ionic concentrations or different ionic species meeting at a junction, such as a membrane or simply a small hole in a Luggin capillary, go back at least to the works of Nernst [4, 5], Planck [6] in the 1880s and 1890, and that of Henderson [7] in 1907. It is Henderson who is credited with the derivation of the equation named after him, for the potential difference across such a junction, see below, although we find essentially the same equation in the 1890 work of Planck [6]. These works were concerned with steady state solutions. Helfferich (in 1958) [8] and Cohen and Cooley [9] computed, by finite differences, time-dependent behaviour at liquid junctions. Many subsequent works were of course published since then, including the recent work of Strutwolf et al. [10, 11], Dickinson et al. [12] and Britz and Strutwolf [13].

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_13

In the following the theory, and applications to liquid junctions and electrolysis are detailed, with three chosen examples to illustrate the simulation methods.

## 13.1 Theory

In this chapter, the symbol $E$ will denote a dimensionless potential field, because no other symbol fits this better, although it clashes with the use of $E$ elsewhere in the book.

The usual way in which diffusion/migration is introduced, for example in Sokalski et al. [14] and Compton et al. [15] (to name just two works) is by the flux density for a species $j$ in a solution containing several ionic species (considering, for the moment, a one-dimensional system), given by the Nernst–Planck equation [4–6]

$$J_j = -D_j \left( \frac{\partial c_j}{\partial x} + z_j c_j \frac{\mathcal{F}}{\mathcal{R}\mathcal{T}} \frac{\partial \phi}{\partial x} \right) \tag{13.1}$$

where $D_j$ is the diffusion coefficient of species $j$ with charge $z_j$, $\mathcal{F}$, $\mathcal{R}$ and $\mathcal{T}$ have their usual meanings and $x$ is the spatial coordinate. The concentration $c_j$ and potential $\phi$ are both time- and space-dependent. The equation can also be written in terms of the potential field, see below. One form or the other can be more convenient in particular simulations.

As well as the above, the potential must be related to the concentrations of all the ionic species, by the Poisson equation,

$$\frac{\partial^2 \phi}{\partial x^2} = -\frac{\mathcal{F}}{\epsilon_r \epsilon_0} \sum_j z_j c_j \tag{13.2}$$

in which $\epsilon_r$ is the relative permittivity of the medium (for bulk water, it is equal to 80) and $\epsilon_0$ is the permittivity of free space, equal to $8.854188 \times 10^{-12}\,\mathrm{F\,m^{-1}}$ (F denoting the Farad). The pair $\epsilon_r \epsilon_0$ thus has the value $0.71 \times 10^{-9}\,\mathrm{F\,m^{-1}}$.

If using the box method [16] or perhaps the finite volume method (also called control volume method [17, 18]), fluxes might be just what is needed. If, however, the point method is to be used, Eq. (13.1) is converted into a time-dependent concentration form, using

$$\frac{\partial c_j}{\partial t} = -\frac{\partial J_j}{\partial x} \, , \tag{13.3}$$

producing the familiar parabolic *pde*,

$$\frac{\partial c_j}{\partial t} = D \left( \frac{\partial^2 c_j}{\partial x^2} + z_j \frac{\mathcal{F}}{\mathcal{R}\mathcal{T}} \left( c_j \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial c_j}{\partial x} \frac{\partial \phi}{\partial x} \right) \right) \tag{13.4}$$

followed by the Poisson equation (13.2).

The *pde*s for the various species and the Poisson equation (13.2) specify the system and allow a solution for a given set of initial and boundary conditions.

The Poisson equation is quite straight-forward to discretise and include in a simulation, but most workers prefer simplifying it. When considering the relative magnitude of the parts of the equation, it becomes clear that significant local charge imbalance is unlikely, so most workers prefer to use what is variously termed "local charge neutrality" or the "electroneutrality condition" (ENC, to be used hereafter) and other names for it. This simply renders the Poisson equation as the approximation

$$\sum_j z_j c_j = 0 \, . \tag{13.5}$$

Much has been written about this to justify the assumption, which has been used in the earliest work on potential field effects whenever the transport of charged species is considered. Nernst [5] assumed the ENC but Planck did not and on p. 186 [6] goes to the root of the matter, estimating that the charges equalise within about $10^{-10}$ s. Strong defences of the use of ENC are seen, for example, in [9, 19–23] (to cite only a few out of many articles), but in [24–28] we note the reverse argument or [28] at least the study of the consequences of neglecting charge separation or, as Oldham terms it [20], charge disparity. Those who advocate the ENC simplification are considering problems where distances are large compared to the Debye length and time scales much larger than $10^{-9}$ s or so; these are "normal" electroanalytical applications. For problems of the formation of liquid junctions or potential fields inside nanopores, very small scales are involved, so that the ENC does not apply. Thus, Jackson [25, p. 2064] writes "…depends on the physical fact that diffusion over a Debye length takes of the order of $10^{-9}$ sec in typical laboratory situations and on the subjective fact that for a human observer $10^{-9}$ sec is an extremely small time".

Interestingly, Goldman [24] considers ENC in his theory of conditions around a membrane separating two electrolytes with different concentrations or composition, but also the further simplification of a constant potential field. This gave better results compared with his experiments. Mafé et al. have examined the conditions under which the Goldman condition is justified, and where it is not [29].

One might ask, why use the ENC at all? In a simulation, as we shall show below, we can readily discretise the Poisson equation (13.2), so why not simply do this? The answer is that for most problems, the ENC assumption simplifies the formulation of the equations in their dimensionless form and in principle allows the elimination of one variable (see below). When working, for example, on the time dependence of conditions at a liquid junction after its onset, working on very small scales, the Poisson equation must be used, whereas for simulations of electrolysis, where time- and distance scales are much larger, the ENC is appropriate and convenient.

There is yet another potential complication. Buck [30] remarks that the concentration gradient term inside the bracket in (13.1) neglects some terms. The equation considers only the spatial gradient of each $j$th species, $D_j \frac{\partial c_j}{\partial x}$, whereas, according to Onsager and Fuoss [31], there are cross-terms from the other ionic species. There has been some theory for the values of the cross-term coefficients [32–36] and Miller concludes [35] that they have a range of 0–25 % of the main term, the diffusion coefficient of the species in question. The upper end of this range, however, holds for rather concentrated electrolytes, so that it is reasonably safe to ignore these cross-terms. There has been a more recent publication on the effect of these terms [37]. In this book, they will now be ignored.

## 13.2 Simulations

Migrational problems have been simulated for a long time, and there are certain classical works that stand out and are often cited. Possibly the earliest effort was that of Helfferich [8] in 1958, simulating conditions in an ion exchanger. Cohen and Cooley simulated a liquid junction in 1965 [9], using a kind of iterative Runge–Kutta method. Scharfetter and Gummel [38] are often cited for the method they used for the drift of charged particles in a semiconductor. Interestingly, they used a block matrix method of solution, as will be described below. The work of Buck [30] was already mentioned for his comment on the Onsager terms. Brumleve and Buck [39] simulated several cases in which migration plays a role, using the backward implicit method and an available solver of the banded systems resulting from the discretisation, iterating with the Newton method (see later). They rendered the solution more efficient by reordering the banded system into a tight diagonal band system, and considered unequal time- and spatial intervals. Newman worked, among other problems, on copper deposition from a solution of copper sulphate in sulphuric acid of various concentrations [19, 40, 41], to determine how the limiting current depends on the ionic strength of the electrolyte. Myland and Oldham [21, 42] derived the ratio of the time-dependent current to that with no migration (the RPC example below) where the two charged species have equal diffusion coefficients, and Bieniasz [43] extended the derivation by not making that assumption, and followed up with a simulation study [44]. Metal deposition at a rotating disk electrode (RDE) was treated as early as 1966 by Newman [40] (also reproduced in his classic text [19, 45]), including a Fortran program. This was also studied by Yen and Chapman [46] investigating the application of orthogonal collocation.

Three examples will now be described in some detail.

## 13.3 Time Development of a Liquid Junction

There has, as mentioned above, been much interest in liquid junctions for a long time. Often only a steady state is of interest, and theoretical junction potentials were derived as early as 1890 by Planck [6] and Henderson in 1907 [7], who is usually cited for his analytical solution, which serves as a standard for comparison. A table of such potentials was provided by Smyrl and Newman [47]. Time-dependent

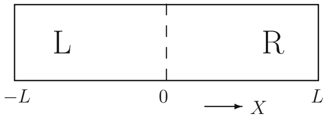


**Fig. 13.1** Stretch of solution of dimensionless length $2L$ *left* (L) and *right* (R) compartments and the junction at $X = 0$

development of the potential field across such a junction has no analytical solution as yet but there have been many simulations over the years [8–14, 39, 48–57], continuing until quite recently. The present description follows that in [13].

Figure 13.1 shows the system to be simulated. At $X = 0$ there is a junction between two electrolytes, in this example the salt AB, both singly ionised, initially at two different concentrations $C_L$ to the left of the junction and $C_R$ to the right, established at time $t = 0$. The concentrations then change, the changes spreading out from the junction with time. The spatial limits $-L$ and $+L$ will be explained below. If we denote the concentrations of $A^+$ and $B^-$, respectively as $a$ and $b$, and their diffusion coefficients as $D_A$ and $D_B$, then, applying (13.4) and (13.2) to the two ions, the time-dependent equations governing the system behaviour are

$$\begin{aligned}\frac{\partial a}{\partial t} &= D_A\left(\frac{\partial^2 a}{\partial x^2} + \frac{\mathcal{F}}{\mathcal{RT}}\left(\frac{\partial a}{\partial x}\frac{\partial \phi}{\partial x} + a\,\frac{\partial^2 \phi}{\partial x^2}\right)\right)\\ \frac{\partial b}{\partial t} &= D_B\left(\frac{\partial^2 b}{\partial x^2} - \frac{\mathcal{F}}{\mathcal{RT}}\left(\frac{\partial b}{\partial x}\frac{\partial \phi}{\partial x} + b\,\frac{\partial^2 \phi}{\partial x^2}\right)\right)\\ 0 &= \frac{\mathcal{F}}{\epsilon_r\epsilon_0}(a-b) + \frac{\partial^2 \phi}{\partial x^2}\ .\end{aligned} \tag{13.6}$$

### 13.3.1 Normalisation

Equations (13.6) and (13.2) are now normalised. The reference length most appropriate here is the Debye length, here defined as

$$L_D = \sqrt{\frac{\epsilon_r\epsilon_0 RT}{\mathcal{F}^2 c^*}}\ , \tag{13.7}$$

where $c^*$ is chosen either as the left-hand or right-hand initial concentration, preferably the smaller of the two to ensure a sufficient length. There are various factors introduced in the denominator inside the square root by various authors, such as 2 [58], $4\pi$ [59] and unity as here [50, 60, 61], and unity is most convenient in the present context, as its use does not add constants to the normalised *pde*s. The Debye length also enters into the normalisation of time. The normalisations are given by [13]

$$\begin{aligned}
A &= a/c^* \\
B &= b/c^* \\
C_L &= c_L/c^* \\
C_R &= c_R/c^* \\
\psi &= \frac{\mathcal{F}}{\mathcal{RT}}\phi \\
E &= -\frac{\mathcal{F}}{\mathcal{RT}}\frac{\partial \phi}{\partial x} \\
X &= x/L_D \\
T &= D^* t/L_D^2 \\
d_A &= D_A/D^* \\
d_B &= D_B/D^* \ .
\end{aligned} \tag{13.8}$$

The reference diffusion coefficient $D^*$ is most appropriately taken as $\sqrt{D_A\,D_B}$. This produces the new dimensionless set of *pde*s

$$\begin{aligned}
\frac{\partial A}{\partial T} &= d_A\Big(\frac{\partial^2 A}{\partial X^2} + \frac{\partial A}{\partial X}\,\frac{\partial \psi}{\partial X} + A\,\frac{\partial^2 \psi}{\partial X^2}\Big) \\
\frac{\partial B}{\partial T} &= d_B\Big(\frac{\partial^2 B}{\partial X^2} - \frac{\partial B}{\partial X}\,\frac{\partial \psi}{\partial X} - B\,\frac{\partial^2 \psi}{\partial X^2}\Big) \\
0 &= A - B + \frac{\partial^2 \psi}{\partial X^2} \ .
\end{aligned} \tag{13.9}$$

Boundary conditions are

$$\tau = 0,\ -L \le X < 0 : \quad A = B = C_L \tag{13.10a}$$

$$\tau = 0\ , 0 < X \le L : \quad A = B = C_R \tag{13.10b}$$

$$\tau = 0,\ X = 0 : \quad A = B = (C_L + C_R)/2 \tag{13.10c}$$

$$\tau = 0,\ -L \le X \le L : \quad \psi = 0 \tag{13.10d}$$

$$\tau > 0,\ X = -L : \quad A = B = C_L; \quad \psi = 0 \tag{13.10e}$$

$$\tau > 0,\ X = +L : \quad A = B = C_R; \quad \frac{\partial \psi}{\partial X} = 0\ . \tag{13.10f}$$

The length $L$ must be chosen such that there are no significant changes in concentrations at $X = \pm L$ at the maximum time $T_{max}$ to which the simulation is to be driven, that is, as usual,

$$L = 6\sqrt{T_{max}}\ . \tag{13.11}$$

If the two diffusion coefficients differ markedly, one might, on the other hand, use

$$L = 6\sqrt{d_m T_{max}} \tag{13.12}$$

with $d_m$ being the larger of the two $d_a$ and $d_b$, as was suggested by Compton et al. [15, p. 151]. Subequation (13.10c) requires comment. It seems intuitively clear that at the instant the junction is started, the concentrations at the junction itself adjust to the mean of the left-hand and right-hand bulk values. This mean value will then be set as the initial value at $X = 0$, if indeed there be a mesh point at that position. The sequence of $X$-positions on the mesh can be chosen to ensure this—and normally would be. If one feels insecure about the assumption of the mean value, the issue can be avoided by choosing a sequence such that there is a mesh point to either side of the junction but none at the junction. Then the two points closest to $X = 0$ will start at the initial bulk values, $C_L$ and $C_R$, respectively. Experiments by the present authors (unpublished) show that it makes no difference which choice is made, and the one that includes a point at $X = 0$ seems more natural.

The actual sequence of positions along $X$ should also be such that there is close spacing near the junction, because strong gradients occur there, especially at short times, but a purely exponentially expanding sequence, away from the centre, is not desirable here; in the present authors' experience, this produces inaccurate results near the extremities of the cell. Therefore, a damped exponentially expanding sequence as described in [13] and Sect.7.1 is better here. Figure 13.2 shows such a sequence. It is rather rough for visibility. It was found [13] that a small number of spatial points normally sufficient when using an exponentially expanding point sequence is used (20–40), is not enough here, and for this reason, about ten times the usual number is needed. This number—200–400—is still small in comparison with what would be needed if equal intervals were used, because of the close spacing needed near the junction.

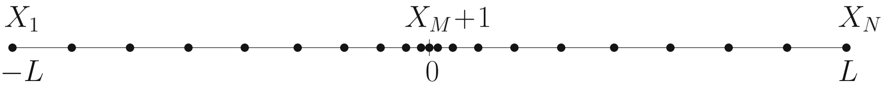


**Fig. 13.2** Point distribution with damped exponential expansion away from the centre along $X$ in the region around a liquid junction

Time, too, can be divided into expanding intervals, as it is probably at increasing time intervals that concentration, potential or potential field gradients are wanted, and here too a damped exponentially expanding sequence of $T$ values proves appropriate.

The number of points on either side of the junction is set at $M$, making the total number of points $N = 2M + 1$, so we now have a sequence of $N$ positions in $X_1, X_2, \dots X_N$ spanning the range $[-L, +L]$, see Fig. 13.2 for a coarse example, and we can discretise the three equations relevant to this system, (13.9). First of all, because varying time intervals are used it is most convenient to use backward implicit with extrapolation. BDF with unequal intervals can be done, and was, by Feldberg and Goldstein [62]. It requires a higher-order BDF, since our preference of three-point BDF leads to $O(\delta t)$ with unequal time intervals. The left-hand side of the first two equations thus becomes, using BI,

$$\frac{\partial A_i}{\partial T} \approx \frac{A'_i - A_i}{\delta T} \tag{13.13}$$

where, as usual in this book, $A'_i$ refers to the new (unknown) $A$ value at index $i$, and similarly for $B$.

There are several ways to handle this simulation, as has been detailed in [13]. If the full Poisson equation is to be used, it is probably most convenient to substitute the field $E$ for $-\partial\psi/\partial X$ in the equations, thereby eliminating one derivative. This method was called **ABE** in [13], whereas the use of $\psi$ was called **AB$\boldsymbol{\psi}$**. Both methods gave the same results. It is perhaps tempting to eliminate either species A or B from the system by substituting for it using the Poisson equation, thus reducing the system to two equations; but this proved quite inefficient in this case. In the following treatment, the potential field $E$ will be used. Then, if one wants the actual potential (relative to some point along $X$), it can be obtained by integrating $E$. The *pde* system then is

$$\begin{aligned}
\frac{\partial A}{\partial T} &= d_A\Big(\frac{\partial^2 A}{\partial X^2} - \frac{\partial A}{\partial X}E - A\,\frac{\partial E}{\partial X}\Big) \\
\frac{\partial B}{\partial T} &= d_B\Big(\frac{\partial^2 B}{\partial X^2} + \frac{\partial B}{\partial X}E + B\,\frac{\partial E}{\partial X}\Big) \\
0 &= A - B - \frac{\partial E}{\partial X}\;,
\end{aligned} \tag{13.14}$$

with the boundary conditions

$$\tau = 0,\ -L \le X < 0:\quad A = B = C_L \tag{13.15a}$$

$$\tau = 0\ ,0 < X \le L:\quad A = B = C_R \tag{13.15b}$$

$$\tau = 0,\ X = 0:\quad A = B = (C_L + C_R)/2 \tag{13.15c}$$

$$\tau > 0,\ X = -L: \quad A = B = C_L \tag{13.15d}$$

$$\tau > 0,\ X = +L: \quad A = B = C_R \tag{13.15e}$$

$$\tau > 0,\ X = \pm L: \quad E = 0 \tag{13.15f}$$

where we assume zero field at both ends.

A word about the potential is in place here. We do not know the absolute value at any point, and all we want is the potential drop across the cell. So we can, either when using the potential $\psi$ or integrating $E$, arbitrarily set $\psi$ to zero somewhere, most conveniently at the left-hand end, $X = -L$. Then the right-hand simulated value corresponds to the potential drop. The choice is not always free, as will be seen in the other example, below.

Spatial discretisations on unequally spaced points are best done using multi-point stencils, for example five-point. However, in order to keep things simple, three-point approximations are used in what follows. The symbols like $\beta(i)_k$ refer to three $\beta$ coefficients $k = 1, 2, 3$ pertaining to point $i$, used to approximate a first derivative at point $i$, and similarly for the $\alpha$ coefficients. These are all precomputed using the Fornberg algorithm [63], implemented in the subroutines `FORN` and `FORNBERG`, described in Appendix E.

At the left-hand end ($X = -L, \quad i = 0$) we have

$$\begin{aligned} A_1 &= C_L \\ B_1 &= C_L \\ E_1 &= 0\ . \end{aligned} \tag{13.16}$$

For clarity, the difference operators $\Delta$ and $\Delta^2$ are introduced, which act on $u_i$ so that

$$\begin{aligned} \Delta u_i &\equiv \beta(i)_1 u_{i-1} + \beta(i)_2 u_i + \beta(i)_3 u_{i+1} \\ \Delta^2 u_i &\equiv \alpha(i)_1 u_{i-1} + \alpha(i)_2 u_i + \alpha(i)_3 u_{i+1} \end{aligned}\ . \tag{13.17}$$

At all points $i = 2 \dots N-1$ we have directly from (13.14),

$$\begin{aligned} \frac{A'_i - A_i}{d_A \delta T} &= \Delta^2 A'_i + \Delta A'_i E_i + A'_i \Delta E'_i \\ \frac{B'_i - B_i}{d_B \delta T} &= \Delta^2 B'_i - \Delta B'_i E_i - B'_i \Delta E'_i \\ 0 &= A'_i - B'_i - \Delta E'_i\ . \end{aligned} \tag{13.18}$$

At the right-hand end, $X = +L, \quad i = N$,

$$\begin{aligned} A_N &= C_R \\ B_N &= C_R \\ E_N &= 0\ . \end{aligned} \tag{13.19}$$

This is now a $3N \times 3N$ coupled nonlinear system of equations. The unknowns are arranged in the vector $[A'_1, B'_1, E'_1, A'_2, B'_2, E'_2, \ldots A'_N, B'_N, E'_N]^T$.

In order to solve for the system, it is rearranged into the form, at index $i$,

$$\begin{aligned} F_{A,i} &= A'_{i-1}\left(\alpha(i)_1 + \beta(0)_1 E'_i\right) + A'_i\left(\alpha(i)_2 + \beta(i)_2 - \frac{1}{d_a\delta T} + \Delta E'_i\right) \\ &\qquad + A'_{i+1}\left(\alpha(i)_3 + \beta(0)_3 E'_i\right) + \frac{A_i}{d_A\delta T} \\ F_{B,i} &= B'_{i-1}\left(\alpha(i)_1 + \beta(0)_1 E'_i\right) + B'_i\left(\alpha(i)_2 + \beta(i)_2 - \frac{1}{d_B\delta T} - \Delta E'_i\right) \\ &\qquad + B'_{i+1}\left(\alpha(i)_3 - \beta(0)_3 E'_i\right) + \frac{B_i}{d_B\delta T} \\ F_{E,i} &= -\beta(i)_1 E'_{i-1} + A'_i - B'_i - \beta(i)_2 E'_i - \beta(i)_3 E'_{i+1} \end{aligned} \tag{13.20}$$

and the aim is to find values of the unknowns that render the left-hand terms equal to zero. Standard Newton iteration is used and a block-tridiagonal approach is most efficient here. We define the vectors $\mathbf{F}_i \equiv [F_{Ai}\, F_{Bi}\, F_{Ei}]^T$ and $\mathbf{U}_i \equiv [A'_i\, B'_i\, E'_i]^T$ and differentiating with respect to all three variables, obtain the $N \times N$ block system

$$\mathbf{J} = \begin{bmatrix} \mathbf{I} & & & & & \\ \mathbf{L}_2 & \mathbf{M}_2 & \mathbf{R}_2 & & & \\ & \ddots & \ddots & & & \\ & & \mathbf{L}_i\ \mathbf{M}_i & \mathbf{R}_i & & \\ & & \ddots & \ddots & & \\ & & & \mathbf{L}_{N-1} & \mathbf{M}_{N-1} & \mathbf{R}_{N-1} \\ & & & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \delta\mathbf{U}_1 \\ \delta\mathbf{U}_2 \\ \vdots \\ \delta\mathbf{U}_i \\ \vdots \\ \delta\mathbf{U}_N \end{bmatrix} = \begin{bmatrix} -\mathbf{F}_1 \\ -\mathbf{F}_2 \\ \vdots \\ -\mathbf{F}_i \\ \vdots \\ -\mathbf{F}_N \end{bmatrix}, \tag{13.21}$$

$\delta\mathbf{U}_i$ being the correction to $\mathbf{U}_i$ at each Newton step. $\mathbf{I}$ is the unit matrix and the other blocks are defined, for $2 \le i \le N-1$,

$$\mathbf{L}_i = \begin{bmatrix} \alpha(i)_1 - \beta(i)_1 E_i & 0 & -\beta(i)_1 \Delta A_i + \beta(i)_1 A_i \\ 0 & \alpha(i)_1 + \beta(i)_1 E_i & \beta(i)_1 \Delta B_i + \beta(i)_1 B_i \\ 0 & 0 & -\beta(i)_1 \end{bmatrix}$$

$$\mathbf{M}_i = \begin{bmatrix} \begin{pmatrix} \alpha(i)_2 - \beta(i)_2 E_i \\ -\Delta E_i - 1/(d_A \delta t) \end{pmatrix} & 0 & -\beta(i)_2 \Delta A_i - \beta(i)_{i2} A_i \\ 0 & \begin{pmatrix} \alpha(i)_2 + \beta(i)_2 E_i \\ +\Delta E_i - 1/(d_B \delta t) \end{pmatrix} & \beta(i)_2 \Delta B_i + \beta(i)_2 B_i \\ 1 & -1 & -\beta(i)_2 \end{bmatrix}$$

$$\mathbf{R}_i = \begin{bmatrix} \alpha(i)_3 - \beta(i)_{i3} E_i & 0 & \beta(i)_3 \delta A_i - \beta(i)_3 A_i \\ 0 & \alpha(i)_3 + \beta(i)_3 E_i & \beta(i)_3 \delta B_i + \alpha(i)_3 B_i \\ 0 & 0 & -\beta(i)_3 \end{bmatrix} . \tag{13.22}$$

The block system is solved by the block-Thomas algorithm described in Chap. 6, Sect. 6.4, adapted to the present problem. We follow the description in [13] but simplified to the case of potential field instead of potential. The difference (simplification) lies in the last row of the block matrix in (13.21), which would contain three blocks if working with potential (see (13.10f)). It goes as follows.

At the top, where we know $\delta U_1 = -F_1$ we set, for convenience,

$$\delta\mathbf{U}_1 = \mathbf{v}_1 \delta\mathbf{U}_2 + \mathbf{w}_1 \tag{13.23}$$

where clearly $\mathbf{v}_1$ is the zero matrix and $\mathbf{w}_1 = -F_1$. Row 2 expands to

$$\mathbf{L}_2 \delta\mathbf{U}_1 + \mathbf{M}_2 \delta\mathbf{U}_2 + \mathbf{R}_2 \delta\mathbf{U}_3 = -\mathbf{F}_2 \tag{13.24}$$

and substituting for $\delta\mathbf{U}_1$,

$$(\mathbf{L}_2 \mathbf{v}_1 + \mathbf{M}_2)\delta\mathbf{U}_2 + \mathbf{R}_2 \delta\mathbf{U}_3 = -(\mathbf{L}_2 \mathbf{w}_1 + \mathbf{F}_2) \tag{13.25}$$

from which we get

$$\delta\mathbf{U}_2 = \mathbf{v}_2 \delta\mathbf{U}_3 + \mathbf{w}_2 \tag{13.26}$$

where

$$\begin{aligned} \mathbf{v}_2 &= -(\mathbf{L}_2\mathbf{v}_1 + \mathbf{M}_2)^{-1}\,\mathbf{R}_2 \\ \mathbf{w}_2 &= -(\mathbf{L}_2\mathbf{v}_1 + \mathbf{M}_2)^{-1}\,(\mathbf{L}_2 w_1 + \mathbf{F}_2)\ . \end{aligned} \tag{13.27}$$

We go on doing this and in general at index $i$,

$$\delta\mathbf{U}_i = \mathbf{v}_i\delta\mathbf{U}_{i+1} + \mathbf{w}_i \tag{13.28}$$

where

$$\begin{aligned} \mathbf{v}_i &= -(\mathbf{L}_i\mathbf{v}_{i-1} + \mathbf{M}_i)^{-1}\,\mathbf{R}_i \\ \mathbf{w}_i &= -(\mathbf{L}_i\mathbf{v}_{i-1} + \mathbf{M}_i)^{-1}\,(\mathbf{L}_i\mathbf{w}_{i-1} + \mathbf{F}_i)\ . \end{aligned} \tag{13.29}$$

All $\mathbf{v}_i$ and $\mathbf{w}_i$ are stored. The process stops at $i = N - 1$, where

$$\delta\mathbf{U}_{N-1} = \mathbf{v}_{N-1}\delta\mathbf{U}_N + \mathbf{w}_{N-1} = \mathbf{w}_{N-1} \tag{13.30}$$

since $\delta\mathbf{U}_N = 0$. This yields $\delta\mathbf{U}_{N-1}$ directly, and back substitution for $i = N-2, N-3, \ldots 2$, using (13.28) then provides the solution for all $\delta\mathbf{U}_i$. These are used to correct the current value of $\mathbf{U}$, and the whole process is repeated until convergence is achieved. This can be defined in several ways: either that the norm of $\mathbf{F}$ is as small as one wishes, or the norm of $\delta\mathbf{U}$, or indeed both. If, for example, in a given iteration at the point where $\mathbf{F}$ is computed, its norm is seen to be sufficiently small, the iteration can be stopped. If not, it might be stopped after $\delta\mathbf{U}$ has been computed, if its norm is sufficiently small. In most cases, numerical experiments show that about two to three iterations are enough. One reason for this is that the changes during a single time step are quite small.

Figure 13.3 shows plots of concentration at the dimensionless times 0.1, 1, 10 and 100 for a liquid junction consisting initially of 0.1 M HCl in the left-hand cell and 1 M HCl in the right-hand cell, simulated by the above method. The relative diffusion coefficients were computed from mobilities tabled in Bard and Faulkner [64, p. 68] or directly from Newman and Thomas-Alyea [19, p. 284]. 800 points were used along $X$, distributed in the damped exponentially expanding sequence. The longest computation (for $T = 100$) took 10 s, as noted in [13], as compared with 19 min for the ABE method, solving the whole matrix directly, rather than using blocks. One could use a sparse matrix solver, but for this still somewhat modest matrix size, it would gain little, because of the overhead of such solvers. In the figure's A panel, slight separation of the concentrations are seen, and these are emphasised in the expanded plot in A1, where the curves for $[\mathrm{H}^+]$ lie to the right of the corresponding curve for $[\mathrm{Cl}^-]$. The separation is most pronounced at short times and becomes smaller as $T$ increases, so that at $T = 10$ it is hardly visible. This underlines the idea of the electric neutrality assumption, which holds for longer times (and of course equilibrium, if there is

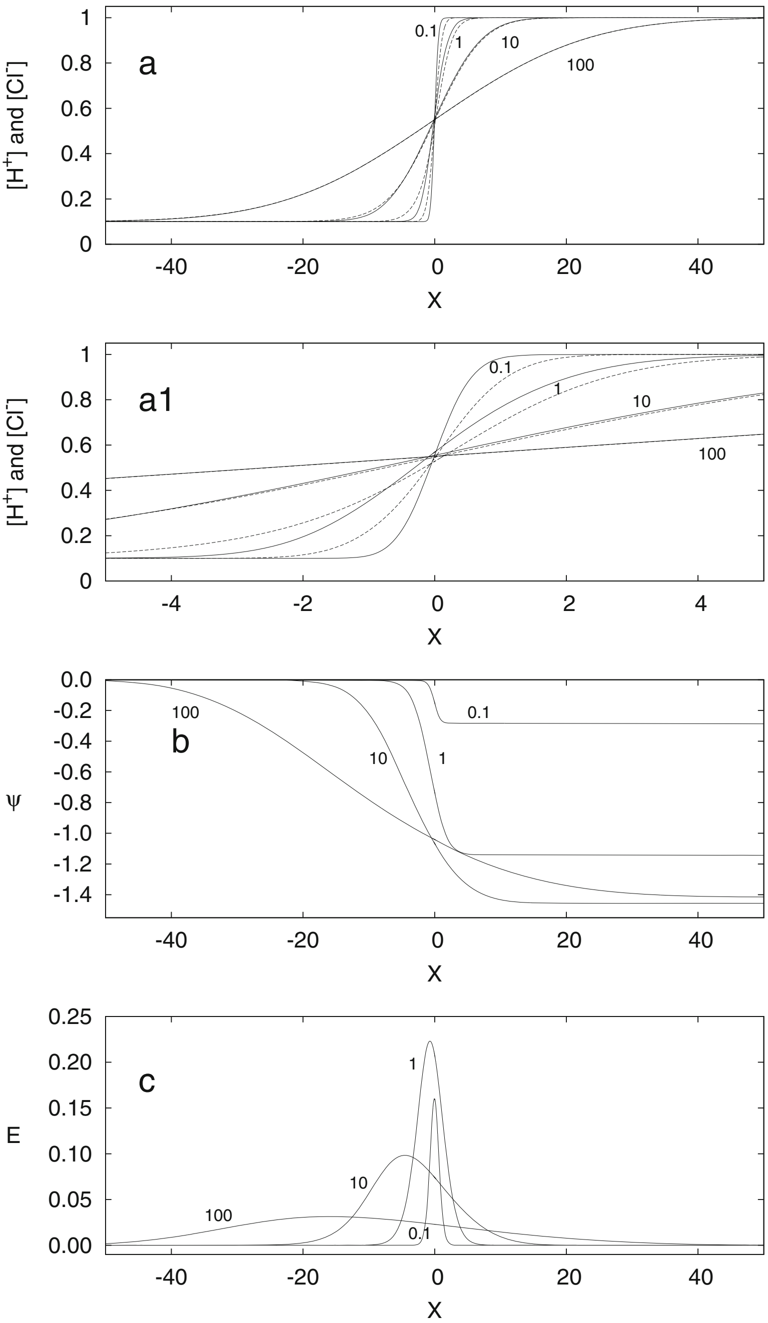


**Fig. 13.3** Profiles of concentrations along $X$ (panels **a** and **a1** , potential **b** and potential field **c** for a liquid junction of HCl, initially 0.1M in the left half and 1M in the right half, at dimensionless times $T = 0.1, 1, 10, 100$. There is noticable separation of the concentrations at times up to about 10, shown more clearly in panel **a1**. In panels **a** and **a1**, the solid line indicates $\mathrm{Cl}^{-}$ and the dashed line $\mathrm{H}^{+}$

equilibrium). Panels B and C in the figure show the potential and potential field distribution, respectively, at the four $T$ values. This example was also simulated by Dickinson et al. [12], and our own results [13] are close to (but not quite equal to) theirs, judging from their figure. The Henderson value for the potential difference over the junction is (dimensionless) $-1.48$ and our simulation computes $-1.49$.

The program can be seen in the Examples link, `LIQU_JUNC.f90`.

## 13.4 RPC Example

This follows the theory in [21, 43] and the simulation in [44]. The system to be simulated is the reversible reduction or oxidation

$$\mathrm{R} + z_R e^- \rightleftharpoons \mathrm{P} \tag{13.31}$$

with R having charge $z_R$, P being an uncharged soluble product, and there is a counterion C with charge $z_C$, opposite in sign to $z_R$. There are thus three species in solution, two of them charged, and there is the potential $\phi$ as the fourth variable. The experiment is chronoamperometry, with a potential step to electrode potential $E$. The reaction is reversible.

The concentrations of substances R, P and C are denoted as $R$, $P$ and $C$, all normalised to the initial concentration of R. The electrode potential is normalised by

$$\theta = \frac{z_R \mathcal{F}}{\mathcal{RT}}(E - E_0) \tag{13.32}$$

where $E_0$ is the equilibrium potential. Further normalisations are

$$\begin{aligned} T &= t/\tau \\ X &= x/L \\ \psi &= \frac{\mathcal{F}}{\mathcal{RT}}\phi \\ d_P &= D_P/D_R \\ d_C &= D_C/D_R \end{aligned} \tag{13.33}$$

where $\tau$ is the observation time, $L = 6\sqrt{d_m}$ (the sufficient diffusion length at $T = 1$, $d_m$ again being the larger of $d_R = 1$ and $d_C$), and the two diffusion coefficient ratios $d_P$ and $d_C$ for P and C from normalisation by the diffusion coefficient $D_R$ of R.

There are analytical solutions for the ratio of the current to that for the same system with excess electrolyte, in both [21] and in [43], the latter considering unequal diffusion coefficients. For this, and converting to our dimensionless units, the flux $j(T)$ of R is

$$j(T) = z_R \frac{d_P^{1/2} \rho}{(1 + \rho \mathrm{e}^{\theta})} \frac{1}{\sqrt{\pi T}} \tag{13.34}$$

where $\rho = \frac{1}{\sqrt{d_E\, d_P}} \left(1 - \frac{z_R}{z_C}\right)$ and $d_E = \frac{d_C(z_R - z_C)}{(z_R d_R - z_C d_C)}$ (note that both expressions contain $d_R$ but this is unity here). For pure diffusion with no migration, the dimensionless flux is [64, p. 178]

$$j_d(T) = z_R \frac{1}{(1 + \mathrm{e}^{\theta})\sqrt{\pi T}} \tag{13.35}$$

so that the ratio, as seen in [43, Fig. 1] (where $\mathrm{e}^{\theta} \to 0$), becomes

$$j/j_d = \frac{d_P^{1/2} \rho (1 + \mathrm{e}^{\theta})}{1 + \rho \mathrm{e}^{\theta}} \ . \tag{13.36}$$

This system can in fact be reduced to a single simple *pde* [43] but because it has an analytical solution it can serve as a good check on the simulation method used for such systems.

The governing equation system for the experiment is, in dimensionless form,

$$\begin{aligned}
\frac{\partial R}{\partial T} &= \frac{\partial^2 R}{\partial X^2} + z_R \left( R \frac{\partial^2 \psi}{\partial X^2} + \frac{\partial R}{\partial X} \frac{\partial \psi}{\partial X} \right) \\
\frac{1}{d_P} \frac{\partial P}{\partial T} &= \frac{\partial^2 P}{\partial X^2} \\
\frac{1}{d_C} \frac{\partial C}{\partial T} &= \frac{\partial^2 C}{\partial X^2} + z_C \left( C \frac{\partial^2 \psi}{\partial X^2} + \frac{\partial C}{\partial X} \frac{\partial \psi}{\partial X} \right) \\
0 &= z_R R + z_C C \ .
\end{aligned} \tag{13.37}$$

The spatial variable $X$ is spread over the stretch $0 : L$ in a number $N + 1$ points, in an exponentially expanding sequence, indexed by $i = 0, 1, \ldots, N$.

Boundary conditions are

$$\begin{aligned}
&T = 0,\ \text{all}\ X: \quad R = 1; \quad P = 0, \quad C = -z_C/z_R; \quad \psi = 0\\
&T > 0,\ X = 0:\\
&\qquad R - \mathrm{e}^{\theta}P = 0\\
&\qquad \left.\frac{\partial R}{\partial X}\right|_{X=0} + z_R R \left.\frac{\partial \psi}{\partial X}\right|_{X=0} + d_P \left.\frac{\partial P}{\partial X}\right|_{X=0} = 0\\
&\qquad \left.\frac{\partial C}{\partial X}\right|_{X=0} + z_C C \left.\frac{\partial \psi}{\partial X}\right|_{X=0} = 0\\
&\qquad z_R R + z_C C = 0\\
&T > 0,\ X = L: \quad R = 1; \quad P = 0, \quad C = -z_C/z_R, \quad \psi = 0\,.
\end{aligned} \tag{13.38}$$

The first condition for $T > 0$ is the Nernst equilibrium at potential $\theta$; the second expresses that the sum of fluxes of R and P is zero; the third that there is no flux in C into or out of the electrode and we have the ENC also holding at the electrode (or very close to it in solution). Note that it was necessary here to set the potential $\psi$ to zero in the bulk region ($X = L$), and to use the potential rather than the potential field $E$. In fact, one finds from the results that $E$ is level near the outer limit $X = L$, so that $E$ could have been used in the equation system, setting a zero $\partial E/\partial X$ at $X = L$ as a boundary condition. But this was not certain beforehand and could not safely be assumed.

The unknowns are taken in a large vector in the order $[R_0, P_0, C_0, \psi_0, R_1, P_1, C_1, \psi_1, \dots, R_N, P_N, C_N, \psi_N]^T$ but are lumped into the vector of four-point vectors $\mathbf{U}_i \equiv [R_i, P_i, C_i, \psi_i]^T,\ i = 0 \dots N$, to prepare for the block-tridiagonal procedure for solving the system. The system of equations (13.37) is nonlinear, and the Newton method is used to solve it. At each index $i$ we have three $4 \times 4$ blocks in the Jacobian matrix: $\mathbf{L}_i$, the left-hand block for the elements at index $i - 1$; $\mathbf{M}_i$, the middle block for index $i$, and $\mathbf{Q}_i$, the right-hand block for index $i + 1$, that symbol chosen here in order to avoid clashes with the concentration symbol $R$. They produce a tridiagonal block system. For this example, three-point BDF was used, started with one BI step.

For the discretisation expressions, it is again convenient to use the notation in Eq. (13.17) on page 347. At index $i = 0$, asymmetric one-sided formulae are used, that is, three-point forward differences.

The $F$ functions are, for $i = 0$,

$$\begin{aligned}
F_{R,0} &= R_0 - \mathrm{e}^{\theta}P_0\\
F_{P,0} &= \Delta R_0 + z_R R_0 \Delta\psi_0 + d_P \Delta P_0\\
F_{C,0} &= \Delta C_0 + z_C C_0 \Delta\psi_0\\
F_{\psi,0} &= z_R R_0 + z_C C_0\,.
\end{aligned} \tag{13.39}$$

For $i = 1, \ldots N - 1$,

$$\begin{aligned}
F_{R,i} &= \Delta^2 R_i - R_i/\delta T + z_R \left(R_i \Delta^2 \psi_i + \Delta R_i \Delta \psi_i\right) - \mathrm{RHS} \\
F_{P,i} &= \Delta^2 P_i - P_i/(d_P \delta T) - \mathrm{RHS} \\
F_{C,i} &= \Delta^2 C_i - C_i/(d_C \delta T) + z_C \left(C_i \Delta^2 \psi_i + \Delta C_i \Delta \psi_i\right) - \mathrm{RHS} \\
F\psi, i &= z_R R_i + z_C C_i
\end{aligned} \tag{13.40}$$

for the first BI step; in subsequent three-point BDF steps, the terms in $1/\delta T$ are replaced by (or, in the program, modified to) $1.5/\delta T$. The operators $\Delta$ and $\Delta^2$ were defined in (13.17) on p. 347.

At the far boundary, $i = N$, the values remain at their initial values. The right-hand side values RHS are constant through a Newton iteration series at a given time step, as they are based on old known values, and for a given variable ($v$ standing for $R$, $P$ or $C$) they are

$$\mathrm{RHS} = \begin{cases} -v_i/(d_v \delta T) & \text{(BI)} \\ ('v_i - 4v_i)/(2d_v \delta T) & \text{(BDF)} \end{cases} \tag{13.41}$$

where $'v_i$ denotes the past value, $v_i$ being the present one. For R, $d_v$ is of course equal to unity.

These equations must be differentiated wrt to all variables for the Newton iteration solving, at each iteration step, the system

$$\mathbf{J}\delta\mathbf{U} = -\mathbf{F}\,. \tag{13.42}$$

The Jacobian is a matrix of blocks, and the system to be solved at every Newton step is

$$\begin{bmatrix}
\mathbf{L}_0 & \mathbf{M}_0 & \mathbf{Q}_0 & & & & \\
\mathbf{L}_1 & \mathbf{M}_1 & \mathbf{Q}_1 & & & & \\
 & \ddots & & \ddots & & & \\
 & & \mathbf{L}_i & \mathbf{M}_i & \mathbf{Q}_i & & \\
 & & & \ddots & & \ddots & \\
 & & & & \mathbf{L}_{N-1} & \mathbf{M}_{N-1} & \mathbf{Q}_{N-1} \\
 & & & & & & I
\end{bmatrix}
\begin{bmatrix} \delta U_0 \\ \delta U_1 \\ \vdots \\ \delta U_i \\ \vdots \\ \delta U_{N-1} \\ \delta U_N \end{bmatrix}
= -
\begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_i \\ \vdots \\ F_{N-1} \\ F_N \end{bmatrix}. \tag{13.43}$$

with $\mathbf{I}$ being the $4 \times 4$ unit matrix.

The blocks are as follows. At the electrode, $i = 0$,

$$\mathbf{L}_0 = \begin{bmatrix} 1 & -\mathrm{e}^{\theta} & 0 & 0 \\ \beta_1 + z_R \Delta\psi_0 & d_P\beta_1 & 0 & z_R\beta_1 R_0 \\ 0 & 0 & \beta_1 + z_C\Delta\psi_0 & z_C\beta_1 C_0 \\ z_R & 0 & z_C & 0 \end{bmatrix};$$

$$\mathbf{M}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_2 & d_P\beta_2 & 0 & z_R\beta_2 R_0 \\ 0 & 0 & \beta_2 & z_C\beta_2 C_0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \tag{13.44}$$

$$\mathbf{Q}_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_3 & d_P\beta_3 & 0 & z_R\beta_3 R_0 \\ 0 & 0 & \beta_3 & z_C\beta_3 C_0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

and for $i = 1, \ldots N - 1$, we then have for BI,

$$\mathbf{L}_i = \begin{bmatrix} \alpha_1 + z_R\beta_1\Delta\psi_i & 0 & 0 & z_R\left(\alpha_1 R_i + \beta_1\Delta R_i\right) \\ 0 & \alpha_1 & 0 & 0 \\ 0 & 0 & \alpha_1 + z_C\beta_1\Delta\psi_i & z_C\left(\alpha_1 C_i + \beta_1\Delta C_i\right) \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{M}_i = \begin{bmatrix} \begin{pmatrix} \alpha_2 - 1/\delta T \\ + z_R(\Delta^2\psi_i + \beta_2\Delta\psi_i) \end{pmatrix} & 0 & 0 & z_R\left(\alpha_2 R_i + \beta_2\Delta R_i\right) \\ 0 & \alpha_2 - 1/(d_P\delta T) & 0 & 0 \\ 0 & 0 & \begin{pmatrix} \alpha_2 - 1/(d_C\delta T) \\ + z_C(\Delta^2\psi_i + \beta_2\Delta\psi_i) \end{pmatrix} & z_C\left(\alpha_2 C_i + \beta_2\Delta C_i\right) \\ z_R & 0 & z_C & 0 \end{bmatrix},$$

$$\mathbf{Q}_i = \begin{bmatrix} \alpha_3 + z_R\beta_3\Delta\psi_i & 0 & 0 & z_R\left(\alpha_3 R_i + \beta_3\Delta R_i\right) \\ 0 & \alpha_3 & 0 & 0 \\ 0 & 0 & \alpha_3 + z_C\beta_3\Delta\psi_i & z_C\left(\alpha_3 C_i + \beta_3\Delta C_i\right) \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{13.45}$$

and at the end point, $i = N$, there is only a single block, the unit matrix.

There is an alternative ENC expression. The ENC, last equation in (13.37), can possibly be improved, as Bieniasz describes [44], by differentiating it wrt time,

$$0 = z_R \frac{\partial R}{\partial T} + z_C \frac{\partial C}{\partial T} \tag{13.46}$$

and substituting the known time derivatives. It turns out, however, that this is no improvement as it is a simple consequence of the properties of the equation set, as pointed out by Østerby (private communication). That is, the resulting equation can be reduced to the simpler one. An implementation produces, as expected, the same results as using the simple ENC.

The block-Thomas solution is similar to that for the liquid junction example, except that here one starts at the bottom of the system, where there is no change in the variables. The system to be solved is (13.43) with the block as defined in (13.44)–(13.45).

We know $\delta\mathbf{U}_N = F_N = 0$. At row $N-1$ the equation

$$\mathbf{L}_{N-1}\boldsymbol{\delta}\mathbf{U}_{N-2} + \mathbf{M}_{N-1}\boldsymbol{\delta}\mathbf{U}_{N-1} + \mathbf{Q}_{N-1}\boldsymbol{\delta}\mathbf{U}_N = -\mathbf{F}_{N-1} \tag{13.47}$$

then becomes

$$\mathbf{L}_{N-1}\boldsymbol{\delta}\mathbf{U}_{N-2} + \mathbf{M}_{N-1}\boldsymbol{\delta}\mathbf{U}_{N-1} = -\mathbf{F}_{N-1} \tag{13.48}$$

and this allows the substitution for $\boldsymbol{\delta}\mathbf{U}_{N-1}$,

$$\boldsymbol{\delta}\mathbf{U}_{N-1} = \mathbf{V}_{N-1}\boldsymbol{\delta}\mathbf{U}_{N-2} + \mathbf{W}_{N-1} \tag{13.49}$$

with $\mathbf{V}_{N-1} = -\mathbf{M}_{N-1}^{-1}\mathbf{L}_{N-1}$ and the vector $\mathbf{W}_{N-1} = -\mathbf{M}_{N-1}^{-1}\mathbf{F}_{N-1}$. In row $N-2$ we can now substitute for $\delta\mathbf{U}_{N-1}$ and get a similar equation as (13.49), reduced to just two unknowns. In general for row $i < N-1$, we have

$$\boldsymbol{\delta}\mathbf{U}_i = \mathbf{V}_i\boldsymbol{\delta}\mathbf{U}_{i-1} + \mathbf{W}_i \tag{13.50}$$

with

$$\begin{aligned} \mathbf{V}_i &= -\left(\mathbf{M}_i + \mathbf{Q}_i\mathbf{V}_{i+1}\right)^{-1}\mathbf{L}_i \\ \mathbf{W}_i &= -\left(\mathbf{M}_i + \mathbf{Q}_i\mathbf{V}_{i+1}\right)^{-1}\left(\mathbf{F}_i + \mathbf{Q}_i\mathbf{W}_{i+1}\right) \ . \end{aligned} \tag{13.51}$$

These are computed for all $i = N-2, N-3, \ldots 1$. In the first row of (13.43), we can now substitute for $\delta\mathbf{U}_2$ in terms of $\delta\mathbf{U}_1$ and then for $\delta\mathbf{U}_1$ in terms of $\delta\mathbf{U}_0$, resulting in an explicit equation for $\delta\mathbf{U}_0$, after which all other $\delta\mathbf{U}_i, i = 1, \ldots, N-1$ can be computed, successively applying (13.50).

The computation results in the current ratio $j/j_d$ that can be checked against the known value (13.36) from [43]. The program is included in Appendix E, program `RPC.f90`.

## 13.5 Copper Deposition on an RDE

Copper is deposited on a RDE from a solution of $CuSO_4$ at concentration $c^*$, to which $H_2SO_4$ has been added, concentration $s$. Steady state, transport limited current at an RDE is assumed. There are three species in solution, and we denote their concentrations as $a, b, c$, respectively standing for $[Cu^{2+}]$, $[H^+]$, and $[SO_4^{2-}]$. Their concentrations are initially $c^*$, $2s$ and $c^* + s$, respectively. The potential is $\phi$. The three species will be indexed by $j = 1, 2, 3$, or denoted by A, B, C, respectively. These are normalised as follows:

$$\begin{aligned} A &= a/c^* \ ; \quad B = b/c^* \ ; \quad C = c/c^* \ ; \quad S = s/c^* \\ d_A &= D_A/D^* \ ; \quad d_B = D_B/D^* \ ; \quad d_C = D_C/D^* \end{aligned} \tag{13.52}$$

where the $D$'s are the diffusion coefficients of the three species, normalised to some reference diffusion coefficient, for example that of $Cu^{2+}$ itself (in which case $d_A = 1$). The potential $\phi$ is normalised as above in Eq. (13.8). The three diffusion coefficients, both dimensioned and dimensionless, are given in Table 13.1, out of the CRC Handbook of Chemistry and Physics [65].

The governing equation for a transport limited electrolysis of a species $j$ at the RDE is

$$D_j \frac{\partial^2 c_j}{\partial x^2} - v_x \frac{\partial c_j}{\partial x} + z_j D_j \frac{\mathcal{F}}{\mathcal{RT}} \left( c_j \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial c_j}{\partial x} \frac{\partial \phi}{\partial x} \right) = 0 \ , \tag{13.53}$$

where $x$ is the axial coordinate away from the disk surface and $\phi$ the potential in Volt. The charges $z_j$ are specifically $z_A = 2$, $z_B = 1$ and $z_C = -2$ for this system.

The fluid velocity normal to the electrode is

$$v_x = -0.51023 \, \nu^{-1/2} \, \omega^{3/2} \, x^2 \ , \tag{13.54}$$

$\nu$ being the kinematic viscosity in $m^2 \, s^{-1}$ and $\omega$ the rotation rate in units of $s^{-1}$. The characteristic length, the diffusion layer thickness, is given as

$$\delta_0 = 1.61166 \, D^{1/3} \nu^{1/6} \omega^{-1/2} \tag{13.55}$$

**Table 13.1** Diffusion coefficients dimensioned and dimensionless of all three species involved

| Species | $D$ ($m^2/s$) | $d$ |
|---|---|---|
| $Cu^{2+}$ | $0.714 \times 10^{-9}$ | 1 |
| $H^+$ | $9.311 \times 10^{-9}$ | 13.0 |
| $SO_4^{2-}$ | $1.065 \times 10^{-9}$ | 1.49 |

Note that they are related to the mobilities $u$ sometimes more easily found, by $D = \frac{\mathcal{RT}}{z\mathcal{F}} u$

The two constants are as given by Levich [66]. Using

$$\begin{aligned} X &= x/\delta_0 \\ V_x &= v_x\delta_0/D^* \end{aligned} \tag{13.56}$$

and (13.55) we finally obtain the normalised *pde*, expressed for species j, (standing for A, B or C above)

$$d_j\frac{\partial^2 C_j}{\partial X^2} + KX^2\frac{\partial C_j}{\partial X} + z_j d_j\left(C_j\frac{\partial^2\psi}{\partial X^2} + \frac{\partial C_j}{\partial X}\frac{\partial\psi}{\partial X}\right) = 0\ , \tag{13.57}$$

with $K = 2.13593$, obtained by lumping the expressions for $V_X$ and $\delta_0$ above.

First we simplify the equation slightly for convenience,

$$\frac{\partial^2 C_j}{\partial X^2} + K_j'(X)\frac{\partial C_j}{\partial X} + z_j\left(C_j\frac{\partial^2\psi}{\partial X^2} + \frac{\partial C_j}{\partial X}\frac{\partial\psi}{\partial X}\right) = 0\ , \tag{13.58}$$

in which $K_j'(X) = \frac{KX^2}{d_j}$ ($j$ standing for $A$, $B$ or $C$). These are precomputed, for all $X$ and $j$. Equation (13.57) must be discretised for all three species, resulting in three algebraic equations, plus one for the potential. We make use of the coefficient trios $\beta_1, \beta_2, \beta_3$ at each point along $X$, for three-point spatial first derivatives and $\alpha_1, \alpha_2, \alpha_3$ for spatial second derivatives. These too are precomputed, for each point in the sequence of points along $X$ (they are dependent on $X$ or its index $i$). At the electrode, a one-sided set of $\beta$ is needed, for the gradients there, referred to $X = 0$.

The maximum (dimensionless) distance from the electrode to be considered is set at

$$L = 5\sqrt{\text{MAX}(d_A, d_B, d_C)}\ , \tag{13.59}$$

that is, five times $\delta_0$, which is found to be sufficient here.

A number $N + 1$ points are placed along the stretch $0 : L$ in an exponentially expanding sequence, indexed by $i = 0, 1, \ldots, N$, and the discrete equation at point $i$, for points in the bulk at $i = 1, \ldots N - 1$, for species A, is

$$\begin{aligned} &\alpha_1 A_{i-1} + \alpha_2 A_i + \alpha_3 A_{i+1} + K_A'(X)\left(\beta_1 A_{i-1} + \beta_2 A_i + \beta_3 A_{i+1}\right) \\ &\quad + z_A\Big(A_i(\alpha_1\psi_{i-1} + \alpha_2\psi_i + \alpha_3\psi_{i+1}) \\ &\qquad\quad + (\beta_1 A_{i-1} + \beta_2 A_i + \beta_3 A_{i+1})\,(\beta_1\psi_{i-1} + \beta_2\psi_i + \beta_3\psi_{i+1})\Big) = 0\ ; \end{aligned} \tag{13.60}$$

for species B it is

$$\begin{aligned}
&\alpha_1 B_{i-1} + \alpha_2 B_i + \alpha_3 B_{i+1} + K'_B(X)\,(\beta_1 B_{i-1} + \beta_2 B_i + \beta_3 B_{i+1})\\
&\quad + z_B\Big(B_i(\alpha_1\psi_{i-1} + \alpha_2\psi_i + \alpha_3\psi_{i+1})\\
&\qquad\quad + (\beta_1 B_{i-1} + \beta_2 B_i + \beta_3 B_{i+1})\,(\beta_1\psi_{i-1} + \beta_2\psi_i + \beta_3\psi_{i+1})\Big) = 0
\end{aligned} \tag{13.61}$$

and for species C,

$$\begin{aligned}
&\alpha_1 C_{i-1} + \alpha_2 C_i + \alpha_3 C_{i+1} + K'_C(X)\,(\beta_1 C_{i-1} + \beta_2 C_i + \beta_3 C_{i+1})\\
&\quad + z_C\Big(C_i(\alpha_1\psi_{i-1} + \alpha_2\psi_i + \alpha_3\psi_{i+1})\\
&\qquad\quad + (\beta_1 C_{i-1} + \beta_2 C_i + \beta_3 C_{i+1})\,(\beta_1\psi_{i-1} + \beta_2\psi_i + \beta_3\psi_{i+1})\Big) = 0\,.
\end{aligned} \tag{13.62}$$

The fourth equation is the electroneutrality condition

$$z_A A_i + z_B B_i + z_C C_i = 2A_i + B_i - 2C_i = 0\,. \tag{13.63}$$

Boundary conditions are

$$\begin{aligned}
X = 0:\ & A = 0; \quad J_B = 0; \quad J_C = 0; \quad 2A_i + B_i - 2C_i = 0\\
X = L:\ & A = 1; \quad B = 2S; \quad C = 1 + S; \quad \psi = 0\,,
\end{aligned} \tag{13.64}$$

the last condition in the first line is the ENC and the last condition in the second line is set arbitrarily. The fluxes are given by

$$\begin{aligned}
J_A &= d_A\left(\left.\frac{\partial A}{\partial X}\right|_{X=0} + z_A A_0 \left.\frac{\partial\psi}{\partial X}\right|_{X=0}\right)\\
J_B &= d_B\left(\left.\frac{\partial B}{\partial X}\right|_{X=0} + z_B B_0 \left.\frac{\partial\psi}{\partial X}\right|_{X=0}\right)\\
J_C &= d_C\left(\left.\frac{\partial C}{\partial X}\right|_{X=0} + z_C C_0 \left.\frac{\partial\psi}{\partial X}\right|_{X=0}\right)\,.
\end{aligned} \tag{13.65}$$

Since $J_B$ and $J_C$ are equated to zero, their diffusion coefficients can be left out. These boundary conditions form the **F** functions at the boundaries. The other $\mathbf{F}_i$ are the left-hand sides of Eqs. (13.60)–(13.63), to be updated at every Newton step.

The unknowns are taken in a large vector in the order $[A_0, B_0, C_0, \psi_0, A_1, B_1, C_1, \psi_1, \ldots, A_N, B_N, C_N, \psi_N]^T$ but are lumped into the vector of four-point vectors $\mathbf{U}_i \equiv [A_i, B_i, C_i, \psi_i]^T,\ i = 0 \ldots N$. The system of equations (13.60)–(13.62) is nonlinear, and the Newton method is used to solve it. At each index $i$ we have three $4\times4$ blocks in the Jacobian matrix: $\mathbf{L}_i$, the left-hand block for the elements at index $i-1$; $\mathbf{M}_i$, the middle block for index $i$, and $\mathbf{R}_i$, the right-hand block for index $i+1$.

The $F$ functions, for the solution $\mathbf{F} = 0$ are in fact Eqs. (13.60)–(13.63) themselves, as the right-hand sides are all zero. At the outer boundary there are fixed concentrations, so that

$$\begin{aligned} F_{A,N} &= 1 \\ F_{B,N} &= 2S \\ F_{C,N} &= 1 + S \\ F_{\psi,N} &= 0 \end{aligned} \tag{13.66}$$

and here, $\boldsymbol{\delta}\mathbf{U}_N$ is zero. The system to be solved at every Newton step is

$$\mathbf{J}\delta\mathbf{U} = -\mathbf{F}\,. \tag{13.67}$$

or in detail

$$\begin{bmatrix} \mathbf{L}_0 & \mathbf{M}_0 & \mathbf{R}_0 & & & & \\ \mathbf{L}_1 & \mathbf{M}_1 & \mathbf{R}_1 & & & & \\ & \ddots & & \ddots & & & \\ & & \mathbf{L}_i & \mathbf{M}_i & \mathbf{R}_i & & \\ & & & \ddots & & \ddots & \\ & & & & \mathbf{L}_{N-1} & \mathbf{M}_{N-1} & \mathbf{R}_{N-1} \\ & & & & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \delta\mathbf{U}_0 \\ \delta\mathbf{U}_1 \\ \vdots \\ \delta\mathbf{U}_i \\ \vdots \\ \delta\mathbf{U}_{N-1} \\ \delta\mathbf{U}_N \end{bmatrix} = \begin{bmatrix} -F_0 \\ -F_1 \\ \vdots \\ -F_i \\ \vdots \\ -F_{N-1} \\ 0 \end{bmatrix} \tag{13.68}$$

with $\mathbf{I}$ being the $4 \times 4$ unit matrix.

Again using the notation (13.17) on page 347, and differentiating equations (13.60)–(13.62) and the boundary conditions, yields the blocks, starting at the electrode

$$\begin{aligned} \mathbf{L}_0 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \beta_1 + z_B\Delta\psi_0 & 0 & \beta_1 z_B B_0 \\ 0 & 0 & \beta_1 + z_C\Delta\psi_0 & \beta_1 z_C C_0 \\ 2 & 1 & -2 & 0 \end{bmatrix}; \\ \mathbf{M}_0 &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & \beta_2 z_B B_0 \\ 0 & 0 & \beta_2 & \beta_2 z_C C_0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \\ \mathbf{R}_0 &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \beta_3 & 0 & \beta_3 z_B B_0 \\ 0 & 0 & \beta_3 & \beta_3 z_C C_0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{13.69}$$

and for $i = 1, \dots N - 1$, we then have

$$\mathbf{L}_i = \begin{bmatrix} \begin{pmatrix} \alpha_1 + K'_A(X)\beta_1 \\ +z_A\beta_1\Delta\psi \end{pmatrix} & 0 & 0 & z_A\,(\alpha_1 A_i + \beta_1\Delta A) \\ 0 & \begin{pmatrix} \alpha_1 + K'_B(X)\beta_1 \\ +z_B\beta_1\Delta\psi \end{pmatrix} & 0 & z_B\,(\alpha_1 B_i + \beta_1\Delta B) \\ 0 & 0 & \begin{pmatrix} \alpha_1 + K'_C(X)\beta_1 \\ +z_C\beta_1\Delta\psi \end{pmatrix} & z_C\,(\alpha_1 C_i + \beta_1\Delta C) \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{M}_i = \begin{bmatrix} \begin{pmatrix} \alpha_2 + K'_A(X)\beta_2 \\ +z_A(\Delta^2\psi + \beta_2\Delta\psi) \end{pmatrix} & 0 & 0 & z_A\,(\alpha_2 A_i + \beta_2\Delta A) \\ 0 & \begin{pmatrix} \alpha_2 + K'_B(X)\beta_2 \\ +z_B(\Delta^2\psi + \beta_2\Delta\psi) \end{pmatrix} & 0 & z_B\,(\alpha_2 B_i + \beta_2\Delta B) \\ 0 & 0 & \begin{pmatrix} \alpha_2 + K'_C(X)\beta_2 \\ +z_C(\Delta^2\psi + \beta_2\Delta\psi_i) \end{pmatrix} & z_C\,(\alpha_2 C_i + \beta_2\Delta C) \\ 2 & 1 & -2 & 0 \end{bmatrix}$$

$$\mathbf{R}_i = \begin{bmatrix} \begin{pmatrix} \alpha_3 + K'_A(X)\beta_3 \\ +z_A\beta_3\Delta\psi \end{pmatrix} & 0 & 0 & z_A\,(\alpha_3 A_i + \beta_3\Delta A) \\ 0 & \begin{pmatrix} \alpha_3 + K'_B(X)\beta_3 \\ +z_B\beta_3\Delta\psi \end{pmatrix} & 0 & z_B\,(\alpha_3 B_i + \beta_3\Delta B) \\ 0 & 0 & \begin{pmatrix} \alpha_3 + K'_C(X)\beta_3 \\ +z_C\beta_3\Delta\psi \end{pmatrix} & z_C\,(\alpha_3 C_i + \beta_3\Delta C) \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{13.70}$$

At the end point, $i = N$, there is only a single block, the unit matrix. The linearised system is solved using the block-Thomas algorithm. Starting at the far end where $\delta\mathbf{U}_N = 0$, leading directly to the same substitution as in (13.49), the algorithm works its way backward up the $i$ index, ending at substitutions for $\delta\mathbf{U}_2$ and $\delta\mathbf{U}_1$, which can be applied to the very first equation,

$$\mathbf{L}_0\boldsymbol{\delta}\mathbf{U}_0 + \mathbf{M}_0\boldsymbol{\delta}\mathbf{U}_1 + \mathbf{R}_0\boldsymbol{\delta}\mathbf{U}_2 = -\mathbf{F}_0 \tag{13.71}$$

and it is solved for $\delta\mathbf{U}_0$, after which all other $\delta\mathbf{U}_i$ can be evaluated. The example program CuRDE.f90 in Appendix E does this.

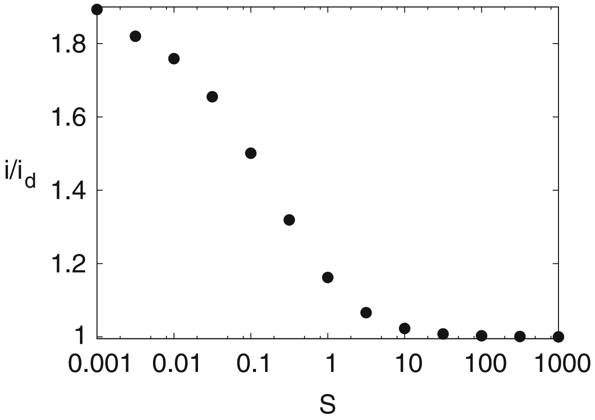


**Fig. 13.4** Ratio $i/i_d$ of the steady state copper deposition current at an RDE to the current in the presence of excess sulphuric acid, against the concentration of added sulphuric acid

Figure 13.4 shows the computed ratio $i/i_d$, that is of the current to that in the presence of excess sulphuric acid as a function of the concentration of sulphuric acid $S$. Newman [40] computed this ratio, plotting it against $\sqrt{r}$, with $r = [\mathrm{H}^+]/[\mathrm{SO}_4^{2-}]$ and stating in a following table that the theoretical value for no added sulphuric acid is 1.8852, without providing more information on this number. The figure in that publication is reproduced in the new edition of his book [19], now with a coauthor. Figure 13.5 shows the profiles along $X$ of the three species' concentrations as well as the potential $\psi$ for the simulation for $S = 1$, that is, sulphuric acid added at the same concentration as the copper sulphate itself. The plot of potential spans the $X$ range $0 \ldots 18.06$, because using (13.59), this is the result. At that $X$ value, the potential is set to zero.

### 13.5.1 Note on Normalisations

The pde (13.53) is normalised by the reference diffusion coefficient $D^*$ (i.e. $d = D/D^*$), the reference concentration $c^*$ as usual and the diffusion length $\delta_0$ as defined in (13.55) (i.e. $X = x/\delta_0$). In the first step this gives, for some unspecified species C,

$$\frac{d}{\delta_0^2}\frac{\partial^2 C}{\partial X^2} - \frac{v_x}{D^*\delta_0}\frac{\partial C}{\partial X} + \frac{zd}{\delta_0^2}\left(C\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial C}{\partial X}\frac{\partial \psi}{\partial X}\right) = 0\,, \tag{13.72}$$

and dividing by $d$ and multiplying by $\delta_0^2$,

$$\frac{\partial^2 C}{\partial X^2} - \frac{\delta_0 v_x}{d\,D^*}\frac{\partial C}{\partial X} + z\left(C\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial C}{\partial X}\frac{\partial \psi}{\partial X}\right) = 0\,, \tag{13.73}$$

leading to Eq. (13.57).

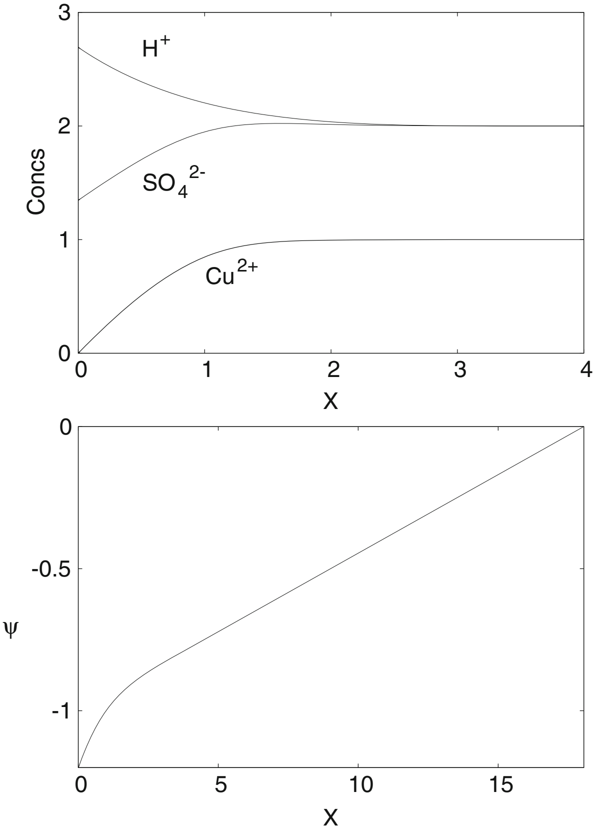


**Fig. 13.5** Profiles along $X$ of the three species' concentrations and the potential (all dimensionless) for $S = 1$

Newman [19, 40] takes a different approach. He normalises his $y$, corresponding to our $x$, as

$$\eta = y\left(\frac{a\nu}{3D^*}\right)^{1/3}\left(\frac{\omega}{\nu}\right)^{1/2} \tag{13.74}$$

and we therefore have different scalings of the distance. Yen and Chapman [46] also used this normalisation. Doing this amounts to normalising distance $y$ by $1.1199\delta_0$. The number corresponds to $1/\Gamma(4/3)$. Proceeding as before, this leads to the same normalised equation as (13.57), now with $K = 3.000$.

A simulation of (13.57) with $K = 2.13593$ and no migration produces a steady state flux equal to 1.00, whereas using $K = 3$, the flux becomes 1.12, or $1/\Gamma(4/3)$.

## References

1. Rubinstein I (1990) Electro-diffusion. SIAM studies in applied mathematics, vol 11. SIAM, Philadelphia
2. Kontturi K, Murtomäki L, Manzanares JA (2008) Ionic transport processes in electrochemistry and membrane science. Oxford UP, Oxford
3. Oldham KB, Feldberg SW (1999) Principle of unchanging total concentration and its implications for modeling unsupported transient voltammetry. J Phys Chem B 103:1699–1704
4. Nernst W (1888) Zur Kinetik der in Lösung befindlicher Körper. Z Phys Chem 2:613–637
5. Nernst W (1889) Die elektromotorische Wirksamkeit der Jonen. Z Phys Chem 4:130–188
6. Planck M (1890) Ueber die Erregung von Electricität und Wärme in Elektrolyten. Ann Phys Chem 39:161–186
7. Henderson P (1907) Zur Thermodynamik der Flüssigkeitsketten. Z Phys Chem 59:118–127
8. Helfferich F, Plesset MS (1958) J Chem Phys 28:418–424
9. Cohen H, Cooley JW (1965) The numerical solution of the time-dependent Nernst-Planck equations. Biophys J 5:145–162
10. Strutwolf J, Manning M, Arrigan DWM (2009) Investigation of potential distribution and the influence of ion complexation on diffusion potentials at aqueous-aqueous boundaries within a dual-stream microfluidic device. Anal Chem 81:8373–8379
11. Strutwolf J, Collins CJ, Adamiak W, Arrigan DWM (2010) Potentiometric investigation of protonation reactions at aqueous-aqueous boundaries within a dual-stream microfluidic structure. Langmuir 26:18526–18533
12. Dickinson EJF, Freitag L, Compton RG (2010) Dynamic theory of liquid junction potentials. J Phys Chem B 114:187–197
13. Britz D, Strutwolf J (2014) Several ways to simulate time-dependent liquid junction potentials by finite differences. Electrochim Acta 137:328–335
14. Sokalski T, Lingenfelter P, Lewenstam A (2003) Numerical solution of the coupled Nernst-Planck and Poisson equations for liquid junction and ion selective membrane potentials. Phys Chem C 107:2443–2452
15. Compton RG, Laborda E, Ward KR (2014) Understanding voltammetry: simulation of electrode processes. Imperial College Press, London
16. Feldberg SW (1969) Digital simulation: a general method for solving electrochemical diffusion-kinetic problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 3. Marcel Dekker, New York, pp 199–296
17. Patankar SV (1980) Numerical heat transfer and fluid flow. Hemisphere Publishing Corporation, New York
18. Morton KW (1996) Numerical solution of convection-diffusion problems. Chapman & Hall, London
19. Newman J, Thomas-Alyea KE (2004) Electrochemical systems, 3rd edn. Wiley, Hoboken, NJ
20. Oldham KB, Bond AM (2001) How valid is the electroneutrality approximation in the theory of steady-state voltammetry? J Electroanal Chem 508:28–40
21. Myland JC, Oldham KB (1999) Limiting currents in potentiostatic voltammetry without supporting electrolyte. Electrochem Commun 1:467–471
22. Feldberg SW (2000) On the dilemma of the use of the electroneutrality constraint in electrochemical calculations. Electrochem Commun 2:453–456
23. Smyrl WH, Newman J (1968) Potentials of cells with liquid junctions. J Phys Chem 72:4660–4671
24. Goldman DE (1943) Potential, impedance, and rectification in membranes. J Gen Physiol 27:37–60
25. Jackson JL (1974) Charge neutrality in electrolytic solutions and the liquid junction potential. J Phys Chem 78:2060–2064

26. Pellicer J, Aguilella VM, Mafé S (1986) Validity of the electroneutrality and Goldman constant-field assumptions in describing the diffusion potential for ternary electrolyte systems in simple, porous membranes. J Membr Sci 29:117–126
27. Smith CP, White HS (1993) Theory of the voltammetric response of electrodes of submicron dimensions. Violation of electroneutrality in the presence of excess supporting electrolyte. Anal Chem 65:3343–3353
28. Aguilella VM, Mafé S, Pellicer J (1987) On the nature of the diffusion potential derived from Nernst-Planck flux equations using the electroneutrality assumption. Electrochim Acta 32:483–488
29. Mafé S, Pellicer J, Aguilella VM (1986) The Goldman constant field assumption: significance and applicability conditions. Ber Bunsenges Phys Chem 90:476–479
30. Buck R (1984) Kinetics of bulk and interfacial ionic motion: microscopic bases and limits for the Nernst-Planck equation applied to membrane systems. J Membr Sci 17:1–62
31. Onsager L, Fuoss RM (1932) Irreversible processes in electrolytes. Diffusion, conductance, and viscous flow in arbitrary mixtures of strong electrolytes. J Phys Chem 36:2689–2778
32. Miller DG (1960) Thermodynamics of irreversible processes. The experimental verification of the Onsager reciprocal relations. Chem Rev 60:15–37
33. Miller DG (1960) Certain transport properties of binary electrolyte solutions and their relation to the thermodynamics of irreversible processes. J Phys Chem 64:1598–1599
34. Lorentz P (1961) The Onsager coefficient $L_{12}$ in transport of binary electrolytes. J Phys Chem 65:704
35. Miller DG (1966) Application of irreversible thermodynamics to electrolyte solutions. I. Determination of ionic transport coefficients $l_{ij}$ for isothermal vector transport processes in binary electrolyte systems. J Phys Chem 70:2639–2659
36. Pikal MJ (1971) Theory of the Onsager transport coefficients $l_{ij}$ and $R_{ij}$ for electrolyte solutions. J Phys Chem 75:3124–3134
37. Riess I (2014) How to interpret Onsager cross terms in mixed ionic electronic conductors. Phys Chem Chem Phys 16:22513–22516
38. Scharfetter DL, Gummel HK (1969) Large-signal analysis of a silicon read diode oscillator. IEEE Trans Electron Dev ED-16:64–77
39. Brumleve TR, Buck RP (1978) Numerical solution of the Nernst-Planck and Poisson equation system with applications to membrane electrochemistry and solid state physics. J Electroanal Chem 90:1–31
40. Newman J (1966) Effect of ionic migration on limiting currents. Ind Eng Chem 5:525–529
41. Newman J (1973) Electrochemical systems. Prentice-Hall, Englewood Cliffs, NJ
42. Oldham KB (2000) Steady-state voltammetry at a rotating disk electrode in the absence of supporting electrolyte. J Phys Chem B 104:4703–4706
43. Bieniasz LK (2002) Analytical formulae for chronoamperometry of a charge neutralisation process under conditions of linear migration and diffusion. Electrochem Commun 4:917–921
44. Bieniasz LK (2004) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 15: patch-adaptive simulation of example transient experiments described by Nernst-Planck-electroneutrality equations in one-dimensional space geometry. J Electroanal Chem 565:273–285
45. Newman J (1991) Electrochemical systems, 2nd edn. Prentice-Hall, Englewood Cliffs, NJ
46. Yen SC, Chapman TW (1985) Computing polarization curves for the rotating disk electrode by orthogonal collocation. Chem Eng Commun 38:159–180
47. Smyrl WH, Newman J (1968) Potentials of cells with liquid junctions. J Phys Chem 72:4660–4671
48. Hafemann D (1965) Charge separation in liquid junctions. J Phys Chem 69:4226–4231
49. Leckey JH, Horne FH (1981) Time-dependent cell potential and single-ion activity coefficients for a concentration cell with liquid junction. J Phys Chem 85:2504–2511
50. Mafé S, Pellicer J, Aguilella VM (1986) Ionic transport and space charge density in electrolytic solutions as described by Nernst-Planck and Poisson Equations. J Phys Chem 90:6045–6050

51. Manzanares JA, Murpby WD, Mafé S, Reiss H (1993) Numerical simulation of the nonequilibrium diffuse double layer in ion-exchange membranes. J Phys Chem 97:8524–8530
52. Josserand J, Lagger G, Jensen H, Ferrigno R, Girault HH (2003) Contact Galvani potential differences at liquid|liquid interfaces Part II. Contact diffusion potentials in microsystems. J Electroanal Chem 546:1–13
53. Perram JW, Stiles PJ (2006) On the nature of liquid junction and membrane potentials. Phys Chem Chem Phys 8:4200–4213
54. Morf WE, Pretsch E, De Rooij NF (2007) Computer simulation of ion-selective membrane electrodes and related systems by finite-difference procedures. J Electroanal Chem 602:43–54
55. Valent I, Neogrády P, Schreiber I, Marek M (2012) Numerical solutions of the full set of the time-dependent Nernst-Planck and Poisson equations modeling electrodiffusion in a simple ion channel. J Comput Interdiscip Sci 3:65–76
56. Valent I, Petrovič P, Neogrády P, Schreiber I, Marek M (2013) Electrodiffusion kinetics of ionic transport in a simple membrane channel. J Phys Chem B 117:14283–14293
57. Jasielec J, Filipek R, Szyszkiewicz K, Fausek J, Danielewski M, et al. (2012) Computer simulations of electrodiffusion problems based on Nernst-Planck and Poisson equations. Comput Mater Sci 63:75–90
58. Seshadri MS (1985) Current-voltage relationship for a neutral membrane. Ber Bunsenges Phys Chem 89:93–97
59. Bass L (1964) Electrical structures of interfaces in steady electrolysis. Trans Faraday Soc 60:1656–1663
60. Hickman HJ (1970) The liquid junction potential - the free diffusion junction. Chem Eng Sci 25:381–398
61. MacGillivray AD (1968) Nernst-Planck equations and the electroneutrality and Donnan equilibrium assumptions. J Chem Phys 48:2903–2906
62. Feldberg SW, Goldstein CI (1995) Examination of the behavior of the fully implicit finite-difference algorithm with the Richtmyer modification: behavior with an exponentially expanding time grid. J Electroanal Chem 397:1–10
63. Fornberg B (1988) Generation of finite difference formulas on arbitrarily spaced grids. Math Comput 51:699–706
64. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
65. Haynes WM (ed) (2015–2016) CRC handbook of chemistry and physics, 96th edn. Taylor and Francis Group, Boca Raton, USA
66. Levich VG (1962) Physicochemical hydrodynamics. Prentice-Hall, Englewood Cliffs, NJ

# Chapter 14
# Convection

Convection has long been coupled with electrochemistry, and the name hydrodynamic voltammetry has become standard. The standard work is the book by Levich [1] although he did not use that term. In electroanalytical chemistry we mainly seek reproducible conditions. These are almost always attained by systems in which a steady convective state is achieved, although not always. Thus, the once popular dropping mercury electrode (see texts such as [2, 3]) has convection around it, but is never in steady state; it might be called a reproducible periodic dynamic state.

The focus in this chapter is on channel electrodes, which are still popular and the way to simulate these illustrates the method generally.

## 14.1 Some Fluid Dynamics

Fluid flow, since it transports material, is enmeshed with diffusion in electrochemical cells. Some basics are therefore in order here.

Useful fluid dynamic systems are partially enclosed systems. Consider the open system consisting of an infinite solid plate at the bottom of a semi-infinite fluid, all at rest. Now, at $t = 0$, let the plate start moving with a certain velocity, in the direction of its plane. If one follows the fluid velocity this generates as a function of time and distance from the moving plate, it is seen that the equation governing this process is of the same form as the diffusion equation, and the solution is mathematically the same as for the Cottrell system, as is shown in the first few pages of standard texts dealing with fluid flow [4, 5]. This, then, is not a system of great use to electrochemists, since it does not lead to a steady state flow distribution. Consider now Fig. 14.1. It shows a sideways view of a channel or slit, of height $2h$, and a depth (into the paper) so great that there are assumed to be no gradients in that direction. The $y$-axis has its origin in the central plane, indicated by the dashed line.

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8_14

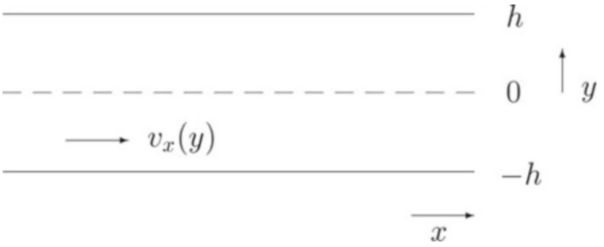


**Fig. 14.1** Flow through a channel

If we ignore entry effects at the inlet end of this channel (see below) and if the flow is laminar (that is, not turbulent), then there is a steady state flow, with no velocity components in the $y$-direction. The component $v_x$ in the $x$-direction will then be a known function of $y$. At the walls ($y = \pm h$), the fluid clings to the solid surface, that is $v_x(\pm h) = 0$. As can be shown [4, 5], the velocity profile is of the parabolic form

$$v_x(y) = v_0 \left(1 - \left(\frac{y}{h}\right)^2\right) \tag{14.1}$$

where $v_0$ is the velocity along the central plane at $y = 0$. It is the maximum velocity in the channel, and is given by

$$v_0 = \frac{h^2}{2\mu} \frac{dp}{dx} , \tag{14.2}$$

in which $\mu$ is the fluid's viscosity and $dp/dx$ is the pressure gradient driving the flow. Another quantity of interest is the mean flow velocity $v_m$ through the slit,

$$v_m = \tfrac{2}{3}\, v_0 \; . \tag{14.3}$$

The above holds only for laminar flows, that is, the Reynolds number is sufficiently small. It is defined as

$$\mathrm{Re} = \frac{v_m L}{\nu} \tag{14.4}$$

where $\nu$ is the kinematic viscosity, equal to $\mu/\rho$, $\mu$ being the fluid viscosity and $\rho$ the fluid density. $L$ is a characteristic length pertaining to the flow, here equal to $2h$, the channel height. For the flow to be laminar, the Reynolds number should be smaller than a few 1000 in such a channel. This is to some extent uncertain, depending on the smoothness of the walls and the way in which the fluid enters the slit. In fact, the phenomenon of relaminarisation of pipe and channel flow [6, 7] indicates that in theory there may not be a limit to Re for laminar flow in an ideally smooth pipe.

For electrochemical purposes, where electrodes are (usually) embedded in the channel bottom, it is convenient to shift the $y$ coordinate so that $y = 0$ at the channel bottom. The equation for the velocity profile then changes to

$$v_x(y) = v_0\left(1 - \left(\frac{y-h}{h}\right)^2\right) \tag{14.5}$$

($v_0$ of course remaining the same but now called $v_x(h)$, as in (14.2)).

Since the velocity of flow has a parabolic function, the velocity profile near the walls is nonlinear. However, in many works, this is approximated by a linearised form, as the gradient right at the walls. This makes the mathematical analysis of diffusion near one of the walls easier. Differentiating (14.1) and setting $y = -h$ (that is, considering the bottom surface), we obtain

$$\left.\frac{dv_x}{dy}\right|_{y=-h} = \frac{2v_0}{h} = \frac{h}{\mu}\frac{dp}{dx} \tag{14.6}$$

so that near the channel bottom ($y = 0$, having now moved the $y$-axis) the profile is

$$v_x(y) \approx \frac{2v_0}{h}y \tag{14.7}$$

for very small $y$. This system is currently one of the most used hydrodynamic cell types, with electrodes embedded in the surfaces.

There was intense early work on the theory of channel flow systems [8–14].

Another geometry, not quite as popular for practical reasons but perhaps equally useful, is a tube, shown in Fig. 14.2. Here the center is the axial line $r = 0$ and the walls are at $r = R$. The laminar flow for this is given by

$$v_x(r) = v_0\left(1 - \left(\frac{r}{R}\right)^2\right) \tag{14.8}$$

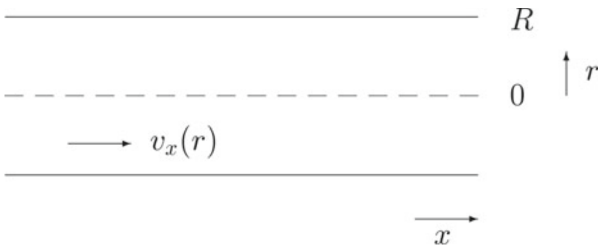


**Fig. 14.2** Flow through a cylindrical tube

where $v_0$ is the velocity along the axis, $r = 0$. It is

$$v_0 = \frac{R^2}{4\mu} \frac{dp}{dx} \tag{14.9}$$

and the average flow velocity through the tube is

$$v_m = \tfrac{1}{2} v_0 \; . \tag{14.10}$$

The Reynolds number is here defined by setting $L$ in (14.4) equal to the tube diameter, $2R$. As for the channel, Re must not exceed about 1500 (in practice) for the flow to be laminar. The velocity gradient at the wall is very similar to that for the slit (14.6), with $y$ replaced by $r$ and $h$ by $R$.

When using a tube or channel to establish a laminar flow, one must be aware of entry effects. At the entry point of the tube or channel, the flow velocity profile will be even across the cross section of the tube. As the flow moves into the tube, the profile gradually becomes parabolic. So, if we rely on its being parabolic, the electrodes must be placed sufficiently far downstream for that to be true. Kay and Nedderman [15] and Schlichting [5] both provide (almost) the same formula for this, expressed here as

$$L = 0.06\, R\, \mathrm{Re} \; , \tag{14.11}$$

so that electrodes must be at least a distance $L$ into the tube. The situation in a channel is similar, with $h$ replacing $R$, as has been shown [16]. The value of the constant depends on the extent to which the profile is to be established. At 0.06, this is about 95 % in terms of the flow velocity at the centre of the tube or channel. Prandtl and Tietjens [17] state a much larger constant, 0.6, based on a 99 % convergence to fully parabolic flow.

Another flow system that has had some use in the past is a jet impinging on a flat wall, shown in Fig. 14.3. There is a narrow jet of fluid flowing downwards out of an orifice in the top wall, hitting the bottom plate. The figure shows some flow lines, which go both in the vertical and horizontal directions. At the point P there is no flow; this is the stagnant point. Electrodes can be placed on or around this point. The flow distribution has been mathematically solved by Glauert [18], whose solutions were extended by Albery and Brett [19], with an empirical constant being provided by Yamada and Matsuda [20]. The profiles are quite complex and will not be gone into here. Other theoretical work is seen in [20–24], with Shukla and Orazem [22] considering a jet impinging on a hemisphere, which provides—unlike impingement on a disk—even material access over the electrode surface.

Other convective systems have been used in electroanalytical chemistry. The oldest one is the dropping mercury electrode [2, 3]. Convection here arises by virtue of the expansion of the growing mercury drop, and the transport equation is pleasantly simple and unidimensional for the simplified case, assuming a spherical drop that is not falling downwards, and assuming that the sphere is large compared

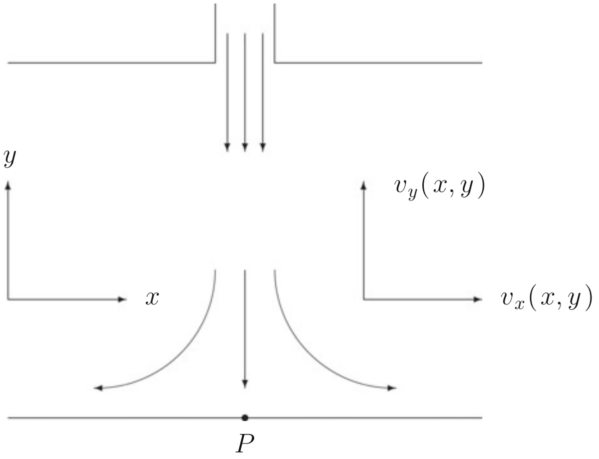


**Fig. 14.3** Flow in the wall jet system

with the diffusion later thickness. This electrode is no longer used very much and will not be further mentioned here.

Another system is the family of rotating electrodes. These are disks and/or rings mounted concentrically with the axis at the end of a cylindrical rod, rotating in an electrolyte. Making some reasonable simplifications, this system also gives rise to a unidimensional transport equation [1], and the velocity distributions around the end of the rod were first derived by von Kármán [25] and Cochran [26], later to be improved by Sparrow and Gregg [27]. Levich solved the case of steady state limiting current (see [1]), using some approximations, later corrected by Gregory and Riddiford [28]. Details can be seen in Bard and Faulkner [2].

The somewhat intractable area of natural convection has been the subject of some analysis [29–35].

## *14.1.1 Layer Relations*

In Chap. 2, the concept of the diffusion layer was established. It is a thickness, within which a large fraction of diffusional changes take place, and at a distance of several times this thickness, practically no more diffusional changes are observed. This layer will here be given the symbol $\delta_D$ ($D$ for diffusion). In fluid dynamics, there is a similar layer, within which most of the velocity changes occur. This is the hydrodynamic layer $\delta_h$. It turns out that for diffusive mass transfer, $\delta_D$ is usually much smaller than $\delta_h$. This is fortunate, because it justifies to some extent the linearised velocity profiles often assumed near walls, making analysis easier. These relations are very lucidly discussed by Vielstich in a classic paper [36].

## 14.2 Electrodes in Flow Systems

Electrodes have been placed in many flow system geometries. The rotating disk or ring-disk electrodes are well known. Narrow rings have been mounted flush inside tubes [37–40], on rotating electrodes such as the classical rotating disk electrode (RDE) mentioned above, treated by Levich [1] and the rotating ring-disk electrode (RRDE) extensively described by the Albery group [41–45] and other variants and applications, too numerous to mention here. Reviews such as that of Albery et al. [46], Penar [47] and Williams and MacPherson [48] (on modulated flows) discuss these, providing many references. Electrodes, both singly [19, 20, 49], as ring-disk [50] or (for the wall jet case) even groups of separate electrodes [23] have been mounted as targets of the flow.

Single electrodes in a flow where a steady state is attained act much like the DME or RDE, in that a sigmoid current/voltage curve is measured, from which information about the electrochemical reaction can possibly be gleaned. Heterogeneous rate constants can, for example, be measured if the flow is sufficiently fast. This was the intent of Bernstein et al. [37] and their turbulent flow in a tube with a ring electrode, and equally intense turbulence is generated at an electrode positioned close to an ultrasonic horn [51]. There are too many references of this kind to mention here. Another intent can also be a reverse of the usual electrochemical aim: so-called electrochemical probes have been used to measure flow rates [52–54].

Double electrodes were suggested, as an added ring outside the central disk on an RDE, by Frumkin et al. [50] and as a second embedded strip downstream of the first in a channel, suggested first by Gerischer et al. [11]. Here, the idea is to produce a substance by electrolysis at the upstream electrode, and to detect it at the other electrode downstream. One speaks of the collection factor $N$, the ratio of the detector current to the generator current. The symbol $N$ was first used by Frumkin et al. Much theory has been presented for the many possible geometries. A very general theory was worked out by Matsuda [12], and also, for what they called the "dimicroelectrode", by Kermiche et al. [55]. Gerischer et al. also attempted a rough first treatment [11], using (as did Kermiche et al.) a linearised velocity profile near the electrode for simplicity. Solutions are not easy to obtain in this area, so this is an intense application of digital simulation.

Another application of double, generator/collector, electrodes is what is called diffusion layer titration. This can be used for a quantitative analysis of some species in solution. The technique was first suggested by Bruckenstein and Johnson [56], and has been followed up since then, with theory [44, 57] and simulations [57–60] (naming just a selection of works).

It may be added that generator/collector cells can be implemented without convection. Double ultramicrobands [61] and interdigitated bands [62–64] have been considered for this purpose, for titrations [57, 60] and for studies in electrochemically generated chemiluminescence (ECL) [65]. See also the review by Amatore [66] with more references therein; see also Chap. 12.

## 14.3 Simulations

The earliest simulations of convective systems were those of the DME [67–70] and the RDE and RRDE. Prater and Bard performed the first simulations of the RRDE [71–73], using the explicit box method. Maloy et al. simulated ECL at an RRDE. Margarit et al. simulated a ring-ring electrode [74, 75] and studied collection factors by simulation; Clarenbach et al. [76, 77] simulated their own modification of the RRDE, also simulating fluid flow around it, as did Mandin et al. [78], using a program package. Feldberg [79] used hopscotch on an RRDE, Nolan [80] used OC. Balslev and Britz [81] used a brute force method to compute the steady state at an RDE with a complex reduction mechanism. Dan et al. [82] applied CN and what they called MDUM, a multigrid method, simulating transients at an RDE, which had been done earlier by Strutwolf [83, 84]. Gooch et al. review a number of simulation methods for hydrodynamic systems [85, 86]. This list is by no means exhaustive; only some representative examples have been cited here.

In flow systems that necessitate consideration of two-dimensional geometry, Flanagan and Marcoux's work is an early example [38]. They examined a variety of conditions, among them the importance of axial diffusion in a tube. They found that neglecting axial diffusion is justified for most flows except the slowest. This is because transport due to the flow dominates in the axial direction, and this holds for electrode lengths that are small compared with the tube radius. This is often called the Levich approximation. Levich [1] related the diffusion layer thickness, which is a function of distance along the electrode and flow velocity, to the tube radius. The condition can then be reduced to the condition

$$\frac{xD}{v_0 R^2} \ll 1 \tag{14.12}$$

which limits the length of the electrode along the length coordinate $x$. This is referred to in Wu [40] and the same condition was given by the early simulation papers of Albery and coworkers [87, 88], who however do not cite Levich. Albery et al. [87] are interesting in that they present an early finite difference simulation in this context, and use some coordinate stretching by transformation as well.

Among the many papers written on the simulation of band electrodes embedded in a channel flow system, Anderson and Marcoux [89] was the first, followed by the early work by Compton and coworkers [90–93]. Anderson and Marcoux simulated a single band in a channel, and experimented with the explicit, trapezoidal (CN) and BI techniques. They concluded that BI is probably the best. This has since been the most used technique in the papers to follow. There are good reasons for this, as outlined clearly by Fletcher [94] and Strikwerda [95]. Fletcher shows that discretisation of convective transport yields stable forms only if what fluid dynamicists call upwinding is used. This amounts to backwards implicit, in the $x$ direction (along the flow), rather than, as in previous chapters, in the time direction. Other algorithms may however be better, as shown by Alhunaizi [96], who considers a certain high-order explicit method the best.

Most of the flow systems used in these channel experiments attain steady state, and, as will be seen below, the $x$ direction can take the place of time. There is a multitude of works on the simulation of the channel electrode system, dominated to a large extent by the Compton school but many others have simulated these systems, and this work continues to the present [65, 85, 97–141], of which [100–103, 115, 121, 122] were published in the last 10 years or so.

Tube electrodes have inspired simulations since 1974 [38, 46, 87, 88, 107, 142–144], and the wall-jet or impinging jet electrode, following the theory works mentioned above, has been simulated by many [46, 91, 145–158].

## 14.4 A Simple Example: The Band Electrode in a Channel Flow

There is clearly a large number of hydrodynamic systems of interest, and the focus here is on the channel flow system with a single narrow band embedded in the channel floor. The ways to discretise this system point the way to other systems. Consider Fig. 14.4. We want the current over a short band in a low wide channel, as shown in the figure. The channel is of height $2h$ and the band has width $l$ in the $x$-direction, the direction of flow. The band's length (into the paper) is $a$, and it is assumed that $a \gg l$. The flow is laminar with velocity $v_x(y)$, a function of $y$ and is given by (14.5). The mean flow velocity is given by (14.3). If the flow is sufficiently fast (as is assumed), then we can ignore diffusion in the direction of the flow, as the flow will dominate transport. The transport equation is then

$$\frac{\partial c}{\partial t} = D\,\frac{\partial^2 c}{\partial y^2} - v_x\,\frac{\partial c}{\partial x}\,. \tag{14.13}$$

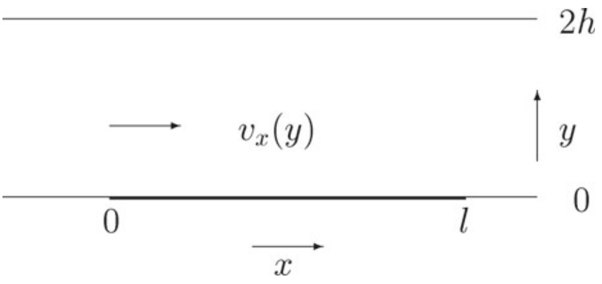


**Fig. 14.4** Band electrode in a rectangular channel

## 14.5 Normalisations

The following normalisations are used. The characteristic time $\tau$ is chosen as the time it takes the mean flow $v_m$ to traverse the length of the electrode, $l$. Thus,

$$\tau = \frac{l}{v_m} \tag{14.14}$$

and so time becomes the normalised $T$,

$$T = t/\tau = \frac{v_m}{l}\, t\,. \tag{14.15}$$

Distances are normalised by $l$, so

$$X = x/l\,, \tag{14.16}$$

$$Y = y/l\,, \tag{14.17}$$

$$H = h/l\,. \tag{14.18}$$

Concentrations are referred to the initial (upstream) bulk value $c_b$,

$$C = c/c_b\,. \tag{14.19}$$

This leads to the new transport equation,

$$\frac{\partial C}{\partial T} = \frac{1}{\text{Pe}}\,\frac{\partial^2 C}{\partial Y^2} - V_X\,\frac{\partial C}{\partial X}\,. \tag{14.20}$$

with

$$V_X = \frac{3}{2}\left\{1 - \left(\frac{Y-H}{H}\right)^2\right\}\,. \tag{14.21}$$

and

$$\text{Pe} = \frac{v_m\, l}{D}\,. \tag{14.22}$$

Pe is the Péclet number, analogous to the Reynolds number, which is defined for this flow as

$$\text{Re} = \frac{2 v_m h}{\nu}\,. \tag{14.23}$$

Whereas Re is the measure of the relative magnitudes of the inertial forces to the viscous forces in the flow, the Péclet number is the measure of the relative magnitudes of transport by convection and diffusion (citing Kay [15]). The length scales used for the two numbers are different (the channel height $2h$ for Re, $l$ for Pe).

We need to know how far away, normal to the electrode, we must compute concentration changes, that is, what value of $Y = Y_{max}$ is sufficient. This can be estimated in the following manner. The mean flow goes past the electrode in time $\tau$, and in that time, a diffusion layer of height about equal to $\sqrt{D\tau}$ can be attained at the downstream end of the electrode. Taking, as usual, six times this length, we get a maximum $y_m$ of $6\sqrt{D\tau}$. Normalising this by $l$ so that $Y_M = y_m/l$ and substituting for $\tau$ from (14.14), we have

$$Y_{max} = 6\sqrt{\frac{D}{lv_m}} = 6/\sqrt{\text{Pe}}\,. \tag{14.24}$$

We now have two situations, with two different boundary conditions. If $2H < 6/\sqrt{\text{Pe}}$, then $Y_{max}$ must be set equal to $2H$, and we apply a no-flux condition to the channel roof. If however $2H > 6/\sqrt{\text{Pe}}$, then we can apply the constant concentration (bulk value) to the level $Y_{max}$.

Figure 14.5 shows a rather coarse grid drawn on the system, for the case $2H > 6/\sqrt{\text{Pe}}$. If one wishes only to compute the current, then points downstream from the electrode need not be computed. If concentrations downstream are of interest, the grid must be extended in that direction. The range in direction $X$ is divided into $N_X$ intervals spaced apart by $\delta X$, starting from the left-hand boundary one interval upstream of the leading electrode edge. The vertical direction is divided into $N_Y + 1$ horizontal lines spaced apart by $\delta Y$. Giving indices $j$ and $i$, respectively, to the $X$ and $Y$ direction, we have a working grid as shown in Fig. 14.6. Note that there are two independent dimensionless variables that must be stated for a given simulation. They are the Péclet number Pe and $H$, the half-height of the channel in electrode length units $l$.

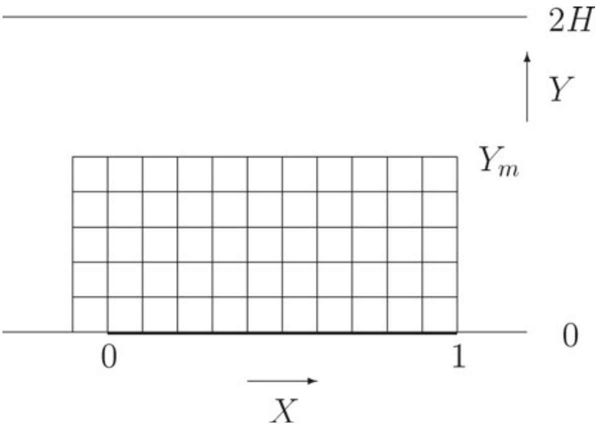


**Fig. 14.5** Band electrode system with grid superimposed

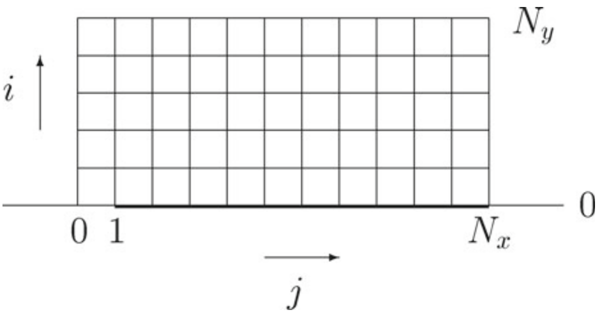


**Fig. 14.6** Band electrode system with indexed grid

We have two situations. The simpler one is that we only want the steady state current, in which case the time derivative drops out of (14.20), leaving only

$$V_X \frac{\partial C}{\partial X} = \frac{1}{\text{Pe}} \frac{\partial^2 C}{\partial Y^2} \,. \tag{14.25}$$

Boundary conditions are then

$$\begin{aligned} X < 0 &: \; C = 1 \\ 0 \le X \le 1, \; Y = 0 &: \; C = 0 \end{aligned} \tag{14.26}$$

$$Y = Y_{max} : \begin{cases} C = 1 & (2H \ge 6/\sqrt{\text{Pe}}) \\ \frac{\partial C}{\partial Y} = 0 & (2H < 6/\sqrt{\text{Pe}}) \end{cases} \,. \tag{14.27}$$

Note that (14.25) is a parabolic *pde*, there being a term on the left-hand side in $\partial C/\partial X$. This suggests a solution analogous to a time march, that is, an $X$-march, starting from "initial conditions" at the grid line just upstream of the electrode, at $j = 0$ in Fig. 14.6, and moving to the right from there, successively computing each vertical row of points. This was done by Anderson and Moldoveanu [89]. As mentioned earlier, it can be shown that the left-hand term in (14.25) is best discretised as a backward difference (upwinding). It is tempting to apply a central difference form here but this causes oscillations [94, 95].

Equation (14.25) is now discretised in the following manner,

$$V_X \frac{C_{i,j} - C_{i,j-1}}{\delta X} = \frac{1}{\text{Pe}} \frac{C_{i-1,j} - 2C_{i,j} + C_{i+1,j}}{\delta Y^2} \,. \tag{14.28}$$

Writing

$$\lambda_i = \frac{\delta X}{V_X \delta Y^2 \text{Pe}} \tag{14.29}$$

(recall that $V_X$ is a function of $Y$ and thus varies with index $i$), we have the discrete form

$$C_{i,j} - C_{i,j-1} = \lambda_i \left(C_{i-1,j} - 2C_{i,j} + C_{i+1,j}\right) \tag{14.30}$$

which is of the form seen in the time-march procedure in Chap. 8 (albeit simpler because here, equal intervals in $Y$ were used), and the solution, by the Thomas algorithm, described in Sect. 8.3, can be used.

The dimensionless current is then given by the integral over the length of the electrode,

$$I = \int_0^1 \left.\frac{\partial C}{\partial Y}\right|_{Y=0} \mathrm{d}X \tag{14.31}$$

which can be implemented by the trapezium or Simpson's rule.

The example program CHANNEL_BAND (Appendix E) is a simple implementation of the above, using two-point upwinding and equal intervals. The system has a known solution, by Levich [1, 11, 159, 160] which, normalised, implies that the current should be proportional to $\mathrm{Pe}^{\frac{1}{3}}$. This is tested in the program, and found, for a number of runs, to be true for some ranges of parameters.

If time dependence is desired, then the full transport equation (14.20) must be discretised, and a time-march performed. The problem is then a 2D one. As in the previous chapter on 2D systems, one would spread the grid points into one long vector of unknowns, either by stacking the horizontal grid lines end on end, starting (as for the UMDE system in the previous chapter) with the bottom row, or perhaps the verticals. Given the above description and those in Chap. 12, the development of this is straight-forward and will not be further pursued here.

A remark on the linearisation of the velocity profile $V_X$ is in order. It clearly applies only for $Y_M \ll H$. Given that one would in any case pre-compute the $N_y$ $V_X$ values, there seems little point in linearisation as in (14.7). This is of greater interest in mathematical analyses but not for simulations.

The procedure as described above, both for the steady state and the time-dependent system, can be extended to the channel with two bands, in generator-collector mode, as shown in Fig. 14.7. There are more boundary conditions, but

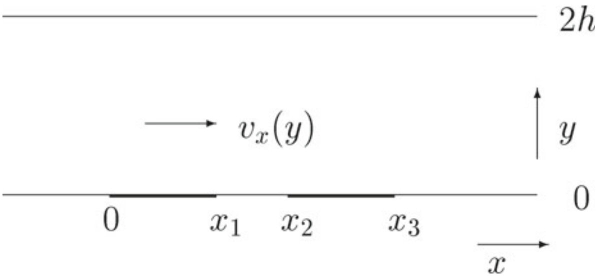


**Fig. 14.7** Channel flow with two bands

they are straight-forward to apply. For details, the reader is referred to a series of articles by the Compton group [90, 92, 110, 112, 135, 161] and more recent works by others [107, 115, 121, 122] (citing just a selection of a large opus).

## References

1. Levich VG (1962) Physicochemical hydrodynamics. Prentice-Hall, Englewood Cliffs, NJ
2. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
3. Galus Z (1994) Fundamentals of electrochemical analysis, 2nd edn. Ellis Horwood, New York (trans: Chalmers RA, Bryce WAJ (eds))
4. Bird RB, Stewart WE, Lightfoot EN (1960) Transport phenomena. Wiley, New York
5. Schlichting H (1965) Grenzschicht-Theorie. G. Braun, Karlsruhe
6. Sreenivasan KR (1982) Laminarescent, relaminarizing and retransitional flows. Acta Mech 44:1–48
7. Willis AP, Kerswell RR (2007) Critical behavior in the relaminarization of localized turbulence in pipe flow. Phys Rev Lett 98:014501-1–014501-4
8. Aoki K, Tokuda K, Matsuda H (1977) Hydrodynamic voltammetry at channel electrodes. Part I. Catalytic currents. J Electroanal Chem 76:217–233
9. Aoki K, Tokuda K, Matsuda H (1977) Hydrodynamic voltammetry at channel electrodes. Part II. Theory of first-order kinetic collection efficiencies. J Electroanal Chem 79:49–78
10. Cope DK, Tallman DE (1986) Calculation of convective-diffusion current at multiple strip electrodes in a rectangular flow channel. Implications for electrochemical detection. J Electroanal Chem 205:101–123
11. Gerischer H, Mattes I, Braun R (1965) Elektrolyse im Strömungskanal. Ein Verfahren zur Untersuchung von Reaktions- und Zwischenprodukten. J Electroanal Chem 10:553–567
12. Matsuda H (1968) Zur Theorie der Elektrolyse mit zwei eng benachbarten Elektroden in Strömungsanordnungen. Allgemeine Formel für die Übertragungsausbeute. J Electroanal Chem 16:153–164
13. Tokuda K, Aoki K, Matsuda H (1977) Hydrodynamic voltammetry at channel electrodes. Part III. Theory of kinetic currents. J Electroanal Chem 80:211–222
14. Zhang W, Stone HA, Sherwood JD (1996) Mass transfer at a microelectrode in channel flow. J Phys Chem 100:9462–9464
15. Kay JM, Nedderman RM (1974) An introduction to fluid mechanics and heat transfer, 3rd edn. Cambridge University Press, Cambridge
16. Schmidt FW, Zeldin B (1969) Laminar flows in inlet sections of tubes and ducts. AIChE J 15:612–614
17. Prandtl L, Tietjens OG (1934) Applied hydro- and aerodynamics. Dover, New York
18. Glauert MB (1956) The wall jet. J Fluid Mech 1:625–643
19. Albery WJ, Brett CMA (1983) The wall-jet ring-disc electrode Part I. Theory. J Electroanal Chem 148:201–210
20. Yamada J, Matsuda H (1973) Limiting diffusion currents in hydrodynamic voltammetry. III. Wall jet electrodes. J Electroanal Chem 44:189–198
21. Laevers P, Hubin A, Terryn H, Vereecken J (1995) A wall-jet electrode reactor and its application to the study of electrode reaction mechanisms. Part I: design and construction. J Appl Electrochem 25:1017–1022
22. Shukla PK, Orazem ME (2004) Hydrodynamics and mass-transfer-limited current distribution for a submerged stationary hemispherical electrode under jet impingement. Electrochim Acta 49:2901–2908
23. Varadi M, Pungor E (1975) Turbulent hydrodynamic voltammetry. I. Distribution of voltammetric current on electrode surfaces. Anal Chim Acta 80:31–37

24. Varadi M, Gratzl M, Pungor E (1976) Turbulent hydrodynamic voltammetry. II. Turbulence frequency studies in a hydrodynamic voltammetric cell. Magy Kem Foly 82:335–338 (in Hungarian)
25. von Kármán T (1921) Über laminare und turbulente Reibung. Z Angew Math Mech 1:233–252
26. Cochran WG (1934) The flow due to a rotating electrode. Proc Camb Philos Soc 30:365–375
27. Sparrow EM, Gregg JL (1959) Heat transfer from a rotating disk to fluids of any Prandtl number. J Heat Transf 81C:249–252
28. Gregory DP, Riddiford AC (1956) Transport to the surface of a rotating disc. J Chem Soc 3756–3764
29. Amatore C, Pebay C, Thouin L, Wang A (2009) Cyclic voltammetry at microelectrodes. Influence of natural convection on diffusion layers as characterized by in situ mapping of concentration profiles. Electrochem Commun 11:1269–1272
30. Amatore C, Klymenko OV, Svir I (2012) Importance of correct prediction of initial concentrations in voltammetric scans: contrasting roles of thermodynamics, kinetics, and natural convection. Anal Chem 84:2792–2798
31. Dolgikh O, Demeter AS, Bastos AC, Topa V, Deconinck J (2013) A practical way to model convection in non-agitated electrolytes. Electrochem Commun 37:20–23
32. Pebay C, Sella C, Thouin L, Amatore C (2013) Mass transport at infinite regular arrays of electrodes submitted to natural convection: theory and experiments. Anal Chem 85:12062–12069
33. Shorygin AP, Danielyan GL (1978) Numerical simulation of electrochemical processes with free convection. Electrochim Acta 23:1227–1231
34. Volgin VM, Volgina OV, Bograchev DA, Davydov AD (2003) Simulation of ion transfer under conditions of natural convection by the finite difference method. J Electroanal Chem 546:15–22
35. Volgin VM, Davydov AD (2010) Numerical simulation of natural convection of electrolyte solution with three types of ions in the electrochemical cell with vertical electrodes. Russ J Electrochem 46:1360–1372
36. Vielstich W (1953) Der Zusammenhang zwischen Nernstscher Diffusionsschicht und Prandtlscher Strömungsgrenzschicht. Z Elektrochem 57:646–655
37. Bernstein C, Heindrichs A, Vielstich W (1978) Investigations of fast electrode processes by means of a micro-ring electrode in turbulent pipe flow. J Electroanal Chem 87:81–90
38. Flanagan JB, Marcoux L (1974) Digital simulation of tubular electrode response in stationary and flowing solution. J Phys Chem 78:718–723
39. Kader BA (1977) Structure of the concentration field near the leading edge of electrochemical mass transfer probes. Sov Electrochem 13:417–423
40. Wu Y, Wang Z (1999) The theoretical behaviors of tubular electrodes: from semi-infinite diffusion to bulk electrolysis. Electrochim Acta 44:2281–2286
41. Albery WJ (1966) Ring-disk electrodes. Part 1 - a new approach to the theory. Trans Faraday Soc 62:1915–1919
42. Albery WJ, Bruckenstein S (1966) Ring-disk electrodes. Part 2 - theoretical and experimental collection efficiencies. Trans Faraday Soc 62:1920–1931
43. Albery WJ, Bruckenstein S, Napp DT (1966) Ring-disk electrodes. Part 3 - current-voltage curves at the ring electrode with simultaneous currents at the disc electrode. Trans Faraday Soc 62:1932–1937
44. Albery WJ, Bruckenstein S, Johnson DC (1966) Ring-disk electrodes. Part 4 - diffusion layer titration curves. Trans Faraday Soc 62:1938–1945
45. Albery WJ, Bruckenstein S (1966) Ring-disk electrodes. Part 5 - first-order kinetic collection efficiencies at the ring electrode. Trans Faraday Soc 62:1946–1954
46. Albery WJ, Jones CC, Mount AR (1989) New hydrodynamic methods. In: Compton RG, Hancock G (eds) Comprehensive chemical kinetics, vol 29. Elsevier, Amsterdam, pp 129–148

47. Penar J (1991/1992) Application of the rotating disc and ring-disc electrodes for investigations of the kinetics and of mechanism of electrode reactions. Theoretical basis. Ann Univ Marie Curie-Skłodowska Lublin-Polonia 46/47:119–172 (in Polish)
48. Williams DE, MacPherson J (1999) Hydrodynamic modulation methods in electrochemistry. In: Compton RG, Hancock G (eds) Comprehensive chemical kinetics, vol 37. Elsevier, Amsterdam, pp 369–438
49. Ang KP, Gunasingham H, Tay BT (1987) Cyclic voltammetry of some quinones and nitroaromatic compounds using a mercury film wall-jet detector and flow injection analysis. J Singapore Natl Acad Sci 16:80–86
50. Frumkin A, Nekrasov L, Levich B, Ivanov J (1959/60) Die Anwendung der rotierenden Scheibenelektrode mit einem Ringe zur Untersuchung von Zwischenprodukten electrochemischer Reaktionen. J Electroanal Chem 1:84–90
51. Banks CE, Compton RG, Fisher AC, Henley IE (2004) The transport limited currents at insonated electrodes. Phys Chem Chem Phys 6:3147–3152
52. Baleras F, Bouet V, Deslouis C, Maurin G, Sobolik V, Tribollet B (1996) Flow measurement in an impinging jet cell with three-segment microelectrodes. Exp Fluids 22:87–93
53. Bartolini R, Fantini L, Gallone P (1976) Fluid velocity measurements by the electrochemical method. Ann Chim (Rome) 66:7–18
54. Mizushina T (1971) The electrochemical method in transport phenomena. Adv Heat Mass Transf 7:87–161
55. Kermiche-Aouanouk F, Daguenet M (1972) Theorie des dimicroéléctrodes. J Chim Phys Physicochim Biol 69:1705–1710 (in French)
56. Bruckenstein S, Johnson DC (1964) Coulometric diffusion layer titrations using the ring-disk electrode with amperometric end point detection. Anal Chem 36:2186–2187
57. Rajantie H, Strutwolf J, Williams DE (2001) Theory and practice of electrochemical titrations with dual microband electrodes. J Electroanal Chem 500:108–120
58. Basak J, Penar J, Sykut K (1987/1988) Digital simulation for determining rate constants in diffusion layer titration on the rotating ring disc electrode. Part II. Second order reactions. Ann Univ Mariae Curie - Skłodowska, Sectio AA XLII/XLIII:43–49
59. van Leeuwen HP, de Jong HG, Holub K (1989) Voltammetry of metal complex systems with different diffusion coefficients of the species involved. Part IV. Simulation of the limiting current for any metal-to-ligand ratio and elaboration to voltammetric titration curves. J Electroanal Chem 260:213–220
60. Svir IB, Oleinick AI, Compton RG (2003) Dual microband electrodes: current distributions and diffusion layer 'titrations'. Implications for electroanalytical measurements. J Electroanal Chem 560:117–126
61. Arkoub IA, Amatore C, Sella C, Thouin L, Warkocz JS (2001) Diffusion at double microband electrodes operated within a thin film coating. Theory and experimental illustration. J Phys Chem B 105:8694–8703
62. Harrington MS, Anderson LB (1990) Analytical strategies using interdigitated filar microelectrodes. Anal Chem 62:546–550
63. Postlethwaite TA, Hutchinson JE, Murray R, Fosset B, Amatore C (1996) Interdigitated array electrodes as an alternative to the rotating ring-disk electrode for determination of the reaction products of dioxygen reduction. Anal Chem 68:2951–2958
64. Seddon BJ, Girault HH, Eddowes MJ (1989) Interdigitated microband electrodes: chronoamperometry and steady state currents. J Electroanal Chem 266:227–238
65. Svir IB, Klimenko AV, Compton RG (2001) The simulation of convective diffusion transport of matter to a channel double microband electrode and its application to electrogenerated chemiluminescence. Radiotekhnika 118:92–101
66. Amatore C (1995) Electrochemistry at ultramicroelectrodes. In: Rubinstein I (ed) Physical electrochemistry. Marcel Dekker, New York, pp 131–208
67. Britz D, Kastening B (1974) On the electrochemical observation of a second-order decay of radicals generated by flash photolysis or pulse radiolysis. J Electroanal Chem 56:73–90

68. Feldberg SW (1969) Digital simulation: a general method for solving electrochemical diffusion-kinetic problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 3. Marcel Dekker, New York, pp 199–296
69. Feldberg SW (1980) Improvements on computer simulation of electrochemical phenomena involving hydrodynamics: the rotating disk and dropping mercury electrode. J Electroanal Chem 109:69–82
70. Ružić I, Smith DE (1974) On the influence of electrode curvature and growth in d.c. and a.c. polarography: the e.e. mechanism with amalgam formation. J Electroanal Chem 57:129–139
71. Prater KB, Bard AJ (1970) Rotating ring-disk electrodes. 1. Fundamentals of the digital simulation approach. Disk and ring transients and collection efficiencies. J Electrochem Soc 117:207–213
72. Prater KB, Bard AJ (1970) Rotating ring-disk electrodes. II Digital simulation of first and second-order following chemical reactions. J Electrochem Soc 117:335–340
73. Prater KB (1972) Digital simulation and modelling. Chem Instrum 3:259–269
74. Margarit J, Lévy M (1974) Étude theéoretique d'une électrode tournante á double anneau. Partie I. Recherche du facteur d'efficacité par une méthode de simulation numérique. J Electroanal Chem 49:369–376
75. Margarit J, Dabosi G, Lévy M (1975) Étude d'une électrode tournante á double anneau. Partie II. Vérification expérimentale des resultats obtenus par voie de simulation numérique. Bull Soc Chim Fr 7–8:1509–1512
76. Clarenbach S, Grabner EW, Brauer E (1973) Digital simulation of a rotating double-ring-electrode. Ber Bunsenges Phys Chem 77:908–913
77. Clarenbach S, Grabner EW (1976) Application of digital simulation to a mass transport problem: calculation of the velocity of flow at a rotating disk. Ber Bunsenges Phys Chem 80:115–121
78. Mandin P, Pauporte T, Fanouillère P, Lincot D (2004) Modelling and numerical simulation of hydrodynamical processes in a confined rotating electrode configuration. J Electroanal Chem 565:159–173
79. Feldberg SW, Bowers ML, Anson FC (1986) Hopscotch-finite-difference simulation of the rotating ring-disc electrode. J Electroanal Chem 215:11–28
80. Nolan JE, Plambeck JA (1990) The EC-catalytic mechanism at the rotating disk electrode. Part II. Comparison of approximate theories for the second-order case and application to the reaction of bipyridinium cation radicals with dioxygen in non-aqueous solutions. J Electroanal Chem 294:1–20
81. Balslev H, Britz D (1992) Direct digital simulation of the steady-state limiting current at a rotating disk electrode for a complex mechanism. Acta Chem Scand 46:949–955
82. Dan C, Van den Bossche B, Bortels L, Nelissen G, Deconinck J (2001) Numerical simulation of transient current responses in diluted electrochemical ionic systems. J Electroanal Chem 505:12–23
83. Strutwolf J (1995) Digitale Simulation elektrochemischer Systeme: Untersuchungen zeitabhängiger Phänomene an rotierenden Scheibenelektroden und Analyse von Cyclovoltammogrammen durch direkte Simulation. Ph.D. thesis, Universität Bielefeld, Bielefeld
84. Strutwolf J, Schoeller WW (1996) Linear and cyclic sweep voltammetry at a rotating disk electrode. A digital simulation. Electroanalysis 8:1034–1039
85. Gooch KA, Fisher AC (2002) Computational electrochemistry: the simulation of voltammetry under hydrodynamic modulation control. J Phys Chem B 106:10668–10673
86. Gooch KA, Qiu FL, Fisher AC (2003) The digital simulation of voltammetry under stagnant and hydrodynamic conditions. In: Bard AJ, Stratmann M, Unwin PR (eds) Encyclopaedia of electrochemistry, volume 2, Instrumentation and electroanalytical chemistry. Wiley-VCH, Weinheim, pp 122–142
87. Albery WJ, Chadwick AT, Coles BA, Hampson NA (1977) The tube electrode and E.S.R.: second order kinetics. J Electroanal Chem 75:229–239
88. Albery WJ, Compton RG, Chadwick AT, Coles BA, Lenkaits JA (1980) Tube electrode and electron spin resonance. First-order kinetics. J Chem Soc Faraday Trans I 76:1391–1401

89. Andersen JL, Moldoveanu S (1984) Numeral simulation of convective diffusion at a rectangular channel flow electrode. J Electroanal Chem 179:107–117
90. Compton RG, Pilkington MBG, Stearn GM (1988) Mass transport in channel electrodes. The application of the backwards implicit method to electrode reactions (EC, ECE and DISP) involving coupled homogeneous kinetics. J Chem Soc Faraday Trans I 84:2155–2171
91. Compton RG, Fisher AC, Latham MH, Brett CMA, Brett AMCFO (1992) Transient measurements at the wall-jet ring disc electrode. J Appl Electrochem 22:1011–1016
92. Compton RG, Coles BA, Fisher AC (1994) Chronoamperometry at channel electrodes. Theory of double electrodes. J Phys Chem 98:2441–2445
93. Compton RG, Coles BA, Gooding JJ, Fisher AC (1994) Chronoamperometry at channel electrodes. Experimental applications of double electrodes. J Phys Chem 98:2446–2451
94. Fletcher CAJ (1991) Computational techniques for fluid dynamics, vol I, 2nd edn. Springer, Berlin
95. Strikwerda JC (1989) Finite difference schemes and partial differential equations. Wadsworth and Brooks/Cole, Pacific Grove, CA
96. Alhumaizi K (2004) Comparison of finite difference methods for the numerical simulation of reacting flow. Comput Chem Eng 28:1759–1769
97. Alden JA, Compton RG (1996) Hydrodynamic voltammetry with channel microband electrodes: axial diffusion effects. J Electroanal Chem 404:27–35
98. Alden JA, Cooper JA, Hutchinson F, Prieto F, Compton RG (1997) Channel electrode voltammetry and reversible electro-dimerisation processes. The reduction of the methylviologen di-cation in aqueous solution. J Electroanal Chem 432:63–70
99. Alden JA, Feldman MA, Hill E, Prieto F, Oyama M, Coles BA, Compton RG (1998) Channel microband electrode arrays for mechanistic electrochemistry. Two-dimensional voltammetry: transport-limited currents. Anal Chem 70:1707–1720
100. Amatore C, Oleinick A, Svir I (2004) Simulation of diffusion-convection processes in microfluidic channels equipped with double-band microelectrode assemblies: approach through quasi-conformal mapping. Electrochem Commun 6:1123–1130
101. Amatore C, Da Mota N, Lemmer C, Pebay C, Sella C, Thouin L (2008) Theory and experiments of transport at channel microband electrodes under laminar flows. 2. Electrochemical regimes at double microband assemblies under steady state. Anal Chem 80:9483–9490
102. Amatore C, Lemmer C, Sella C, Thouin L (2011) Channel microband chronoamperometry: from transient to steady-state regimes. Anal Chem 83:4170–4177
103. Amatore C, Lemmer C, Perrodin P, Sella C, Thouin L (2011) Theory and experiments of microelectrodes performing as concentration probes within microfluidic channels with high temporal resolution. Electrochem Commun 13:1459–1461
104. Bidwell MJ, Alden JA, Compton RG (1996) Channel microband electrodes: a complete working surface for potential step transients. J Electroanal Chem 414:247–251
105. Bidwell MJ, Alden JA, Compton RG (1996) Hydrodynamic voltammetry with channel microband electrodes: the simulation of voltammetric waveshapes. J Electroanal Chem 417:119–128
106. Bidwell MJ, Alden JA, Compton RG (1997) Electroanalysis in flowing systems - the propagation of depletion effects downstream of a channel micro-band electrode. Electroanalysis 9:383–389
107. Bieniasz LK (2013) Automatic solution of the Singh and Dutt integral equations for channel or tubular electrodes, by the adaptive Huber method. J Electroanal Chem 693:95–104
108. Bortels L, Deconinck J, Bossche BVD (1996) The multi-dimensional upwinding method as a new simulation tool for the analysis of multi-ion electrolytes controlled by diffusion, convection and migration. Part 1. Steady state analysis of a parallel plane flow channel. J Electroanal Chem 404:15–26
109. Cooper JA, Alden JA, Oyama M, Compton RG, Okazaki S (1998) Channel electrode voltammetry: the kinetics of the complexation of the chloranil radical anion with $M^{2+}$ ions by waveshape analysis. J Electroanal Chem 442:201–206
110. Cooper JA, Compton RG (1998) Channel electrodes - a review. Electroanalysis 10:141–155

111. Ferrigno R, Brevet PF, Girault HH (1997) Finite element simulation of the amperometric response of recessed and protruding microband electrodes in flow channels. J Electroanal Chem 430:235–242
112. Fisher AC, Compton RG (1991) Chronoamperometry at channel electrodes: a general computational approach. J Phys Chem 95:7538–7542
113. Fisher AC, Compton RG (1992) A general computational approach to linear sweep voltammetry at channel electrodes. J Appl Electrochem 22:38–42
114. Fisher AC, Compton RG (1992) The EC' mechanism: split waves at the channel electrode. Electroanalysis 4:311–315
115. Fuhrmann J, Zhao H, Holzbecher E, Langmach H, Chojak M, Halseid R, Jusys Z, Behm J (2008) Experimental and numerical model study of the limiting current in a channel flow cell with a circular electrode. Phys Chem Chem Phys 10:3784–3795
116. Fulian Q, Stevens NPC, Fisher AC (1998) Computer-aided design and experimental application of a novel electrochemical cell: the confluence reactor. J Phys Chem B 102:3779–3783
117. Fulian Q, Fisher AC, Riley DJ (2000) The computer aided design and experimental development of a new device for the measurement of electrochemiluminescence. Electroanalysis 12:503–508
118. Gooch KA, Williams NA, Fisher AC (2000) The computer-aided design of a new hydrodynamic device for studying mechanisms of chemical transfer across the liquid|liquid interface. Electrochem Commun 2:51–55
119. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. An E reaction at a channel microband electrode. Electrochem Commun 2:567–575
120. Henstridge MC, Rees NV, Compton RG (2012) A comparison of the Butler-Volmer and asymmetric Marcus-Hush models of electrode kinetics at the channel electrode. J Electroanal Chem 687:79–83
121. Holm T, Sunde S, Seland F, Harrington DA (2015) A semianalytical method for simulating mass transport at channel electrodes. J Electroanal Chem 745:72–79
122. Klymenko OV, Oleinick AI, Amatore C, Svir I (2007) Reconstruction of hydrodynamic flow profiles in a rectangular channel using electrochemical methods of analysis. Electrochim Acta 53:1100–1106
123. Leslie WM, Alden JA, Compton RG, Silk T (1996) ECE and DISP processes at channel electrodes: analytical theory. J Phys Chem 100:14130–14136
124. Ma S, Wu Y, Wang Z (1999) Spectroelectrochemistry for a coupled chemical reaction in the channel cell. Part I. Theoretical simulation of an EC reaction. J Electroanal Chem 464:176–180
125. Miles AB, Compton RG (2001) Simulation of square-wave voltammetry at a channel electrode: E, EC and ECE processes. J Electroanal Chem 499:1–16
126. Moldoveanu S, Anderson JL (1985) Numerical simulation of convective diffusion at a microarray channel electrode. J Electroanal Chem 185:239–252
127. Ou TY, Moldoveanu S, Anderson JL (1988) Hydrodynamic voltammetry at an interdigitated electrode array in a flow channel. Part II. Chemical reaction succeeding electron transfer. J Electroanal Chem 247:1–16
128. Pastore P, Magno F, Lavagnini I, Amatore C (1991) Digital simulation via the hopscotch algorithm of a microelectrode-based channel flow-through amperometric detector. J Electroanal Chem 301:1–13
129. Prieto F, Aixill WJ, Alden JA, Coles BA, Compton RG (1997) Voltammetry under high mass transport conditions. The high-speed channel electrode and transient measurements. J Phys Chem B 101:5540–5544
130. Prieto F, Oyama M, Coles BA, Alden JA, Compton RG, Okazaki S (1998) Mechanistic determination using arrays of variable sized channel microband electrodes. The oxidation of 2,3,7,8-tetra-methoxythianthrene in the presence of pyridine in acetonitrile solution. Electroanalysis 10:685–690

131. Qiu F, Compton RG, Coles BA, Marken F (2000) Thermal activation of electrochemical processes in a Rf-heated channel flow cell: experiment and finite element simulation. J Electroanal Chem 492:150–155
132. Rajendran L (2000) Padé approximation of ECE and DISP processes at channel electrodes. Electrochem Commun 2:186–189
133. Rajendran L (2000) Padé approximation of EC' processes at channel electrodes. J Electroanal Chem 487:72–74
134. Rajendran L (2006) Two-point Padé approximation of mass transfer rate at microdisk electrodes in a channel flow for all Péclet numbers. Electrochim Acta 51:5407–5411
135. Rees NV, Klymenko OV, Maisonhaute E, Coles BA, Compton RG (2003) The application of fast scan cyclic voltammetry to the high speed channel electrode. J Electroanal Chem 542:23–32
136. Somov SI, Brainin MI, Baraboshkin DA (1996) Mass transfer in gas channels of electrochemical cells based on solid-oxide electrolyte at small concentrations of electrochemically active components in the gas: II. Results of numerical calculations. Russ J Electrochem 32:1103–1107
137. Stevens NPC, Fisher AC (1998) Transient voltammetry under hydrodynamic conditions. Electroanalysis 10:16–20
138. Svir IB, Klimenko AV, Compton RG (2000) Two approaches for digital simulation of the channel flow cell problem. Radioelek Informatika 2:29–33
139. Tait RJ, Bury PC, Finnin BC, Reed B, Bond AM (1993) An explicit finite difference simulation for chronoamperometry at a disk microelectrode in a channel flow solution. J Electroanal Chem 356:25–42
140. Ueno K, Kim HB, Kitamura N (2003) Characteristic electrochemical responses of polymer microchannel-microelectrode chips. Anal Chem 75:2086–2091
141. Unwin PR, Compton RG (1989) The use of channel electrodes in the investigation of interfacial reaction mechanisms. In: Compton RG, Hancock G (eds) Comprehensive chemical kinetics, vol 29. Elsevier, Amsterdam, pp 173–296
142. Engblom SO, Cope DK, Tallman DE (1996) Diffusion current at the tubular band electrode by the integral equation method. J Electroanal Chem 406:23–31
143. Lovrić M, Kormorsky-Lovrić Š, Kahlert H, Scholz F (2007) A model of mass transport near the tube wall in a flow-injection manifold. Anal Chim Acta 602:75–81
144. Singh T, Singh RP (2000) Linear sweep voltammetry of irreversible charge transfer coupled with irreversible catalytic reaction under diffusion-convection control: an integral equation approach. Indian J Pure Appl Math 31:363–374
145. Alden JA, Hakoura S, Compton RG (1999) Finite difference simulations of steady-state voltammetry at the wall-jet electrode. Effects of radial diffusion and working curves for common electrochemical mechanisms. Anal Chem 71:827–836
146. Ball JC, Compton RG, Brett CMA (1998) Theory of anodic stripping voltammetry at wall-jet electrodes. Simulation of spatially differential stripping and redeposition phenonema. J Phys Chem B 102:162–166
147. Bitziou E, Rudd NC, Edwards MA, Unwin PR (2006) Visualization and modeling of the hydrodynamics of an impinging microjet. Anal Chem 78:1435–1443
148. Bitziou E, Rudd NC, Unwin PR (2007) Microjet ring electrode (MJRE): development, modelling and experimental characterisation. J Electroanal Chem 602:263–274
149. Coles BA, Compton RG, Brett CMA, Brett AMCFO (1995) Ohmic distortion of current-potential curves at wall-jet electrodes. J Electroanal Chem 381:99–104
150. Compton RG, Greaves CR, Waller AM (1990) A general computation method for mass-transport problems involving wall-jet electrodes and its application to simple electron-transfer, ECE and DISP1 reactions. J Appl Electrochem 20:575–585
151. Compton RG, Greaves CR, Waller AM (1990) The wall-jet electrode and its application to the study of electrode reactions with the coupled homogeneous kinetics: the DISP1 reaction. J Appl Electrochem 20:586–589

152. Compton RG, Fisher A, Tyley GP (1990) The wall-jet electrode and the study of electrode reaction mechanisms: the EC reaction. J Appl Electrochem 20:912–915
153. Compton RG, Fisher AC, Tyley GP (1991) The wall-jet electrode and the study of electrode reaction mechanisms: the EC' (catalytic) reaction. J Appl Electrochem 21:2–5
154. Compton RG, Fisher AC, Latham MH, Wellington RG, Brett CMA, Brett AMCFO (1993) Wall-jet electrodes: the importance of radial diffusion. J Appl Electrochem 23:98–102
155. Fisher AC, Compton RG, Brett CMA, Brett AMCF (1991) The wall-jet electrode. Potential step chronoamperometry. J Electroanal Chem 318:53–59
156. Klymenko OV, Gavaghan DJ, Harriman KE, Compton RG (2002) Finite element simulation of electrochemically reversible, quasireversible and irreversible linear sweep voltammetry at the wall tube electrode. J Electroanal Chem 531:25–31
157. Laevers P, Hubin A, Terryn H, Vereecken J (1995) A wall-jet electrode reactor and its application to the study of electrode reaction mechanisms. Part II: a general computational method for the mass transport problems involved. J Appl Electrochem 25:1023–1030
158. Melville J, Simjee N, Unwin PR, Coles B, Compton RG (2002) Hydrodynamics and mass transport in wall tube and microjet electrodes. 1. Finite element simulations. J Phys Chem B 106:2690–2698
159. Compton RG, Dryfe RAW, Wellington RG, Hirst J (1995) Modelling electrode reactions using the strongly implicit procedure. J Electroanal Chem 383:13–19
160. Matsuda H (1967) Zur Theorie der stationären Strom-Spannungs-Kurven von Redox-Elektrodenreaktionen in hydrodynamischer Voltammetrie. II. Laminare Rohr- und Kanalströmungen. J Electroanal Chem 15:325–336
161. Alden JA, Compton RG (1996) The multigrid method, MGD1: an efficient and stable approach to electrochemical modelling. The simulation of double electrode problems. J Electroanal Chem 415:1–12

# Chapter 15
# Performance

In this chapter, the performance of the various methods that have been described is examined. This involves convergence, stability and economy of computer time. Some of the more sensible simulation methods are compared. Sensitivity analysis is briefly mentioned.

## 15.1 Convergence

The aim of a simulation is to approximate the underlying exact solution as accurately as desired in a minimum of computer time. Solution is achieved by some discrete formula, which has truncation errors, due to neglect of some (higher) Taylor terms in the discretisation formulae, as well as machine roundoff errors. Truncation errors must become smaller as we make the intervals both in time and space smaller and the errors must, at least, not grow in the course of a number of steps. This property is called convergence. In the limit, as $\delta T$ and $\delta X$ (that is, $H$) approach zero, the errors must also do so. In order for this to happen, two conditions must hold. The first is that the discretisation expression used must be consistent with the differential equation it approximates. The second is that the expression must be stable. This means that an error in the solution at a given step is not amplified by subsequent steps. These two issues will be examined separately.

Generally, it can be said that consistency is not as great a problem as stability, as pointed out by Lapidus and Pinder [1]. Inconsistent discretisations have been devised, but they are rather rare.

Another way of viewing convergence is not whether a given solution converges towards the exact solution, but how it does so. Does it approach smoothly, the errors keeping on one side of the zero line, or does it approach with oscillations? Not all regard oscillations as a bad thing [2]. In electrical engineering, for example, an optimal control circuit is often the one that responds to a step change in an input

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8_15

with a strongly damped oscillation, a smooth response being slower. However, if the oscillations persist for a long time, they are again not optimal. This is mirrored in simulation in the ways some stable methods behave. BI has what some regard as a pleasant, smooth, approach to the solution, while CN, when using large $\lambda$ values, oscillates. CN is, however, the method with higher-order of accuracy, and if the oscillations are damped quickly, it is the better method. It is possible to prevent the CN oscillations by damping them in the first few steps [3–6], see Chap. 8.

To illustrate the above, consider Fig. 15.1. For the figure, equal intervals were used, a time interval of 0.01 and as seen, three different values of $\lambda$ (which determine $N$, the number of spatial intervals). The current was computed as a five-point approximation using our `G0FUNC`, and the error is relative to the known Cottrell solution. As $\lambda$ increases to 0.5 (its maximum value for stability with EX) EX is seen to oscillate more and more but in a damped manner, while the error for BI is damped smoothly, although it is comparable with that of method EX (perhaps

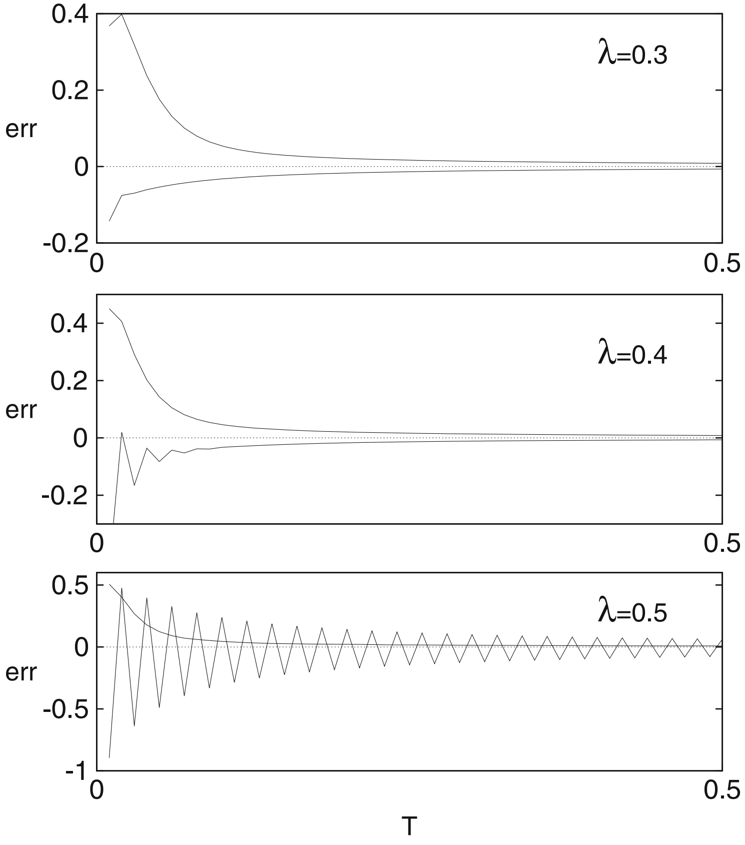


**Fig. 15.1** Relative errors in the simulated Cottrell current, using methods EX and BI, for the indicated $\lambda$ values. The *top curves* are for BI

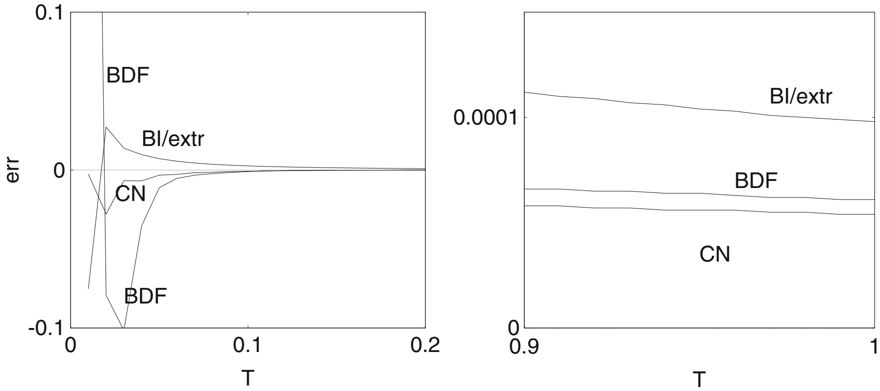


**Fig. 15.2** Relative errors in the simulated Cottrell current, using the indicated methods, with $\lambda = 3$

slightly larger). At $\lambda = 0.5$ EX has not stopped oscillating at the time limit of unity, which no doubt is the reason that those using EX often choose $\lambda$ at 0.45 [7, p. 794], or somewhere between 0.4 and 0.49 [8–10]. Figure 15.2, on the other hand, now shows the relative error, still using equal spatial intervals and a time interval of 0.01 and the five-point current approximation for three implicit methods, all for $\lambda = 3$. At this $\lambda$ value, EX is of course unstable but the implicit methods are not. CN was run with 10 Pearson pre-steps to damp out its oscillations, and BI was coupled with second order extrapolation; three-point BDF was used with the rational start, that is, starting with a single BI step. All three methods quickly converge to a very small relative error, so that only a short time span is shown, and appear to converge roughly equally well after some largish swings during the first few steps. This is seen in the left-hand figure. The right-hand figure shows the values close to $T = 1$ to compare the methods' convergence. It was necessary to increase the outer spatial limit from the usual 6 to 8, because the relative errors are now so small that the usual limit affects them. We note that CN is the most accurate, closely followed by BDF, followed by BI/extrap.

In what follows, the issues of consistency, stability and efficiency are addressed.

## 15.2 Consistency

In the present context we have two intervals: $\delta T$ in time, and $H$ in space. A given discretisation is said to be consistent if, as both of these intervals approach zero, the discretisation approaches the *pde* it is meant to approximate. Take the simple explicit discretisation on equal intervals, in Eq. (5.2), which we rearrange into the form

$$C'_i = \lambda C_{i-1} + (1 - 2\lambda)C_i + \lambda C_{i+1} \ . \qquad (15.1)$$

The three terms at a small interval away from $C_i$ can be Taylor-expanded around the value $C_i$. We have (dropping the index $i$ for the derivatives)

$$C_i' = C_i + \delta T \frac{\partial C}{\partial T} + \frac{\delta T^2}{2} \frac{\partial C^2}{\partial T^2} + O(\delta T^3)$$

$$C_{i-1} = C_i - H \frac{\partial C}{\partial X} + \frac{H^2}{2} \frac{\partial C^2}{\partial X^2} - \frac{H^3}{6} \frac{\partial C^3}{\partial X^3} + \frac{H^4}{24} \frac{\partial C^4}{\partial X^4} - O(H^5)$$

$$C_{i+1} = C_i + H \frac{\partial C}{\partial X} + \frac{H^2}{2} \frac{\partial C^2}{\partial X^2} + \frac{H^3}{6} \frac{\partial C^3}{\partial X^3} + \frac{H^4}{24} \frac{\partial C^4}{\partial X^4} + O(H^5)$$

where the $O(..)$ terms indicate where the Taylor series were cut off. Inserting these in (15.1) and tidying up, this becomes

$$\frac{\partial C}{\partial T} + \frac{\delta T}{2} \frac{\partial^2 C}{\partial T^2} + O(\delta T^2) = \frac{\partial^2 C}{\partial X^2} + \frac{H^2}{12} \frac{\partial^4 C}{\partial X^4} + O(H^4) \tag{15.2}$$

and it is clear that as both $\delta T$ and $H$ approach zero, the equation approaches the *pde* we are in fact trying to approximate.

Most discrete approximations that have been mentioned in this book are consistent, except one. This is the DuFort–Frankel method [11], described on page 185 in Chap. 9. It is stable for all $\lambda$, yet has a consistency problem. Giving Eq. (9.18) the same treatment as above, one ends with

$$\frac{\partial C}{\partial T} + \frac{\delta T^2}{6} \frac{\partial^3 C}{\partial T^3} + O(\delta T^3) = \frac{\partial^2 C}{\partial X^2} + \frac{H^2}{12} \frac{\partial^4 C}{\partial X^4} - \frac{\delta T^2}{H^2} \frac{\partial^2 C}{\partial T^2} + O(H^4)\ . \tag{15.3}$$

It is seen that again, two of the three remaining Taylor terms vanish as the intervals approach zero, but the one containing $\frac{\delta T^2}{H^2}$ does not. As pointed out by Shih [12] and Strutwolf [13], what is happening here is that the approximation is consistent not with the parabolic *pde*, but with a hyperbolic one instead. For large $\lambda$ (leading also to large $\frac{\delta T^2}{H^2}$), the DuFort–Frankel method is not suitable for the simulation of parabolic problems. The method has been suggested [14] for use in electrochemical simulations but since Rudolph [15] pointed out problems with it, it has not been used again in electrochemistry (as far as we know).

## 15.3 Stability

The stability of a given simulation method can be defined rigorously mathematically, or more loosely. A loose description might be that given in Smith [16, p. 47], namely that the amplification of initial conditions be limited. This means that if, due to truncation or roundoff, there be errors in, say, the concentration values at a given step, these errors are not amplified without bound in subsequent steps. There are various categories of stability, to be seen in the relevant texts [1, 16, 17] (to cite only three of a multitude of such texts). Below, a rather brief and less technical treatment will be given than is provided in these texts. Several methods for determining stability are described, and some special conditions that can affect stability.

From the range of methods for determining stability of a given algorithm such as EX, CN, BI or BDF, etc., this chapter restricts itself to the heuristic, the Neumann and the matrix methods, as well as a fourth that makes use of the stability function.

Stability can be divided into a number classes. We refer to *conditional* and *unconditional* stability. EX is a conditionally stable method, because there is a restriction on the value of $\lambda$, whereas CN and Laasonen are unconditionally stable. There is, however, a difference between their stabilities. Dahlquist [18, 19] and Henrici [20] refer to *weak* and *strong* stability. This has since been tightened to a number of sub-cases of stability, with a rough division between those methods that show **A-stability** and those that show **L-stability**. For more details, see such texts as Smith [16] and Hairer and Wanner [21]. Here it will suffice to describe them in less detail. Methods that are A-stable (such as CN) have some error propagators close to (but less than) unity in magnitude for large $\lambda$ (for error propagation, see the Neumann method, below). L-stable methods, on the other hand, have error propagators that all approach zero as $\lambda$ grows larger. Laasonen (BI) is one such method. One prefers methods to be L-stable.

### 15.3.1 Heuristic Method

Lapidus and Pinder [1] describe the simplest of all stability determinations and call it the heuristic method. With this method, one tries to compute a few steps in time from a perturbation in initial conditions, and sees how the perturbation is propagated after a number of steps. In Fig. 15.3, the explicit method is used, Eq. (15.1), setting $\lambda = 0.5$. At some time level $N\delta T$, an error is placed at a point along $X$ in Fig. 15.3, following Dahlquist [22, p. 386] (the value unity planted among a row of zeroes). Points marked in the figure without numbers are assumed to have zero values. It is seen that in subsequent steps after the $N$th, the error is spread along $X$, but with decreasing magnitude. It is known that for $\lambda \leq 0.5$, this method is stable, and the figure suggests this. Now consider Fig. 15.4, where the same calculation has been done using $\lambda = 1$. Obviously, the error is quickly amplified, and the method is unstable.

Planting a perturbation in a simulation can be a useful way of testing the stability, especially under special conditions.

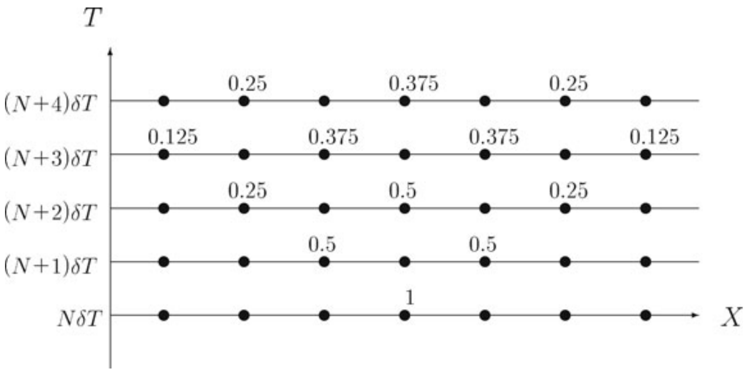


Fig. 15.3 Propagation of a single error for EX with $\lambda = 0.5$

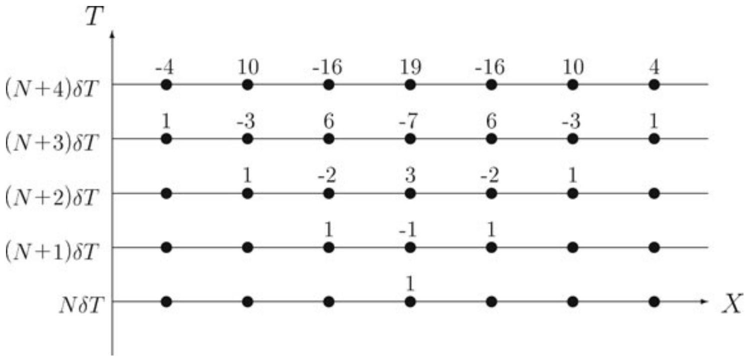


Fig. 15.4 Propagation of a single error for EX with $\lambda = 1$

## 15.3.2 Von Neumann Stability Analysis

A more analytical method of stability analysis is the method of von Neumann [23, 24] (note that [24] is mostly incorrectly cited as being of the year 1951 [25]). The method focusses on an interior point along $X$ in the grid and looks at the propagation of an error at that point, making certain reasonable assumptions, using Fourier series (which is why the method on occasion is called the Fourier series method).

We need first to develop an argument that allows us to separate concentrations from their errors. Let the vector $\mathbf{C}$ of concentrations along a space coordinate $X$ be the sum of a vector $\hat{\mathbf{C}}$ of exact values, with an error vector $\boldsymbol{\epsilon}$ added:

$$\mathbf{C} = \hat{\mathbf{C}} + \boldsymbol{\epsilon} \ . \tag{15.4}$$

We are interested in what happens to the errors, and the linear nature of (15.4) allows us to subtract the concentrations out of the diffusion equations, leaving only the errors. These have simpler boundary conditions; for example, errors far away from the electrode are zero. This simplifies the form of the equations describing the changes in errors. The diffusion equation for the errors is then

$$\frac{\partial \epsilon}{\partial T} = \frac{\partial^2 \epsilon}{\partial X^2} \tag{15.5}$$

with either a Dirichlet or derivative (Neumann) boundary condition at the electrode, and $\epsilon(X \to \infty) = 0$. The equation is discretised, using equal intervals, at the point with index $i$, as usual. For example, using method EX, we have

$$\epsilon_i' = \epsilon_i + \lambda \left( \epsilon_{i-1} - 2\epsilon_i + \epsilon_{i+1} \right) \ . \tag{15.6}$$

Two substitutions are now made. Firstly, it is assumed that at a given point, the value of $\epsilon$ there changes with time in a general exponential manner,

$$\epsilon = \epsilon_0 \, \mathrm{e}^{\alpha T} \tag{15.7}$$

in which $\alpha$ is some complex constant and $\epsilon_0$ is an initial value. Setting, for convenience, $\exp(\alpha \delta T) = \xi$, so that we can substitute for $\epsilon'$, as in

$$\epsilon(T + \delta T) = \xi \, \epsilon(T) \ . \tag{15.8}$$

$\xi$ is an amplification factor, and the object of the exercise is now to find out under what conditions it is less than unity in magnitude, which is the condition for stability.

The other substitution is for the whole vector of errors, which is expressed as a Fourier series along the space coordinate ($N$ points along $X$):

$$\epsilon(X) = \sum_{i=0}^{N} a_i \mathrm{e}^{j\beta_i X_i} \ , \tag{15.9}$$

where $X_i = iH, i = 1 \ldots N$ and $\beta_i$ are "frequencies" as inverse intervals along $X$, $j$ here being the imaginary number $\sqrt{-1}$. The coefficients $a_i$ are unknown. The error vector is thus a sum of a number of Fourier components, and we can limit our view to any one of them. Writing simply $\beta$ for any one of the range of values, and substituting both (15.8) and (15.9) in (15.6), we have for the point at index $i$, going from time step $k$ to $k + 1$,

$$\xi^{k+1} \mathrm{e}^{j\beta i H} = \xi^k \mathrm{e}^{j\beta i H} + \lambda \xi^k \left( \mathrm{e}^{j\beta(i-1)H} - 2\mathrm{e}^{j\beta i H} + \mathrm{e}^{j\beta(i+1)H} \right) \tag{15.10}$$

(taking the common $\xi^k$ term on the right-hand side outside the bracket). Division by $\xi^k$ and $\mathrm{e}^{j\beta i H}$ produces

$$\xi = 1 + \lambda \left( \mathrm{e}^{-\beta H} - 2 + \mathrm{e}^{\beta H} \right) \tag{15.11}$$

leading to

$$\xi = 1 - 4\lambda \sin^2\left(\frac{i\beta}{2}\right) . \tag{15.12}$$

Since the maximum value for the $\sin^2$ term is unity, this equation leads to the condition for $|\xi| \leq 1$, that $\lambda \leq \frac{1}{2}$. This is the well-known $\lambda$ limit for the method EX.

Repeating this for the method BI, where the discrete equation at point $i$ is

$$\epsilon'_i = \epsilon_i = \lambda \left(\epsilon'_{i-1} - 2\epsilon'_i + \epsilon'_{i+1}\right) . \tag{15.13}$$

leads to

$$\xi = \frac{1}{1 + 4\lambda \sin^2(\frac{i\beta}{2})} \tag{15.14}$$

which satisfies the stability condition for all values of $\lambda$. What is more, the greater $\lambda$ is, the closer $\xi$ approaches zero.

A similar analysis for CN results in

$$\xi = \frac{1 - 2\lambda \sin^2\left(\frac{i\beta}{2}\right)}{1 + 2\lambda \sin^2\left(\frac{i\beta}{2}\right)} \tag{15.15}$$

which also sets no limits on $\lambda$, fulfilling the condition. In this case, however, as $\lambda$ increases, $\xi \to -1$, which explains the oscillatory behaviour of the method CN.

This sort of analysis can be applied to other methods. Britz and Strutwolf [26] applied it to the BDF method using five-point discretisation along $X$, and, also for five-point approximations, Strutwolf and Britz [27] applied it to extrapolation. For a multilevel method such as BDF, the analysis results in a polynomial in $\xi$, and complex roots are possible. For example, Lapidus and Pinder [1] treat the Dufort/Frankel method; it results in a quadratic equation in $\xi$ but it is clear that it is unconditionally stable (even though we have seen that it is not consistent).

### 15.3.3 Matrix Stability Analysis

The von Neumann method described above usually works well, and is reasonably easy to apply. One reason it works well, despite the fact that it totally ignores conditions at the boundaries, is that errors often arise at interior points away from the boundaries and spread from there [private communication with O. Østerby, 1996]. However, boundary conditions can affect stability, especially if there are derivative (or mixed) boundary conditions [28–32]. It might be safer to consider all points in space in some way. The following somewhat brief treatment is described in greater mathematical detail in such texts as Smith [16] or Lapidus and Pinder [1].

#### 15.3.3.1 Using Eigenvalues

Equation (15.5), when discretised for, say, a potential jump experiment (Cottrell), gives rise to a system of *ode*s, depending on the method of discretisation used. For example, using the EX method, we have for the *i*th equation

$$\epsilon_i' - \epsilon_i = \lambda\left(\epsilon_{i-1} - 2\epsilon_i + \epsilon_{i+1}\right) \tag{15.16}$$

and recalling that at the electrode and the outer boundary in the bulk, $\epsilon$ is zero, we can write the whole system (15.16) in the form

$$\boldsymbol{\epsilon}' - \boldsymbol{\epsilon} = \lambda \mathbf{A} \boldsymbol{\epsilon} \tag{15.17}$$

with concentrations (or errors) now represented as vectors, and $\mathbf{A}$ being the matrix of coefficients, in this case given by

$$\mathbf{A} \equiv \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} . \tag{15.18}$$

The equation can be rearranged explicitly for $\boldsymbol{\epsilon}'$,

$$\boldsymbol{\epsilon}' = \mathbf{P}\boldsymbol{\epsilon} \tag{15.19}$$

with

$$\mathbf{P} = [\mathbf{I} + \lambda \mathbf{A}] . \tag{15.20}$$

One way of describing the simulation is as a series of multiplications of the error vector with the propagation matrix $\mathbf{P}$. The matrix method of stability analysis focusses on $\mathbf{P}$ and its effect on the whole error vector. That vector must not grow without limit, and to ensure this, there are some related conditions. One of them is, that that eigenvalue of $\mathbf{P}$ having the largest magnitude must not exceed unity in magnitude. In fact, in this particular case (see Smith [16] or any similar text), the eigenvalues are known in analytical form, and this leads again to the condition on $\lambda$, that is, $\lambda \le 0.5$. Not all cases of simulation methods lead to propagation matrices with analytical eigenvalues, and in these cases, they must be found numerically.

They can be complex, as is the case, for example, for methods like BDF using 3 or more points. Some examples are now given.

One convenient way to illustrate stability is to plot the eigenvalue of maximum magnitude in the complex plane for a number of $\lambda$ values. This provides the so-called spectral radius of the method. In the following examples, a value of $N = 20$ was used throughout. The actual number of spatial points makes little difference to the look of the diagrams to follow, and the chosen value was small enough for short computation times. Generally, the range of $\lambda$ was chosen as $10^{-6} \le \lambda \le 10^{6}$, in a geometric sequence, except for method EX, where the range was narrowed to around the stability limit, $0.49 \le \lambda \le 0.51$. Consider Fig. 15.5. Maximum-magnitude eigenvalues were calculated for matrix $\mathbf{P}$ as in (15.20). The points on the right of the figure are those for $\lambda$ up to (and including) 0.5. For greater values, the eigenvalues jump to the negative real end and fall outside the unity circle. That is, in the parlance of numerical analysis, the spectral radius exceeds unity. This means that errors will increase in magnitude, and also oscillate as they do so. This is exactly what one sees in practice, where increasing oscillations in concentrations are observed.

A method known to be stable is BI. The discretisation for the errors is

$$\epsilon_i' - \epsilon_i = \lambda\left(\epsilon_{i-1}' - 2\epsilon_i' + \epsilon_{i+1}'\right) \tag{15.21}$$

which forms the system

$$[\mathbf{I} - \lambda\mathbf{A}]\boldsymbol{\epsilon}' = \mathbf{P}\boldsymbol{\epsilon} \tag{15.22}$$

leading to the propagation matrix

$$\mathbf{P} = [\mathbf{I} - \lambda\mathbf{A}]^{-1} \,. \tag{15.23}$$

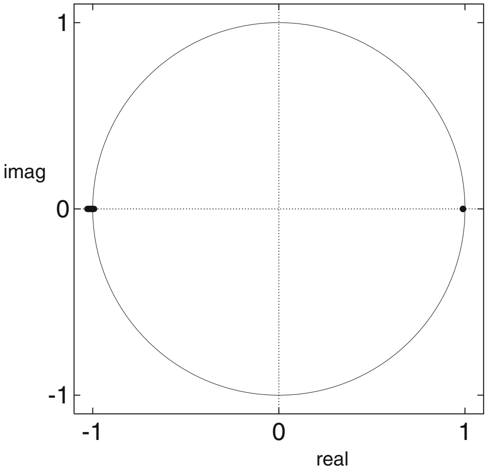


**Fig. 15.5** Maximum absolute eigenvalues for EX for $0.49 \le \lambda \le 0.51$ ($N = 20$)

**Fig. 15.6** Maximum absolute eigenvalues for BI for $10^{-6} \leq \lambda \leq 10^6$ ($N = 20$)

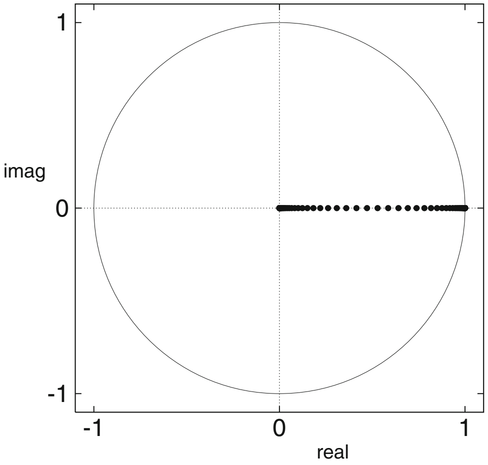


**Fig. 15.7** Maximum absolute eigenvalues for CN for $10^{-6} \leq \lambda \leq 10^6$ ($N = 20$)

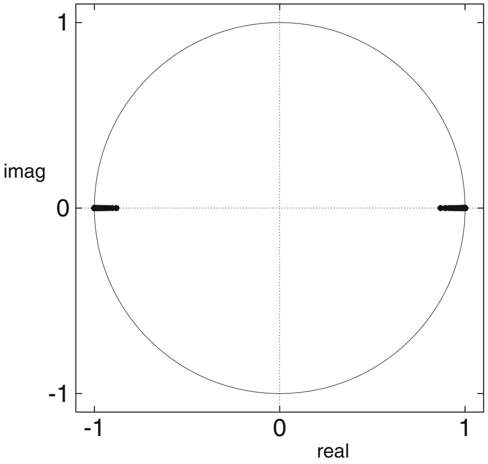


Figure 15.6 shows the spectral radii for this, for the range $10^{-6} \leq \lambda \leq 10^6$, in a logarithmic sequence. For increasing $\lambda$ the points move from right to left in the figure, towards the centre (0,0). All eigenvalues are real, and all fall inside the unit circle, confirming the stability of BI. Since the eigenvalues approach zero with increasing $\lambda$, errors will be damped more effectively for high $\lambda$ values.

Another method worth considering is CN, and Fig. 15.7 shows the result. Here, discretisation is

$$\epsilon_i' - \epsilon_i = \frac{\lambda}{2}\left(\epsilon_{i-1}' - 2\epsilon_i' + \epsilon_{i+1}'\right) + \frac{\lambda}{2}\left(\epsilon_{i-1} - 2\epsilon_i + \epsilon_{i+1}\right) \tag{15.24}$$

which forms the system

$$[\mathbf{I} - \frac{\lambda}{2}\mathbf{A}]\boldsymbol{\epsilon}' = [\mathbf{I} + \frac{\lambda}{2}\mathbf{A}]\boldsymbol{\epsilon} \tag{15.25}$$

leading to the propagation matrix

$$\mathbf{P} = [\mathbf{I} - \frac{\lambda}{2}\mathbf{A}]^{-1}[\mathbf{I} + \frac{\lambda}{2}\mathbf{A}] \ . \tag{15.26}$$

Again, for small $\lambda$ values, the points move from the right, being close to unity, towards the left. Unlike with BI, however, they do not approach zero but at some $\lambda$ value, depending on $N$, they cross to the negative side and approach $-1$. This underlines the oscillatory behaviour of CN, especially for large $\lambda$.

The methods considered above all show purely real eigenvalues. One method for which they are complex is BDF. For three-point BDF, the discretisation is

$${}'\epsilon_i - 4\epsilon_i + 3\epsilon'_i = 2\lambda\left(\epsilon'_{i-1} - 2\epsilon'_i + \epsilon'_{i+1}\right) \tag{15.27}$$

(recalling that ${}'\epsilon$ stands for $\epsilon(T - \delta T)$). This is a slightly more complex situation. The system can be written in vector-matrix form as

$$[3\mathbf{I} - 2\lambda\mathbf{A}]\boldsymbol{\epsilon}' = 4\boldsymbol{\epsilon} -{}'\boldsymbol{\epsilon} \tag{15.28}$$

and clearly, propagation is now not simply a matter of multiplying a single vector by a propagation matrix; we have two old vectors to deal with, ${}'\boldsymbol{\epsilon}$ and $\boldsymbol{\epsilon}$. First we write (15.28) in the usual form,

$$\boldsymbol{\epsilon}' = [3\mathbf{I} - 2\lambda\mathbf{A}]^{-1}(4\boldsymbol{\epsilon} -{}'\boldsymbol{\epsilon}) \ . \tag{15.29}$$

which we write, letting $\mathbf{B} = [3\mathbf{I} - 2\lambda\mathbf{A}]^{-1}$, as

$$\boldsymbol{\epsilon}' = \mathbf{B}(4\boldsymbol{\epsilon} -{}'\boldsymbol{\epsilon}) \ . \tag{15.30}$$

This will be needed later. We now need another equation. First, consider that when using BDF, we calculate the new vector from the two old ones, and the previous present one becomes the previous older one. This can be expressed as a vector of vectors operation. Let the new double-length vector $\mathbf{e} \equiv [\boldsymbol{\epsilon} \quad {}'\boldsymbol{\epsilon}]^T$, consisting of the two known vectors, and the new one, $\mathbf{e}' \equiv [\boldsymbol{\epsilon}' \quad \boldsymbol{\epsilon}]^T$, the present one and the new one, to be calculated. To express this mathematically as a propagation from $\mathbf{e}$ to $\mathbf{e}'$, we need one more equation, and augment system (15.29) to the system of systems

$$\begin{aligned} \mathbf{e}' &= \mathbf{B}(4\boldsymbol{\epsilon} -{}'\boldsymbol{\epsilon}) \\ \mathbf{e} &= \mathbf{e} \ . \end{aligned} \tag{15.31}$$

The seemingly redundant second equation now enables us to write this in matrices-in-matrix form:

$$\left[\mathbf{e}'\right] = \begin{bmatrix} 4\mathbf{B} & -\mathbf{B} \\ \mathbf{I} & 0 \end{bmatrix} \mathbf{e} \tag{15.32}$$

where clearly the propagation matrix **P** is the $2 \times 2$ matrix of matrices. This can be expanded into a larger $2N \times 2N$ matrix, and eigenvalues computed. Figure 15.8 shows the result. As with BI, the eigenvalues for small $\lambda$ are at the right of the figure, close to unity and real initially and move left as $\lambda$ increases. At some $\lambda$ value, depending on $N$, the eigenvalues become complex and pair up into complex conjugates, following, with increasing $\lambda$, a circular path towards the centre of the circle. So, as for BI, high $\lambda$ means effective error damping, but the damping factor is complex. Complex propagation factors imply the possibility of oscillations with periods perhaps of several time intervals. For higher-point BDF variants, the points do not follow a circular path, but pairs of arcs with increasing magnitude, with some values in the left-hand plane; and eventually, for more than 7 points, reaching outside the unit circle, indicating instability [33].

This illustrates how diagrams of spectral radii can provide information on the stability of a given method.

So far, the analysis has been for the Cottrell method. It is of interest to see how it changes for derivative boundary conditions. This only changes a single line in the discrete system of equations. For constant current, for example, we might use a two-point expression

$$\epsilon_1 - \epsilon_0 = 0 \tag{15.33}$$

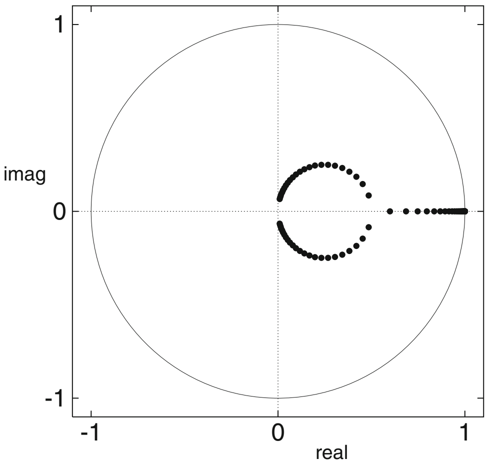


**Fig. 15.8** Maximum absolute eigenvalues for three-point BDF for $10^{-6} \leq \lambda \leq 10^{6}$ ($N = 20$)

(there is no current for errors at the electrode) or

$$\epsilon_0 = \epsilon_1 \tag{15.34}$$

and this changes the first equation for EX to

$$\epsilon_1' = \epsilon_1 + \lambda(\epsilon_1 - 2\epsilon_1 + \epsilon_2) \tag{15.35}$$

which makes the corresponding coefficient matrix

$$\mathbf{A} \equiv \begin{bmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix}. \tag{15.36}$$

The analysis is otherwise the same, but the effects are not, as will be discussed below. Extension to a greater number of points to express the derivative is obvious.

#### 15.3.3.2 Using the Matrix Norm

A different but mathematically equivalent way to perform the matrix analysis is to use matrix norms rather than eigenvalues. Smith [16] states that the condition for all eigenvalues not to exceed unity in magnitude is equivalent to the demand that the infinity norm of the propagation matrix **P** does not exceed unity. That norm can be more easily computed than eigenvalues. In fact, all norm definitions can be used for this purpose. The infinity norm of a matrix is defined as the largest of the sums of the moduli of the row values. So this is another criterion for stability. It is, however, a little more complicated than the use of eigenvalues. For a method such as BI, we find that for all $\lambda$ values, the norms are a little less than unity. For CN and BDF3, however, the norms can be greater than unity and this might cause concern. However, a slightly more relaxed condition for stability is that the norms of increasing multiples of the propagation matrix do not increase indefinitely. The limit of this series of norms may be greater than unity, but it must exist. If this condition holds, then errors will not build up indefinitely, and the method is stable. Such norms exceeding unity have been found for certain electrochemical simulations [31], for CN under mixed boundary conditions. So, although norms are easier (and faster) to compute than eigenvalues, their use is not as straight-forward as the use of eigenvalues.

### 15.3.4 Some Special Cases

There are some special conditions in electrochemical simulations that have an effect on stability.

Bieniasz and Britz [34–37] investigated the effect of the presence and rate of homogeneous chemical reaction (hcr) accompanying electron transfer and found some effects. Writing $\xi = K\delta T$ ($K$ being the dimensionless first-order hcr rate constant as in Chap. 5, Eq. (5.12)), Bieniasz found [36] that for method EX, using the parallel discretisation as in (5.13), the upper permissible $\lambda$ value for stability decreases linearly with $\xi$, and reaches zero for $\xi = 2$. For larger $\xi$ values and this discretisation, EX is always unstable. For the sequential discretisation (see Sect. 5.4, page 93 and Appendix C, page 455), in which the chemical reaction is allowed to act on concentrations that have undergone diffusional changes, the situation is more complex but the same upper limit on $\xi$ holds. A similar stability decrease and the same limit on $\xi$ was found for explicit RK. Stable methods such as CN or Saul'yev were unaffected by a hcr in the sense that they remained stable for all $\xi$ values. These findings were then confirmed numerically [37].

Another factor affecting stability is a derivative boundary condition. Keast and Mitchell pointed out potential problems with CN in this regard [32], and investigations in the electrochemical context revealed some problems with methods otherwise thought to be unconditionally stable, such as CN and Saul'yev [28–30]. The CN method was found to become unstable for $\lambda > 4$ if a heterogeneous rate constant setting a derivative boundary condition decreased with time [28], as happens, for example, in linear sweep experiments on a quasireversible system. Also, certain (unusual) ways to discretise the derivative boundary conditions can render CN unstable [31]. Fortunately, these effects are seen under somewhat extreme conditions; indeed, they are difficult to demonstrate numerically, so they might be considered of only academic interest.

## 15.4 The Stability Function

Another way of investigating stability, that at the same time provides information on the behaviour of a given method, is what Gourlay and Morris [38] call the symbol of the algorithm, also called the symbol of the method [16] or, more logically perhaps, the stability function [17]. The various authors employ various ways of expressing the function (or symbol), all equivalent. It is developed from Padé approximations to the general solution of the diffusion equation. Equation (15.5) can be semidiscretised to the system of *ode*s as

$$\frac{\mathrm{d}\boldsymbol{\epsilon}}{\mathrm{d}T} = \frac{1}{H^2}\mathbf{A}\boldsymbol{\epsilon} \tag{15.37}$$

with $\mathbf{A}$ defined in (15.18) for the Cottrell experiment. At this stage, a particular simulation method, that is, a time-integration algorithm, has not been specified. The general solution of this, analogously with the plain *ode* $y' = -ay$ is

$$\boldsymbol{\epsilon}(T) = \exp(\tfrac{T}{H^2}\mathbf{A}) \tag{15.38}$$

where the exponential is understood as its series definition, for some matrix $\mathbf{M}$,

$$\exp(\mathbf{M}) = \mathbf{I} + \mathbf{M} + \frac{\mathbf{M}^2}{2!} + \frac{\mathbf{M}^3}{3!} + \dots \,. \tag{15.39}$$

This leads, for a given step of length $\delta T$, to the solution

$$\boldsymbol{\epsilon}' = \exp(\lambda\mathbf{A})\boldsymbol{\epsilon} \,. \tag{15.40}$$

It is at this point that the various Padé approximations to the exponential functions come in. One of them is simply the sequence on the right-hand side of (15.39) cut off after the first two terms, producing the propagation equation,

$$\boldsymbol{\epsilon}' = (\mathbf{I} + \lambda\mathbf{A})\boldsymbol{\epsilon} \tag{15.41}$$

which is clearly the method EX, and the truncated series is the (0,1) Padé approximation. Smith [16] tabulates a number of these approximations. For example, the (1,0) variant generates

$$\boldsymbol{\epsilon}' = \frac{1}{(\mathbf{I} - \lambda\mathbf{A})}\boldsymbol{\epsilon} \tag{15.42}$$

which is seen to be the BI method. There are a number of approximants, the (1,1) one leading to CN. The various algorithms have in this way been unified under one system.

The next step is to consider the eigenvalues of $\mathbf{A}$ in the propagation formulae such as (15.41) and (15.42). These are all negative, and we set $z = -\lambda\omega$ following Gourlay and Morris [38] and Strutwolf and Schoeller [39], where $\omega$ are the eigenvalues, all negative. Since $-\lambda\omega$ can have any value greater than zero, we can dissect a particular propagation equation in terms of $z$. Thus, for method EX, we have for the stability function $R(z)$,

$$R(z) = 1 - z \tag{15.43}$$

which decreases linearly with $z$ and goes below $-1$ for $z > 2$. For a maximum magnitude eigenvalue equal to $-4$ (which is the case), this goes back to the known stability criterion $\lambda \leq 0.5$.

For BI, (15.42) becomes the stability function

$$R(z) = \frac{1}{1+z} \tag{15.44}$$

and for CN (using the (1,1) Padé form [16]), it is

$$R(z) = \frac{1-\frac{z}{2}}{1+\frac{z}{2}}\,. \tag{15.45}$$

An interesting case is extrapolation, and we consider the simplest variant, the second-order case. It is a number of steps using BI. First one takes two successive steps with half step size, and subtracts from twice the result of this the result of one whole step. Thus, we can directly write [39].

$$R(z) = \frac{2}{\left(1+\frac{z}{2}\right)^2} - \frac{1}{1+z} \tag{15.46}$$

which expands to

$$R(z) = \frac{1+z-\frac{1}{4}z^2}{1+2z+\frac{5}{4}z^2+\frac{1}{4}z^3}\,. \tag{15.47}$$

Figure 15.9 shows the result for the three methods BI, CN and second-order extrapolation. Note first the smooth but rather slow decline to zero of BI. Note next that CN crosses zero at some $z$, and therefore, some $\lambda$, and approaches $-1$ for large arguments. This illustrates what we know about CN, that is it becomes more and more oscillatory for large $\lambda$. Finally, second-order extrapolation also crosses zero,

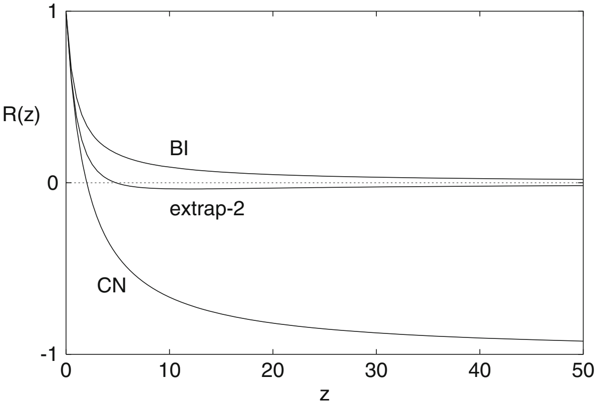


**Fig. 15.9** Stability function $R(z)$ for the marked methods

but then approaches zero again, and of the three methods, it converges most rapidly to zero. This is an attractive feature. It remains to be seen, however, how the method compares with others in terms of efficiency.

## 15.5 Accuracy Order

In other chapters, the **order** of accuracy of various methods is referred to. Here this concept is defined and two methods of calculating the order are presented.

A simulation results in a number (or a vector of numbers) at some time. Depending on the dimensionality of the problem, the simulation uses intervals in time $\delta T$ and one or more space intervals. Often there is only one space interval, here given the symbol $H$. A result—a current, or a concentration, for example—will, due to truncation errors, have an error associated with it, that can be expressed in the following way. The discussion is, for the moment, restricted to an *ode* with interval size $h$. Then the simulated result at time $t$ can be written as a polynomial

$$u = \hat{u} + a\,h + b\,h^2 + \ldots \tag{15.48}$$

where $\hat{u}$ is the underlying true solution (known or not, see below), and $a, b, \ldots$ are constants. For a given method, the polynomial will be dominated by a certain power term. Let this term be the power $p$, so that the above equation can be written as

$$u = \hat{u} + O(h^p)\ . \tag{15.49}$$

The number $p$ is of great interest, more so than the constants, which are generally unknown and are usually unimportant when deciding on a given method. This is because high order accuracy means that if we decrease $h$, we dramatically improve the accuracy. Conversely, this is not the case for a small $p$. So, first-order methods such as EX or BI mean that we must decrease the intervals greatly in order to achieve some target accuracy, implying perhaps unacceptable computing time. However, this is not always the only criterion; a given high-order method might require so much computing time, that a lower-order method is preferable.

### *15.5.1 Order Determination*

There are two ways of determining the order of a method, depending upon whether we have an exact solution to compare with, or not.

If we know an exact solution, Method 1 is used. First a result $u_1$ is calculated, using interval size $h$. From (15.48), we have

$$u_1 = \hat{u} + a\,h + b\,h^2 + \ldots\ . \tag{15.50}$$

The simulation is then repeated at a new interval size, $\alpha h$, so that the new result is

$$u_2 = \hat{u} + a\alpha\,h + b\alpha^2\,h^2 + \ldots\ . \tag{15.51}$$

Usually, $\alpha$ is chosen equal to 2. We can now calculate errors associated with the approximate solutions,

$$e_1 = u_1 - \hat{u} = a\,h + b\,h^2 + \ldots \tag{15.52}$$

and similarly for $e_2$. Dividing $e_2$ by $e_1$, we have

$$\frac{e_2}{e_1} = \frac{a\alpha\,h + b\alpha^2\,h^2 + \ldots}{a\,h + b\,h^2 + \ldots}\ . \tag{15.53}$$

If the dominant term is that with $h$, the ones in higher powers can be dropped and we get

$$\frac{e_2}{e_1} \approx \alpha\ . \tag{15.54}$$

If the dominant term is that in $h^2$, all others are dropped and we get

$$\frac{e_2}{e_1} \approx \alpha^2\ . \tag{15.55}$$

The order, expressed as the power in $h$, is then equal to $\log_\alpha(e_2/e_1)$. As mentioned above, usually the factor 2 is conveniently chosen for $\alpha$.

If we do not know an exact solution, we can still estimate the error order by Method 2, as described by Østerby [40]. We must use one more interval size, $\alpha^2 h$. We then have a third result:

$$u_3 = \hat{u} + a^2\alpha\,h + b\alpha^4\,h^2 + \ldots\ . \tag{15.56}$$

Instead of using errors (we do not know them), we use

$$\frac{u_3 - u_2}{u_2 - u_1} = \frac{\alpha(\alpha-1)\,ah + \alpha^2(\alpha^2-1)\,bh^2}{(\alpha-1)\,ah + (\alpha^2-1)\,bh^2} \tag{15.57}$$

and again, we obtain the factor $\alpha$ for a first-order result and $\alpha^2$ for a second-order result, etc. The unknown exact solution is subtracted out in this process. Often, this method yields less clear results than the first method requiring only two simulations, particularly if the actual errors are large.

In the preceding, only a single interval has been considered. In electrochemical simulations, there are at least two: in time and space; and there may be more, if the simulation is in more than one dimension. In such cases, the various intervals might affect the accuracy to different orders. Thus, both methods EX and BI have concentration accuracies of order $O(\delta T, H^2)$, while CN has $O(\delta T^2, H^2)$. It is clear now that EX and BI lack a high-order time derivative. It is also clear that the order does not tell the whole story; we know that CN has a higher accuracy order than both EX and BI, but we also know that it is oscillatory for the first so many steps. The order tells us nothing about this. When measuring the order of a method with several different interval sizes, one must (obviously) not only vary the one being measured, keeping the others constant; one must also ensure that the other interval sizes are rather small, so that the accuracy of the result is not limited by them. Otherwise, the order calculation will not be as clear.

One might also suspect that a higher-order method might still have large errors compared with some lower-order method, because we ignore the polynomial constants, and they are surely different for different methods. This is so but in all practical cases, the constants do not affect the accuracy as strongly as the order itself, so a higher-order method always leads to smaller errors. It might, however, do so at the expense of more computing time, and this is gone into next.

Another aspect is that of starting a given method with another. For example, one way to start BDF is to use BI for the first step. A single BI step (see Chap. 3) is second-order accurate with respect to the time interval. The next step is BDF3 (three-point BDF), and so on. One might expect that the second-order error introduced by the first BI step can be "diluted" by subsequent higher-order steps. However, this is not so. Normally, the largest errors appear at the start of a simulation, and remain to contaminate the result. So the described procedure, the rational start, yields second-order accuracy, no matter what order BDF one uses. This is why, when using this start, one might as well use three-point BDF. Higher-order BDF forms do not improve the result, as is seen in Table 15.1. In that table, the simple *ode*, $y' = -y$ was simulated, using 100 steps of interval size 0.01. Four ways of handling BDF were used: the simple start without any time correction ("simp"); the simple start with correction by half a time interval ("simp+"); the rational start, working up from BI as described ("rat"), and lastly, adapting the Kimble and White (KW) method to provide a high-order start, as described on page 76. The orders were calculated using Method 1. Method 2 was also used and gave the same results. The actual errors are also shown. The same problem was then simulated using extrapolation (here meaning specifically BI with extrapolation) with orders 2 . . . 4, as described in Sect. 4.9 on page 72, with the same interval size and number of steps.

**Table 15.1** Error orders for some BDF starts, for the *ode* $y' = -y$

| Start | $k = 2$ (BI) | $k = 3$ | $k = 4$ | $k = 5$ |
|---|---|---|---|---|
| simp | 1.00 (−2.7) | 1.01 (−2.7) | 1.00 (−2.7) | 1.00 (−2.7) |
| simp+ | 1.99 (−5.2) | 2.01 (−4.7) | 1.98 (−5.1) | 2.00 (−4.1) |
| rat | 1.00 (−2.7) | 2.00 (−4.8) | 1.99 (−4.6) | 2.00 (−4.6) |
| KW | – (−2.3) | 1.98 (−4.9) | 3.00 (−7.0) | 3.97 (−9.1) |

The numbers in brackets are the log(abs(errors))

**Table 15.2** Calculated orders for the *ode* $y' = -y$ using extrapolation at various orders and log(abs(errors)) in brackets

| Extrapolation order | Accuracy order (error) |
|---|---|
| 2 | 1.99 (−5.2) |
| 3 | 2.99 (−7.6) |
| 4 | 3.97 (−10.2) |

Table 15.2 shows the results. In Table 15.1, there is one anomaly, for BI ($k = 2$) with the time correction. This is not expected to be a good method but turned out second-order accurate with a comparatively very small error. This must be regarded as fortuitous, due to time-varying time shifts [41] and perhaps to the particular *ode* used as test case. The other results are as expected. Note the order 2 for the rational start for all orders BDF, and the impressive high orders for KW, and corresponding increasingly smaller errors, for the KW start. Note also that the calculated orders are not exact integers, due to some uncertainties, or due to terms other than the dominant order terms playing a small part. Table 15.2 also shows expected results, and shows that extrapolation at high order results in conveniently small errors.

Finally, for two-dimensional (UMDE) simulations, Gavaghan found [42] an $O(h^{1/2})$ for errors in concentration at edge points.

## 15.6 Sensitivity Analysis

Very briefly, this rather large subject in the general area of chemical kinetics [43–45] was carried into electrochemistry in the studies by Bieniasz et al. [46–48]. It asks the question, when fitting some parameter to a proposed mechanism by means of simulation using some simulation output (concentrations or current or some other result), how sensitive to the changes in the output is the value of the fitted parameter. This is expressed in the form of a sensitivity function $s$. If the simulation yields, for example, an array of concentrations $c(x, t, p)$, where $x$ are positions in space, $t$ the time (which may enter the problem) and the parameter(s) $p$, then the function is defined [46] as $s = \partial c/\partial p$, which is an expression of the sensitivity to changes in concentration. This can be useful in estimating the reliability of fitted parameters by a series of simulations. This subject will not be persued further here.

## 15.7 Accuracy, Efficiency and Choice

We come now to the choice of method. There are no hard and fast rules here, the final choice depending to a large extent on personal preference and the inclination towards programming. Computers are now so fast that all but hard simulation problems such as CVs of, say, 2D problems or 3D problems execute in a very short time—usually just a few seconds. In such a case, the main bottleneck will be the programming itself, including finding the initial programming errors. If a given method results in a savings of a second or so, it might not be worth the extra effort in terms of paper work and programming. However, there are some rough guidelines we can provide here.

First of all, we must be clear about what we mean by accuracy, or what we wish to be accurate. Usually, the result of a simulation is a current function of time, so one argument might be, that we do not care about concentrations, but the current must be accurate [49]. There is the possibility of error cancelling, which can lead to small errors in a current, while the concentration errors are larger (in a relative sense). This may not be reliable for all parameter choices. Empirical adjustments have also been known. Thus, Feldberg in his seminal chapter [9] introduced the doubtful device of subtracting half a time interval from all $t$ values in a simulation using the box method (EX). Although there was—and is—no justification for this, it seemed to give better results. The practice has been followed by others, for example in the text of Bard and Faulkner [7], as a sort of tradition. If one uses better methods than explicit, then it becomes clear that this device is not appropriate—except in the case of BDF, using the simple start. In their original work on the BDF method, Mocak and Feldberg advocated the subtraction of half a time interval. This was examined in some detail [41, 50], and it was found that it indeed leads to a dramatic improvement in accuracy (see also Table 15.1). This indicated that there was in fact a justification for the device, and it was then shown to be so [51, 52]. This is a remarkable coincidence, and the only case in which this device is reasonable. For other cases such as method EX the device must be regarded as an empirical adjustment, and positive results from using it as fortuitous.

The numerical solution produces concentration values, and one must therefore strive to obtain as accurate values for these as possible, so that currents calculated from them might also be accurate. Bieniasz now makes a practice of showing errors across the whole concentration profile, when reporting a new simulation method [53–55], or at least a few samples from the profile [56].

Accuracy alone is not a sufficient criterion for a good method, however. One can, for example, usually drive method EX to any target accuracy, by simply refining the time and spatial intervals sufficiently. The result may then require excessive computer time, but not always. The target accuracy itself is subject to discussion. It must be kept in mind that the computed currents are normally compared with experimentally measured values, in order to obtain some experimental parameters. The measurements can rarely be carried out with better than 1 % accuracy, because they rely on components such as resistors, etc. which have, at best, that level of accuracy. So one might set a target accuracy at a relative $10^{-3}$. Then, one measures the computing time needed to achieve that target accuracy. The most efficient method is then the one using the least computing time. This will rarely be the method that provides the best accuracy for a given space/time grid. For example, the KW start for BDF (see Table 15.1 and Sect. 4.10.1) clearly provides a high-order start and results in impressively small errors for the higher-order BDF variants. However, applied to electrochemical digital simulation, it was found to be inefficient [50], and it was found that for BDF, the most efficient method is the simple start with subsequent subtraction of half a time interval. Thus, we seek efficient methods that minimise computing time for a target accuracy.

Another factor is programming effort, including planning on paper. The KW start mentioned above is a case in point. It is not trivial to implement for a *pde* in a computer program, and we might consider ourselves lucky that it was shown to be inefficient. The same might be thought of the Rosenbrock methods, which are less easy to program than, say, BDF or the extrapolation variants, and also for OC. This is to a large extent subjective. Someone who has worked with these harder methods such as Rosenbrock and OC might not consider them hard to program, and will then achieve good execution efficiency from them, whereas others will stick to easier methods. Here, the discussion will be restricted to a chosen few methods, regarded, by the present authors, as good compromises between programming effort and efficiency.

CN is not among this group. As described earlier in Chap. 8, CN leads to initial oscillations. These can be damped by several devices [3, 6], but these are either not very effective, or demand increased programming effort. Thus, the Pearson method [57], in which the first step is simply subdivided into a number of equal substeps such that each one corresponds to a $\lambda$ value of around unity, will obviously mean very many substeps for large $\lambda$, although it is easy to implement from a programming point of view. Less easy is the use of unequal substeps, but here some experience is necessary in order to choose the most suitable expansion factor [3] (and may necessitate an LU decomposition at every substep). Lastly, the effective device of taking one or more steps using BI, then following with CN as suggested [3, 58], necessitates the addition of the BI method to a given CN program. Some might consider this too much effort. Given that there are other implicit methods, as easy to program as CN, that are also as efficient, we can exclude CN from our menu of choices.

The two methods that stand out in terms of efficiency and convenience are BDF and BI with extrapolation. Both require minimal programming effort, and can be extended to higher-order spatial derivatives. However, in the case of BDF, a limit is encountered. For the most convenient start-up methods such as the simple or the rational start, the accuracy from BDF is limited to $O(\delta T^2)$. This means for one thing that one need not go beyond three-point BDF (which is $O(\delta T^2)$ in itself), but that no marked improvement can be gained from higher-order spatial derivative approximations, because there will then be a mismatch between the accuracy orders with respect to the time and spatial intervals.

BI with extrapolation does require extra concentration arrays (as does BDF) leading to less convenient programming, but higher-order extrapolation can rather easily be achieved, and thus the use of higher-order spatial derivative approximations can be used to advantage.

It has been found by numerical experiment [50] that these two methods are about the most efficient, when combined (especially in the case of extrapolation) with higher-order spatial derivative approximations. The variants found to be best are three-point BDF using the rational start or the simple start and with subtraction of half a time interval, and extrapolation, possibly with the higher spatial derivative orders. We find the simple start for BDF a little awkward however, as it produces values at offset time points, so we prefer the rational start with BI for BDF. This is a matter of choice.

A single example of a comparison is now presented. The Cottrell system was simulated, using unequal intervals with 40 points and a smallest spatial interval equal to 0.01. This gives an expansion factor $\gamma$ of 1.11, quite moderate. Two simulation methods were chosen. Second-order extrapolation was chosen, being quite simple to program, as well as second-order BDF, using the rational start. The simulation was run with various time intervals (expressed as $N_T$, the number of steps to $T = 1$), and the error in the current at that time measured, as a three-point approximation on the nonuniform grid. The second spatial derivative was approximated by a four-point formula, which allows the use of a modified Thomas algorithm, see Sect. 8.4 on page 151. In the figures, this is given as the logarithm of the relative absolute error. That is, if the simulated current be $i_{sim}$ and the analytically known value be $i_{anal}$, then what is plotted (marked in the figures as `"log|err|"`) is the quantity $\log_{10}((i_{sim} - i_{anal})/i_{anal})$. Figure 15.10 shows the results. Note that both curves show a similar accuracy at small $N_T$. If one measures the gradient there, it verifies the expectation of an accuracy order of $O(\delta T^2)$. However, extrapolation then shows a sharp dip, followed by an approach to a constant error. The dip is due to the actual error crossing the zero line. BDF simply flattens out and approaches a slightly smaller constant error. This arises because with decreasing time intervals per step, the error from the second spatial derivative approximation makes itself felt more and more. The curve for BDF does not show the dip. Looking at the two curves, one might consider both methods as roughly equal, BDF perhaps being a little better, in terms of accuracy. However, what interests us is the computing time (cpu) used. This was also measured and is shown in Fig. 15.11. The times are very short, measured in milliseconds, so they were measured by letting the working parts of the programs repeatedly execute a sufficient number of times, so that an accurate time could be measured. Now a difference is noted: for very small $N_T$ the

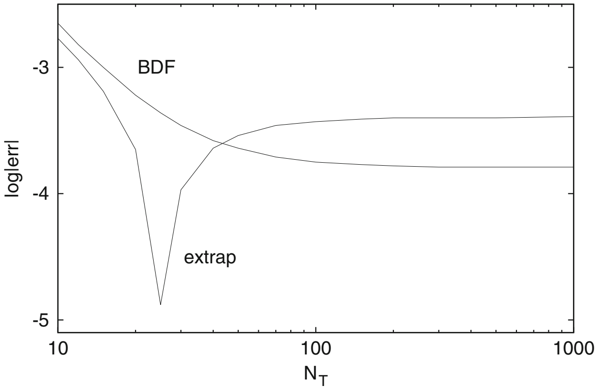


**Fig. 15.10** $\log_{10}((i_{sim} - i_{anal})/i_{anal})$ vs $N_T$ for BI with extrapolation and BDF (both second-order)

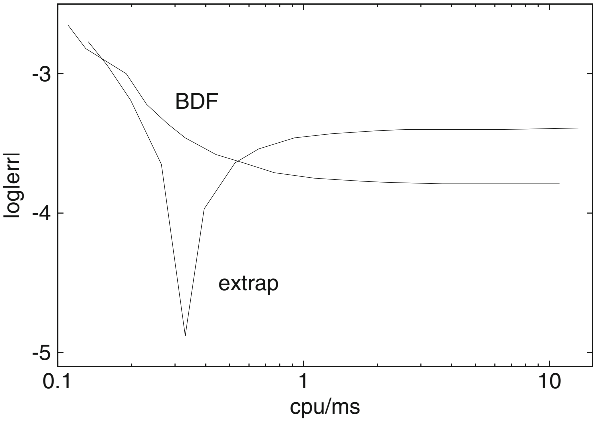


**Fig. 15.11** $\log_{10}((i_{sim} - i_{anal})/i_{anal})$ vs cpu time for extrapolation and BDF (both second-order)

two methods reach the $10^{-3}$ mark at about the same cpu time, but for larger $N_T$ BDF is more accurate and BI/extrap never reaches $10^{-4}$. This low error level might, however, be uninteresting to most. If we choose some target accuracy, for example $10^{-3}$ or $-3$ on the log-scale, then both methods are equally good.

### *15.7.1 Determining Accuracy*

If we do not have an analytical solution, but still wish to estimate the error of a simulation, convergence runs might be the answer. Having established that a given program works (see the next chapter), it can be run with increasingly finer intervals in both time and space until the results do not change significantly further, and use these values as a standard. This was done for the diffusion limited potential step simulation at an ultramicrodisk, ultramicroband and a capped or finite cylinder [59–61], providing tables of reference current values over large ranges of time, from which values at any time can be obtained by interpolation. The relevant subroutines are provided in Appendix E. Bieniasz has recently published [62] his method of solution for currents near a band edge and a band, which can provide values with machine accuracy, albeit at the cost of extensive computations. This work was followed by another providing approximation constants, making it possible to compute such currents in a very short time, with the same accuracy [63].

Another way is what has been called algorithms with guaranteed accuracy, by estimating the error by a process similar to extrapolation, subdividing intervals and estimating the error from the change in the output. Nann and Heinze pioneered this approach for the disk [64] and scanning tunnelling electrochemical microscope [65]. Harriman et al. published a series of papers describing their FEM method doing this for the disk level with the insulating plane [66, 67], for recessed or elevated disks [68], the channel [69] and band [70], as well as the LSV experiment [71], and Gavghan et al. followed with another study in 2006 [72]. The work of Bieniasz on his patch-adaptive strategy, estimating the local error by means of repeatedly locally halving intervals, until the estimated errors all are small enough, falls into the same category, providing values of approximately known accuracy [73–78].

## 15.8 Two- (and Three-)Dimensional Problems

The above applies largely to one-dimensional problems, but two-dimensional simulations are more and more the norm. All the methods described can be applied to these, but there are some sensible choices. For the ultramicrodisk electrode case, the present authors have recently done a detailed study and concluded [79] that one of the more useful transformations such as the Verbrugge/Baker [80] (see Chap. 12 for this and others) is best, combined with multipoint approximations (optimally using five or seven points), and either BI with extrapolation or BDF with the BI start.

Another popular method (almost always with a transformation) is ADI, as this allows solution by a tridiagonal Thomas algorithm separately for the rows and columns. An ADI variant was also the choice for the three-dimensional problem of a rectangular electrode [81]. See Chap. 12 for more details.

For more complex geometries in two- and three-dimensional space FEM or BEM might be a choice, especially if one has access to commercial FEM software such as ANSYS® or COMSOL Multiphysics®. In any case, for commercial packages, convergency tests are still necessary to gain insight into the quality of the simulation.

## 15.9 Summary of Methods

Here, a brief summary is given of all those methods that might be of interest, with their advantages and disadvantages, as seen by the present authors. References are not given, as they are provided in the sections of the book that are referred to.

- **EX** Chap. 5. EX has limited stability and is accurate only to $O(\delta T)$. It is easy to program, but inefficient.
- **Explicit RK**: Chap. 4, Sect. 4.5. RK has limited stability and is marginally more efficient than EX. It is easy to program.
- **BI**: Chap. 4, Sect. 4.6 and Chap. 8, Sect. 8.1. BI is unconditionally stable but has, like EX, an accuracy of only $O(\delta T)$. It has a smooth response to initial transients and can be valuable as a first step before three-point BDF (p. 69) or as one or more first steps before CN to damp oscillations (p. 148), and is the basis for higher-order methods such as extrapolation and BDF.

- **CN**: Chap. 8, Sect. 8.2. CN has an accuracy of $O(\delta T^2, H^2)$, and is unconditionally stable. CN oscillates, however, with initial transients. With large $\lambda$, these oscillations persist over many steps, rendering CN useless. The oscillations can be damped (Sect. 8.5.1) but this defeats to some extent the simplicity of the method. For LSV or chronopotentiometry simulations, however, where no sharp initial transients occur, this might be a good method.
- **BDF**: Chaps. 4 and 8. BDF is a stable and non-oscillatory method that can be driven to higher-orders with respect to the time interval. Realistic starting strategies reduce the order to $O(\delta T^2)$, so the three-point variant is recommended, using the "rational" start of a single BI step, see Sect. 4.8.1. It is reasonably easy to program.
- **Extrapolation** Chap. 4, Sect. 4.9 and Chap. 8, Sect. 8.5.2.2. Like BDF, this is based on BI driven to a higher-order. Up to $O(\delta T^4)$ is feasible to program, and there are no starting problems. Extra steps are, however, required, and the preferred second-order method is about as efficient as second-order BDF.
- **Unequal intervals** Chap. 7. These are essential for most programs. The second spatial derivative requires four points if second-order is wanted (and is recommended). With four-point discretisation, an efficient extended Thomas algorithm can be used, obviating the need for a sparse solver. Very few points can then be used across the concentration profile. For two-dimensional simulations, direct three-point discretisation on the unequally spaced grid was shown to be comparable with using transformation and discretisation in transformed space.
- **Hopscotch** Chap. 9, Sect. 9.2.5. Hopscotch is stable and explicit but has the problem of "propagational inadequacy", so the ability to use large $\lambda$ values is lost. Nevertheless, hopscotch continues to be used.
- **Rosenbrock** Chap. 4, Sect. 4.12 and Chap. 9, Sect. 9.4. It is a stable implicit RK method, and using ROS2 or ROWDA3, it can be computed explicitly. ROWDA3 is exceptionally accurate at $O(\delta T^3)$ and is especially useful for nonlinear boundary conditions (e.g. page 203), where it obviates the need for Newton iterations. Rosenbrock is not, however, trivial to program and might be best left to programmers of professional simulation packages.
- **Higher-order methods** Chap. 9, Sect. 9.2.2 for multipoint discretisations. The four-point variant with unequal intervals is probably optimal; the system can be solved using an extended Thomas algorithm without difficulty. Numerov methods (Sect. 9.2.7) can achieve higher-orders with only three-point approximations to the spatial second derivative. They are not trivial to program.
- **DuFort/Frankel** Chap. 9, Sect. 9.2.3. It is stable and explicit but inconsistent and shares the propagational inadequacy problem.
- **Saul'yev** Chap. 9, Sect. 9.2.4. Saul'yev is stable and is calculated explicitly. If the RL and LR variants are combined, it produces results equal in accuracy and order to those from CN, including its propensity to oscillate in response to an initial transient. So like CN, this might be a good method for LSV simulations.
- **OC** Chap. 9, Sect. 9.6. OC produces impressively accurate results with only a few spatial points. It can be regarded as a kind of MOL, and although practitioners

tend to propose advanced techniques for solving the set of *ode*s, other methods can be used, such as a Rosenbrock method or BDF. The method is not used as much as one might expect.

- **MOL/DAE** Chap. 9, Sect. 9.3. A set of *pde*s is only spatially discretised, leaving a set of *ode*s to solve. Adding the boundary conditions then produces a set of DAEs, usually solved using a professional package such as DASSL. MOL seems to be more difficult to program than some other methods. It will probably find its greatest use in general simulation packages.
- **Box method** Chap. 9, Sect. 9.1. This is the original electrochemical simulation method. With boxes, most of the above techniques can be applied. There is an unresolved issue of whether this method is inherently better than the point method, or not.
- **FEM and the like** Chap. 9, Sect. 9.5. These are possibly the most efficient methods for more complex geometries, can be made adaptive to changing conditions, but are not trivial to program. Commercial packages (not specially dedicated to electrochemical systems) are available.

## References

1. Lapidus L, Pinder GF (1982) Numerical solution of partial differential equations in science and engineering. Wiley, New York
2. Gresho PM, Lee RL (1981) Don't suppress the wiggles - they're telling you something! Comput Fluids 9:223–253
3. Britz D, Østerby O, Strutwolf J (2003) Damping of Crank-Nicolson error oscillations. Comput Biol Chem 27:253–263
4. Martínez-Ortiz F, Molina A, Laborda E (2011) Electrochemical digital simulation with highly expanding grid four point discretization: can Crank-Nicolson uncouple diffusion and homogeneous chemical reactions? Electrochim Acta 56:5707–5716
5. Østerby O (2002) Five ways of reducing the Crank-Nicolson oscillations. Technical Report Daimi PB-558, Department of Computer Science, Aarhus University
6. Østerby O (2003) Five ways of reducing the Crank-Nicolson oscillations. BIT Numer Math 43:811–822
7. Bard AJ, Faulkner LR (2001) Electrochemical methods. Wiley, New York
8. Feldberg SW, Auerbach C (1964) Model for current reversal chronopotentiometry with second-order kinetic complications. Anal Chem 36:505–509
9. Feldberg SW (1969) Digital simulation: a general method for solving electrochemical diffusion-kinetic problems. In: Bard AJ (ed) Electroanalytical chemistry, vol 3. Marcel Dekker, New York, pp 199–296
10. Feldberg SW (1980) Improvements on computer simulation of electrochemical phenomena involving hydrodynamics: the rotating disk and dropping mercury electrode. J Electroanal Chem 109:69–82
11. DuFort EC, Frankel SP (1953) Stability conditions in the numerical treatment of parabolic differential equations. Math Tables Aids Comput 7:135–152
12. Shih TM (1984) Numerical heat transfer. Hemisphere Publishing Corporation, Washington
13. Strutwolf J (1995) Digitale Simulation elektrochemischer Systeme: Untersuchungen zeitabhängiger Phänomene an rotierenden Scheibenelektroden und Analyse von Cyclovoltammogrammen durch direkte Simulation. Ph.D. thesis, Universität Bielefeld, Bielefeld

14. Feldberg SW (1990) A fast quasi-explicit finite difference method for simulating electrochemical phenomena. Part I. Application to cyclic voltammetric problems. J Electroanal Chem 290:49–65
15. Rudolph M (1991) A fast implicit finite difference algorithm for the digital simulation of electrochemical processes. J Electroanal Chem 314:13–22
16. Smith GD (1985) Numerical solution of partial differential equations, 3rd edn. Oxford University Press, Oxford
17. Hairer E, Wanner G (1991) Solving ordinary differential equations II. Stiff and differential-algebraic problems. Springer, Berlin
18. Dahlquist GG (1963) A special stability problem for linear multistep methods. BIT Numer Math 3:27–43
19. Dahlquist GG (1963) Stability questions for some numerical methods for ordinary differential equations. Proc Symp Appl Math 15:147–158
20. Henrici P (1962) Discrete variable methods in ordinary differential equations. Wiley, New York
21. Hairer E, Wanner G (1983) On the instability of the BDF formulas. SIAM J Numer Anal 20:1206–1209
22. Dahlquist G, Björk Å (1974) Numerical methods. Prentice-Hall, New Jersey
23. von Neumann J, Richtmyer RD (1950) A method for the numerical calculation of hydrodynamic shocks. J Appl Phys 21:232–237
24. O'Brien GG, Hyman MA, Kaplan S (1950) A study of the numerical solution of partial differential equations. J Math Phys 29:223–251
25. Britz D (1998) An error propagation in the numerical literature. BIT 38:217–218
26. Britz D, Strutwolf J (2000) Higher-order spatial discretisations in electrochemical digital simulation. 1. Combination with the BDF algorithm. Comput Chem 24:673–684
27. Strutwolf J, Britz D (2001) Use of high-order discretisations in digital simulation. 2. Combination with the extrapolation algorithm. Comput Chem 25:511–520
28. Bieniasz LK, Østerby O, Britz D (1995) Numerical stability of finite difference algorithms for electrochemical kinetic simulations: matrix stability analysis of the classic explicit, fully implicit and Crank-Nicolson methods and typical problems involving mixed boundary conditions. Comput Chem 19:121–136
29. Bieniasz LK, Østerby O, Britz D (1995) Numerical stability of the Saul'yev finite difference algorithms for electrochemical kinetic simulations: matrix stability analysis for an example problem involving mixed boundary conditions. Comput Chem 19:357–370
30. Bieniasz LK, Østerby O, Britz D (1995) Numerical stability of finite difference algorithms for electrochemical kinetic simulations. Matrix stability analysis of the classic explicit, fully implicit and Crank-Nicolson methods, extended to the 3- and 4-point gradient approximation at the electrodes. Comput Chem 19:351–355
31. Bieniasz LK, Østerby O, Britz D (1997) The effect of the discretization of the mixed boundary conditions on the numerical stability of the Crank-Nicolson algorithm of electrochemical kinetic simulations. Comput Chem 21:391–401
32. Keast P, Mitchell AR (1966) On the instability of the Crank Nicolson formula under derivative boundary conditions. Comput J 9:110–114
33. Johannsen K, Britz D (1999) Matrix stability of the backward differentiation formula in electrochemical digital simulation. Comput Chem 23:33–41
34. Bieniasz LK, Britz D (1993) Electrochemical kinetic simulations of mixed diffusion/homogeneous reaction problems by the Saul'yev finite difference algorithms. Anal Chim Acta 278:59–70
35. Bieniasz LK, Britz D (1993) Efficiency of electrochemical kinetic simulations by orthogonal collocation and finite difference methods. A comparison. Acta Chem Scand 47:757–767
36. Bieniasz LK (1993) The von Neumann stability of finite-difference algorithms for the electrochemical kinetic simulation of diffusion coupled with homogeneous reactions. J Electroanal Chem 345:13–25

37. Britz D, Østerby O (1994) Some numerical investigations of the stability of electrochemical digital simulation, particularly as affected by first-order homogeneous reactions. J Electroanal Chem 368:143–147
38. Gourlay AR, Morris JL (1980) The extrapolation of first order methods for parabolic partial differential equations. II. SIAM J Numer Anal 17:641–655
39. Strutwolf J, Schoeller WW (1997) Digital simulation of potential step experiments using the extrapolation method. Electroanalysis 9:1403–1408
40. Østerby O (1998) The error of the Crank-Nicolson method for linear parabolic equations with a derivative boundary condition. Technical Report PB-534, DAIMI, Aarhus University
41. Britz D (1998) Time shift artifacts and start-up protocols with the BDF method in electrochemical digital simulation. Comput Chem 22:237–243
42. Gavaghan DJ (1997) How accurate is your two-dimensional numerical simulation? Part 1. An introduction. J Electroanal Chem 420:147–158
43. Rabitz H, Kramer M, Dacol D (1983) Sensitivity analysis in chemical kinetics. Ann Rev Phys Chem 34:419–461
44. Rabitz H (1987) Chemical dynamics and kinetics phenomena as revealed by sensitivity analysis techniques. Chem Rev 87:101–112
45. Turányi T (1990) Sensitivity analysis of complex kinetic systems. Tools and applications. J Math Chem 5:203–248
46. Bieniasz LK, Speiser B (1998) Use of sensitivity analysis methods in the modelling of electrochemical transients. Part 1. Gaining more insight into the behaviour of kinetic models. J Electroanal Chem 441:271–285. Erratum: ibid. 452:139 (1998)
47. Bieniasz LK, Dümmling S, Speiser B, Würde M (1998) Use of sensitivity analysis methods in the modelling of electrochemical transients. Part 2. Model expansion and model reduction. J Electroanal Chem 447:173–186
48. Bieniasz LK, Speiser B (1998) Use of sensitivity analysis methods in the modelling of electrochemical transients. Part 3. Statistical error/uncertainty propagation in simulation and in nonlinear least-squares parameter estimation. J Electroanal Chem 458:209–229
49. Rudolph M (2004) Digital simulations on unequally spaced grids. Part 3. Attaining exponential convergence for the discretisation error of the flux as a new strategy in digital simulations of electrochemical experiments. J Electroanal Chem 571:289–307
50. Britz D, Strutwolf J, Thøgersen L (2001) Investigation of some starting protocols for BDF (FIRM) in electrochemical digital simulation. J Electroanal Chem 512:119–123
51. Britz D (2001) Consistency proof of Feldberg's simple BDF start in electrochemical digital simulation. J Electroanal Chem 515:1–7
52. Britz TJ, Britz D (2003) Mathematical proof of the consistency of Feldberg's simple BDF start in electrochemical digital simulations. J Electroanal Chem 546:123–125
53. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
54. Bieniasz LK (2002) Use of the Numerov method to improve the accuracy of the spatial discretisation in finite-difference electrochemical kinetic simulations. Comput Chem 26:633–644
55. Bieniasz LK (2004) Improving the accuracy of the spatial discretisation in finite-difference electrochemical kinetic simulations, by means of the extended Numerov method. J Comput Chem 25:1075–1083
56. Bieniasz LK (2004) A fourth-order accurate, Numerov-type, three-point finite-difference discretization of electrochemical reaction-diffusion equations on nonuniform (exponentially expanding) grids in one-dimensional space geometry. J Comput Chem 25:1515–1521
57. Pearson CE (1965) Impulsive end condition for diffusion equation. Math Comput 19:570–576
58. Rannacher R (1982) Discretisation of the heat equation with singular initial data. Z Angew Math Mech 62:T346–T348
59. Britz D, Poulsen K, Strutwolf J (2004) Reference values of the diffusion-limited current at a disk electrode. Electrochim Acta 50:107–113. See Erratum, ibid. 53:8101 (2008)

60. Britz D, Poulsen K, Strutwolf J (2005) Reference values of the diffusion-limited chronoamperometric current at a microband electrode. Electrochim Acta 51:333–339. See Erratum, ibid. 53:7805 (2008)
61. Britz D, Østerby O, Strutwolf J (2010) Reference values of the chronoamperometric response at cylindrical and capped cylindrical electrodes. Electrochim Acta 55:5629–5635
62. Bieniasz LK (2015) Theory of potential step chronoamperometry at a microband electrode: complete explicit semi-analytical formulae for the Faradaic current density and the Faradaic current. Electrochim Acta 178:25–33
63. Bieniasz LK (2016) Highly accurate, inexpensive procedures for computing theoretical chronoamperometric currents at single straight electrode edges and at single microband electrodes. J Electroanal Chem 760:71–79
64. Nann T, Heinze J (1999) Simulation in electrochemistry using the finite element method. Part 1. The algorithm. Electrochem Commun 1:289–294
65. Nann T, Heinze J (2003) Simulation in electrochemistry using the finite element part 2: scanning electrochemical microscopy. Electrochim Acta 48:3975–3880
66. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Application to a simple model problem. Electrochem Commun 2:150–156
67. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. Theory. Electrochem Commun 2:157–162
68. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. First-order EC' mechanism at inlaid and recessed discs. Electrochem Commun 2:163–170
69. Harriman K, Gavaghan DJ, Houston P, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. An E reaction at a channel microband electrode. Electrochem Commun 2:567–575
70. Harriman K, Gavaghan DJ, Houston P, Kay D, Süli E (2000) Adaptive finite element simulation of currents at microelectrodes to a guaranteed accuracy. $ECE$ and $EC_2E$ mechanisms at channel microband electrodes. Electrochem Commun 2:576–585
71. Harriman K, Gavaghan DJ, Süli E (2004) Simulation of linear sweep voltammetry using an adaptive finite element algorithm. J Electroanal Chem 573:169–174
72. Gavaghan DJ, Gillow K, Süli E (2006) Adaptive finite element methods in electrochemistry. Langmuir 22:10666–10682
73. Bieniasz LK (2000) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 5. A finite-difference adaptive space/time strategy based on a patch-type local uniform grid refinement, for kinetic models in one-dimensional space geometry. J Electroanal Chem 481:115–133. Corrigendum: ibid. 565:131 (2004)
74. Bieniasz LK (2000) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 6. Testing of the finite-difference patch-adaptive strategy on example models with solution difficulties at the electrodes, in one-dimensional space geometry. J Electroanal Chem 481:134–151. Corrigendum: ibid. 565:133 (2004)
75. Bieniasz LK, Bureau C (2000) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 7. Testing of the finite-difference patch-adaptive strategy on example models with moving reaction fronts, in one-dimensional space geometry. J Electroanal Chem 481:152–167. Corrigendum: ibid. 565:135 (2004)
76. Bieniasz LK (2002) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 10. Extension of the patch-adaptive strategy to kinetic models involving spatially localised unknowns at the boundaries, multiple space intervals, and nonlocal boundary conditions, in one-dimensional space geometry. J Electroanal Chem 527:1–10. Corrigendum: ibid. 565:137 (2004)
77. Bieniasz LK (2004) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 14: extension of the patch-adaptive strategy to time-dependent models involving migration-diffusion transport in one-dimensional space geometry,

and its application to example experiments described by Nernst-Planck-Poisson equations. J Electroanal Chem 565:251–271

78. Bieniasz LK (2004) Use of dynamically adaptive grid techniques for the solution of electrochemical kinetic equations. Part 15: patch-adaptive simulation of example transient experiments described by Nernst-Planck-electroneutrality equations in one-dimensional space geometry. J Electroanal Chem 565:273–285
79. Britz D, Østerby O, Strutwolf J (2012) Minimum grid digital simulation of chronoamperometry at a disk electrode. Electrochim Acta 78:365–376
80. Verbrugge MW, Baker DR (1992) Transient diffusion and migration to a disk electrode. J Phys Chem 96:4572–4580
81. Strutwolf J (2005) Computational study of chronoamperometry at rectangular microelectrodes. Electroanalysis 17:1547–1554

# Chapter 16
# Programming

In Appendix E selected downloadable example modules, functions, subroutines and programs are discussed. All have been tested in the form in which they appear. Nevertheless, this does not guarantee that there are no remaining bugs. The word "bug" encompasses the spectrum from "cosmetic, of little consequence", through "potentially serious under certain input conditions" to "fatal". The middle of the spectrum is, of course, the region causing the programmer the greatest trouble.

## 16.1 Language and Style

There are, at the time of writing, two strong contenders for the choice of a compiled language for digital simulation programs: Fortran and C, including C++. The Fortran standard is maintained and extended by an international working group [1] and recent versions are Fortran 90, 95, 2003 and 2008. Despite the unfavourable comments computer scientists reserve for Fortran, that language—certainly since the Fortran 90 standard and especially even later versions—is eminently suitable for numerical analysis. This is partly because of the large volume of existing Fortran scientific subroutines (and intrinsic functions), although most of these are now also available for C and C++. Especially C++ appears to be more powerful but also harder to learn, and the code is not as readily understandable as that for Fortran 90 and later versions. Most of the past criticism of Fortran is now unfounded. Thus, one can have structures (collection of different data types under one name), pointers (enabling linked lists, for example) and recursion, a powerful tool for some applications. Another very useful feature since Fortran 90 is whole-array operations and array sections, which make programs more compact (eliminating many loops) and thus more easily readable. Dynamic memory allocation is supported since Fortran 90. One might say that Fortran is more for the occasional programmer, while C++ is more suitable for writing general programs such as packages to solve

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_16

a variety of simulation problems (see Chap. 17). There are free Fortran compilers available such as gfortran [2], G95 [3] and (at least in the past, free for Linux operating systems), Intel Fortran 95 [4]. The Fortran 95/2003/2008 compiler of the GNU project, gfortran, originally developed for Linux/Unix has been ported to MS Windows and MacOS. Present day Fortran includes object-oriented programming support such as type extension and inheritance, polymorphism, dynamic type allocation, and type-bound procedures, providing complete support for abstract data types. The latest Intel compiler is described [4] as having "Full language Fortran 95, full Fortran 2003, plus significant Fortran 2008 features". The present authors have been using gfortran (JS) and Intel Fortran 95 (DB) for some time.

Other languages can of course be used, such as Pascal or Basic or even Java, but the former two are rarities in digital simulation these days. There is an extensive text book on electrochemical digital simulation using Mathematica [5] by Honeychurch [6]; unfortunately, the book is difficult to access. Batchelor et al. used Python and Julia [7], both script languages. MATLAB [8] has seen some use, in conjunction with COMSOL Multiphysics® (formerly called FEMLAB®) [9–14]. Even Excel has been used for simulation [15, 16].

Having decided on Fortran, one strong recommendation is the use only of standard Fortran as far as practicable. Then, programs will be transportable. The example programs in the Appendices should all run on any computer with a standard Fortran 90/95/2003/2008 compiler. A useful text, one of many on Fortran 95/2003, is that by Metcalf et al. [17].

## 16.2 Debugging

It is well known that more time is spent correcting programs than actually planning or typing them in. With very large system programs, it is assumed [18] that some errors remain, and this is probably true for large simulation packages. However, the relatively short programs written for single applications by an electrochemist should be free of errors, and these are mostly rather easy to find.

Syntax errors are flagged by the compiler and are quickly eliminated. When a program is syntactically correct and compiled (and linked) successfully, it may still contain errors and these are harder to find. The use of `implicit none` is strongly recommended, as it allows the detection of typing errors in variable names. Clearly structured, modular programming also helps avoid errors or helps to localise them when they occur.

All variables must be given a value before being used. Some implementations start a program with all variables set to zero. It is tempting but dangerous to use this, as other installations leave previous values in all memory locations. Also it is good practice to set the compiler to checking array bounds. Some compilers allow a program to go beyond array bounds, and this can cause strange errors that are otherwise hard to trace. Checking slows down the program execution, but the check can be switched off when the program is finally considered correct. This can be called error prevention. Programming style can also help. There should be comments explaining the less obvious lines, but not too many, or they will distract. The use of array sections, whenever arrays are addressed or passed to subroutines, is strongly recommended. For example, copying one array into another can be done simply by the statement

```
D = C
```

if they have the same declared dimensions, but it is better to explicitly indicate the index ranges with

```
D(1:N) = C(1:N)
```

This may be a little tedious but is good practice.

Finding program bugs is an art but certain techniques will help. The simplest method, after unsuccessfully reading through the problem program, is to explain the program to someone else. Often, one sees the error while doing this. This is called "egoless programming" [19]. Then there are diagnostic tools. Most Fortran implementations have a debugging facility which, when enabled, allows running a program with stops at strategic places, at which one can display and even change some variable values. These tools can be a little unwieldy. A simpler method is to insert `print` statements at suspect places, narrowing them down until the error has been cornered. These days, extensive output is no problem, since we work at screen terminals and thus do not have to handle large volumes of paper. One very useful method is to display concentration values. One need not print them all, a select number usually being enough. For example, concentrations at all the edges of a domain are of interest (and are "the usual suspects"). Often, one sees obviously incorrect values, which then point to the part of the program containing the error.

A difficult situation is a new simulation, with unknown results. How can we be sure that the results are correct? Often, the simulated system has special cases with known results; these should of course be checked. If we are developing a new simulation method, it can be checked against others known to work. For critical work, it may be necessary to write several different programs—perhaps written by different people—and to make sure that all converge to the same results as simulation intervals approach zero. The present authors have for some years programmed both using finite difference programs and the commercial package COMSOL Multiphysics®, demanding that the two should produce essentially the same results. This can detect errors, if the two differ by more than the expected round-off differences.

A new program should be treated with suspicion, as if it were certain to contain bugs, even (or especially) if the results look "good". It is often possible to reduce the input parameters such that the results are known.

## 16.3 Libraries

One finds that a number of subroutines are used repeatedly in different programs and these are best placed into a library, possibly already compiled (object library), or in the form of a text library. There is a number of such routines discussed in Appendix E. The code `stuff.f90` is a module that the present authors use with every program by naming it in the compilation. As is seen, it defines some precision types, gives $\pi$ to two different precision levels, etc. This makes the types in all programs and subroutines compatible with each other. The functions and subroutines are all in a large precompiled object library. These routines are all thoroughly tested.

## References

1. http://www.nag.co.uk/sc22wg5/
2. https://gcc.gnu.org/fortran
3. http://www.g95.org
4. https://software.intel.com/en-us/fortran-compilers
5. https://www.wolfram.com/mathematica/
6. Honeychurch MJ (2004–2006) Simulating electrochemical reactions with Mathematica. IBNH, St. Lucia, QLD
7. Batchelor-McAuley C, Yang M, Hall EM, Compton RG (2015) Correction factors for the analysis of voltammetric peak currents measured using staircase voltammetry. J Electroanal Chem 758:1–6
8. http://www.mathworks.com/products/matlab/
9. Amatore C, Sella C, Thouin L (2006) Electrochemical time-of-flight responses at double-band generator-collector devices under pulsed conditions. J Electroanal Chem 593:194–202
10. Leonhardt K, Avdic A, Lugstein A, Pobelov I, Wandlowski T, Wu M, Gollas B, Denuault G (2011) Atomic force microscopy-scanning electrochemical microscopy: influence of tip geometry and insulation defects on diffusion controlled currents at conical electrodes. Anal Chem 83:2971–2977
11. Leonhardt K, Avdic A, Lugstein A, Pobelov I, Wandlowski T, Gollas B, Denuault G (2013) Scanning electrochemical microscopy: diffusion controlled approach curves for conical AFM-SECM tips. Electrochem Commun 27:29–33
12. Liljeroth P, Johans C, Slevin CJ, Quinn BM, Kontturi K (2002) Disk-generation/ring-collection scanning electrochemical microscopy: theory and application. Anal Chem 74:1972–1978
13. Liljeroth P, Johans C, Slevin CJ, Quinn BM, Kontturi K (2002) Micro ring-disk electrode probes for scanning electrochemical microscopy. Electrochem Commun 4:67–71
14. Snowden ME, Edwards MA, Rudd NC, Macpherson JV, Unwin PR (2013) Intrinsic electrochemical activity of single walled carbon nanotube-Nafion assemblies. Phys Chem Chem Phys 15:5030–5038
15. Morf WE, Koudelka-Hep M, de Rooij NF (2006) Theoretical treatment and computer simulation of microelectrode arrays. J Electroanal Chem 590:47–56
16. Morf WE, Pretsch E, De Rooij NF (2007) Computer simulation of ion-selective membrane electrodes and related systems by finite-difference procedures. J Electroanal Chem 602:43–54

17. Metcalf M, Reid J, Cohen M (2004) Fortran 95/2003 explained. Oxford University Press, Oxford
18. Gilb T (1976) Software metrics. Studentlitteratur, Lund
19. Weinberg GM (1971) The psychology of computer programming. Van Nostrand Reinhold, New York

# Chapter 17
# Simulation Packages

Not every electrochemist wishes to write his or her own simulation programs, and there are a number of ready-made programs that can be obtained through the Internet or otherwise, some commercial, some free, and some that are online programs. These can be convenient but all have some limitations of various kinds. There have been several reviews describing these packages [1–5].

The following contains a number of Internet addresses. These are all active at the time of writing (January 2016) but some of them may not remain so indefinitely.

If using the MOL approach (Chap. 9), one needs to solve a set of *ode*s, if the boundary conditions can be incorporated into the differential equations. This will be the case with some simple boundary conditions such as those for the potential jump experiment. There are a number of freely available *ode* solvers, among them `VODE` [6], `LSODE` [7] and `PSODE` [8]. Some of these have been compared for efficiency [9, 10], where also the sparse solvers `MA28` [11, 12] and `Y12M` [13] were compared. Many of these routines can be downloaded from the `netlib` site [14]. If the boundary conditions (as is usual) are discretised in the form of algebraic equations, then they form, together with the system of *ode*s, a system of DAEs, as also mentioned in Chap. 9. For these, there is the program `DASSL`, described first by Petzold in 1983 [15], and again in the text of Brenan et al. [16], and can be downloaded from [17]. It uses BDF to solve the system.

There are some non-electrochemical *pde* solvers that some find convenient, such as `PHREEQC` [18] or the elliptic equation solver (useful for potential field or steady state computations) `PLTMG` [19], using the multigrid algorithm [20–23] for increased speed.

The Spanish Horno group (see Chap. 9, Sect. 9.9 for many references to this) casts the electrochemistry into an electrical model and uses `SPICE` and later `PSPICE` [24] to solve the resulting network model. The same method is used by a Chinese group [25–28], who in fact have written a general purpose electrochemical simulator around this technique (see below). Possibly the earliest use of a general simulator was that by Klinger et al. [29], who used the simulator `S/360 CSMP` of

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8_17

IBM, written for the IBM/360 machine. CSMP was meant to simulate control processes. The authors used it to simulate CV of an adsorbed species. ESTYM_PDE is a program based on orthogonal collocation (see Sect. 9.6) on moving finite elements [30]. ESTYM_PDE was applied to simulate CVs complicated by homogeneous reactions [31–36], metal deposition and dissolution processes[37–39], hydrogen evolution and dissolution [40] and amalgam formation [41].

There are general purpose commercial simulation packages using FEM. Some workers in Switzerland [42–48] preferred Flux Expert. Recent years have seen increased use of COMSOL Multiphysics® [49] for simulating electrochemical systems; some references have been mentioned in Chap. 12. A short review has been published [50] and more references can be found there. Refinement strategies of predefined meshes (in contrast to adaptive meshing) for a few electrochemical examples have been discussed [51]. ANSYS® [52], another commercial general purpose FEM package, has its strength in computational fluid dynamics (CFD) and has mainly been applied to simulate flow problems in electrochemistry, such as in fuel cells [53, 54] and turbulent flow at electrodes [55], to name just a few examples. Both program packages provide modules ("add-on products") which facilitate the input (*pde*s, boundary conditions) for certain problems. For example for COMSOL's version 5.1 (the current version at the time of writing) modules for batteries and fuel cells, electrodeposition, corrosion and electrochemistry are available, to mention just modules related to electrochemistry. An important feature of both FEM packages is the ability to perform multiphysics simulations, i.e. the combination of multiple physical phenomena in one model. One example is the design and optimization of thermo-electrochemical cells [56], where equations for mass and heat transfer, fluid dynamics and electrokinetics are solved by COMSOL Multiphysics®.

Of greatest interest here is the group of programs that to a lesser or greater degree solve a variety of electrochemical simulation problems. There are quite a number of these, a few of them somewhat prominent and commercial, and some more or less private but accessible to others.

ELSIM [57–59] is freely available from the author [60]. It has been updated since its earliest version around 1992, but is still DOS-based, that is, there is no Windows version as yet. A transfer to a current operating system would be welcomed. ELSIM accepts input in the form of reaction equations, in which case the program itself generates the governing equations; or the user can enter the governing equations directly. ELSIM is not limited to a discrete number of mechanisms or experiments, these being determined by what the user enters. Even the method used for simulation can be chosen (within some limits, indicated by the program when necessary). The reaction-diffusion equations (also convection) are solved by finite difference methods. Boundary conditions are implemented in the form of a set of differential-algebraic equations (DAEs, see Chap. 9, e.g. page 198), allowing a very general formulation of the dynamics for the interfacial species. Indeed, it is possible to simulate an interfacial process in the absence of dynamical transport of a bulk species. Besides finite-difference simulation, ELSIM allows the solution of user defined integral equations related to electrochemical systems [61]. This has been used to solve the nonlinear integral equation describing a CV experiment at partially blocked electrodes [62]. ELSIM is written mostly in C using a C++ compiler and the source code is available and can be modified by the user.

DigiSim [63], sold by Bioanalytical Systems, Inc. [64], is also a general simulator, although it only offers (one space dimensional) CV simulations. It is general in the sense that any mechanism can be entered by the user and the program does the rest. It can be "tricked" into a few other simulation cases such as a potential step (by making the sweep very fast and applying a long holding time after the sweep) or even an electrochemical luminescence experiment by setting a huge diffusion coefficient for a fictitious species that in fact is the light source [65]. However, CV is DigiSim's forte. The program is Windows-based and easy to use, but can be led into accepting nonsensical input, as reported in a review [66]. This has possibly since been rectified. The program is written in C++.

One of the authors of DigiSim has produced his own separate program, DigiElch [67] which is now distributed by Gamry Instruments [68]. Besides one-dimensional diffusion DigiElch is also capable of two-dimensional diffusion simulations for UMBEs and UMDEs. CV, chronoamperometry, square wave voltammetry and electrochemical impedance spectroscopy can be simulated.

Another well-known simulator is attached to the textbook by Gosser [69] on a diskette. It has also been reviewed [70, 71].

The program ELECTROCHEMIST.com, formerly called Polar and Polarograph, has been vigorously promoted by its author. The commercial program can simulate voltammetry and chronoamperometry. A description of the program can be found on a webpage [72]. We are not aware of the algorithms used in this program.

Speiser et al. began in 1989 to write general simulation code EASI (for ElectroAnalytical SImulation), extended to EASIEST, as parameter estimation was added. These are described in a series of papers [73–76], and the code, which was in C++ and meant to run under the Unix operating system, was available from the author. The project has now been terminated, in favour of a new, more general one, EChem++ [77–85]. The project is established as a long-term project based on an object-oriented design written in the C++ programming language. In the long run, the task of EChem++ is not only to serve as a framework for simulation, but also to control electrochemical experiments in real time as well as data analysis. The aim is to create a very general "problem solving environment" for electrochemistry. The object-oriented design allows a breakdown of complex structures into small reusable parts, which, among other advantages, facilitates the step-by-step extension of the project. For example, one of the the first steps in the EChem++ project was the development of a C++ class collection of electrochemical excitation functions [78], e.g. potential-time curves for LSV and CV or of almost any other shape. The excitation functions can be used to drive an experiment or as part of dynamical boundary conditions in numerical simulations. The simulation part of EChem++ is based on adaptive FEM for spatial integration and adaptive time integration using Rosenbrock schemes [81]. Echem++ is an open source project under the Gnu Public Licence (GPL) [86]. The software runs (currently) under Linux and the source code and further information are available from sourceforge [87].

There is at present one online simulator to the authors' knowledge, that of Ohta [88], which can be used by anyone for a set of simulation situations.

Apart from the above, there is a plethora of more or less publically accessible electrochemical simulators. Carlo Nervi continues with his ESP package [89]. CONDESIM [90] is a commercial software package for simulating several experiment types. There is also CONDECON [90], a data treatment package for convolution/deconvolution analysis of voltammograms. Other packages are TRANSIENT [91], SIMULA, described in [92] as part of the "Seraphim project" but without any details being given. A Chinese group developed a package called EEGNA (exponentially expanded grid network approach) [25–28]. The local ECL-PACKAGE (not accessible to others) is used by Svir and coworkers [93] for simulations of electrochemical luminescence experiments. Similar local general programs are presented by Penar et al. [94] for rotating electrode simulations and finally, there is an OC package for LSV by Villa et al. [95]. Both the two last programs are available from the authors of these articles.

## 17.1 Kinetic Compilers

The user has a certain model in mind which he or she wants to simulate using one of the ready-made program packages. The model may consist of a set of parameters and variables. First, the geometry itself and the transport mode (diffusion, diffusion-migration, convection) have to be specified. Afterward the program has to be informed about several parameters and variables concerning the kinetics of the model, like number of species, initial concentrations, diffusion coefficients, heterogeneous reactions with their rate constants, formal potentials and transfer coefficients, homogeneous reactions with their rate constants, equilibrium constants, adsorption isotherms, adsorption rates, etc., and combinations of all of these.

A simulation program intended for others than just its programmer needs to provide a more or less convenient interface for the complex input by the user. Usually a symbolic input (using ASCII characters) is required. A simple example, often encountered in electrochemistry would be the reaction $A^+ + e^- \rightleftharpoons B$ together with values of $E_0, k^0, \alpha$ (formal potential, heterogeneous rate constant, transfer coefficient) for an electrode reaction according to the Butler–Volmer formalism. The input has to be translated into governing equations which can be handled by the implemented solver routines of the program. Here the *kinetic compiler* (synonymous expressions are *reaction compiler* or *chemical compiler*) comes into play. Generally speaking, a kinetic compiler automatically translates the user input describing the kinetic part of a model into mathematical equations which can be handled by the solver. A kinetic compiler might also check for consistency of the input. Such compilers are part of ELSIM [96], DigiSim [97], DigiElch and EChem++ [80] (see also errata [79]). It should be mentioned that outside the electrochemical context, reaction compilers and translators have a longer history, see [96] for a comprehensive list. However, these are mostly designed for simulations independent of space, involving only *ode*s.

The kinetic compiler of DigiSim [97] allows for an intuitive input of heterogeneous (single) electron transfer reactions and first and second-order homogeneous reactions. *Thermodynamically superfluous reactions* (TSRs) [97] are detected and the user will be informed about parameters which are automatically adjusted for consistency. A TSR is a reaction whose equilibrium constant and formal potential is determined by previously specified values for equilibrium constants and formal potentials of one or more other reactions. DigiSim is designed for the simulation of potential sweep experiments and the kinetic compiler has some limitations, as mentioned in [80, 96, 97]. DigSim is a commercial program and the kinetic compiler cannot be modified or extended, nor has the user access to the internal problem presentation, though it can be "tricked" into some work-arounds, see above.

DigiElch's kinetic compiler is an extension of the compiler used by DigiSim in order to address the wider spectra of experiments possible to simulate with DigiElch.

The kinetic compiler used with the ELSIM program has been described in detail [96] and the integration into ELSIM has been outlined [59]. ELSIM's kinetic compiler translates the reaction mechanism given by the user into a target text of the governing equations and problem description files are created which can be checked and modified by the user. Thus, ELSIM's compiler allows for more flexibility regarding the simulation of more complex kinetics, though the input is more complex compared to DigiSim or DigiElch. A check of the presence of TSRs and violation of mass balance is performed. In a second stage, a formula translator generates the equations for the internal solver.

Ecco, Electro Chemical COmpiler [80], see also Errata [98], is the kinetic compiler component of the EChem++ project [80] and enables the modelling of homogeneous and surface reactions and heterogeneous electron transfers as well as adsorption processes. Following the EChem++ philosophy, Ecco is separated from the subsequent simulation procedure and can therefore be combined with any numerical method. This ensures a high reuseability of the kinetic compiler. As part of the EChem++ project, the source code of Ecco is available [87] under the GPL [86].

## 17.2 Parameter Estimation

One motivation of performing simulations is the interpretation of experimental data, e.g. voltammograms or chronoamperometric data, by estimation of physical parameters. This can be achieved by fitting simulated curves to experimental ones in a *nonlinear regression analysis* process. The user provides a model to the simulation program and an *objective function* is then minimized by systematic variation of the model parameter. The best fit is achieved when a global minimum of the object function has been found. This function is usually the sum of the squared residuals (*least squares*)

$$\chi(\mathbf{a})^2 = \sum_{i=1}^{N} w_i\,(y_i - f(\mathbf{a}, x_i))^2 \tag{17.1}$$

where $N$ is the total number of data points, $w_i$ is their individual weight, the vector **a** contains $M$ model parameter $(a_1, a_2, \ldots a_M)$, $x_i$ is the independent variable, $y_i$ is the $i$'th (experimental) data point and $f(\mathbf{a}, x_i)$ is the model function, i.e. the value of the simulated curve for a parameter set **a** at $x_i$. For example, in analysing (fitting) of an experimental CV or LSV, a model of a species undergoing an electrode reaction according to the Butler–Volmer formalism is assumed. Then $y_i$ is the current, $x_i$ is the potential and the parameter vector **a** may contain $k^0$, $\alpha$, $E^{0'}, \ldots$. The function $f(\mathbf{a}, x_i)$ is the simulated value of the current for a certain parameter set **a** and a voltage $x_i$. DigiSim, DigiElch, ELSIM and the program of Gosser [69] from his out of print book provide an option of least squares fitting of experimental data. The $\chi^2$ function spans an $M$ dimensional hypersurface, where $M$ is the number of model parameters to estimate. Search algorithms are used to locate a minimum by varying the model parameter set. There are two principal ways. A direct search does not make use of the gradients of the hypersurface but varies the parameters according to certain heuristic rules. The *downhill simplex* method by Nelder and Mead [99] is a direct search algorithm and is implemented in the program CVSIM by Gosser [69], and in a modified version [100] in ELSIM [59]. Other methods like Newton–Raphson or Newton iteration (see [101] or textbooks on numerical mathematics, e.g. [102, 103]) and Levenberg–Marquardt [101, 104, 105] use the information obtained from partial derivatives of the $\chi^2$ function with respect to the parameters for the search of a minimum of the object function. The Newton–Raphson method is used by DigiSim [63, 106] and DigiElch [67]. The simulation program ELECTROCHEMIST.com [72] provides data fitting of experimental curves (we have no information about the method). A comparison of the downhill simplex and Levenberg–Marquardt method for fitting voltammetric data has been published and the authors favour the Levenberg–Marquardt algorithm as a good compromise between robustness and convergence rate [107]. Not relying on partial derivatives during a search contributes to the robustness and stability of the downhill simplex method. However, in our experience [108] it also decreases the efficiency, i.e. the speed of the search. By direct comparison of both methods to fit an experimental CV, using $\mathbf{a} = [k^0, \alpha, E^{0'}, R_u, C_d]^T$ ($R_u$ and $C_d$ are uncompensated resistance and capacity of the double layer, respectively) as the vector of the model parameters, a modified version of the Levenberg–Marquardt algorithm presented in the Numerical Recipes book [101] was more efficient than a downhill simplex search [108].

By its nature, the downhill simplex lacks statistical information about the estimated parameter. This information is obtained from the shape or gradients of the $\chi^2$ hypersurface at the minimum. A suitable way to overcome this situation is to fit a quadratic function to the vertices of the final simplex, as first suggested by Nelder and Mead [99] and later investigated by Phillips and Eyring [109] and Brumby [110, 111]. This final step adds quadratic convergence to the downhill simplex method and enables the calculation of statistical information about the quality of the fit. In an electrochemical context this has been done [108] using the strategy of Brumby [110, 111]. Statistical information is necessary to judge the quality of a fit. A good agreement between a simulated and an experimental voltammogram, that is a minimum value of the $\chi^2$ function is a criterion which cannot be trusted alone. Statistical information like parameter uncertainties, confidence intervals, confidence regions, covariance matrices and residual plots are necessary to judge the validity of the model and to examine parameter coupling. Coupled parameters are parameters that are not independent in the sense that combinations of different values of the parameters lead to identical or almost identical values of the $\chi^2$ function. A typical example for coupled parameters in cyclic voltammetry is the heterogeneous rate constant $k^0$ and uncompensated resistance $R_u$, as has been shown by statistical analysis [108].

Finally, during the procedure of minimizing the object function (17.1), simulations are repeated perhaps a thousand times with systematic variation of the model parameters. Therefore the importance of the efficiency of the simulation method should not be underestimated.

## References

1. Speiser B (1996) Numerical simulation of electroanalytical experiments: recent advances in methodology. In: Bard AJ, Rubinstein I (eds) Electroanalytical chemistry, vol 19. Marcel Dekker, New York, pp 1–108
2. Bieniasz LK (2002) Towards computational electrochemistry - a kineticist's perspective. In: Conway BE, White RE (eds) Modern aspects of electrochemistry, vol 35. Kluwer/Plenum, New York, pp 135–195
3. Bieniasz LK, Britz D (2004) Recent developments in digital simulation of electroanalytical experiments. Pol J Chem 78:1195–1219
4. Speiser B (2014) Numerical simulations in electrochemistry. In: Kreysa G, Ota K-i, Savinell RF (eds) Encyclopedia of applied electrochemistry. Springer, Heidelberg, pp 1380–1385
5. Speiser B (2016) Application of digital simulation. In: Hammerich O, Speiser B (eds) Organic electrochemistry, 5th edn. CRC Press, Boca Raton, pp 205–227
6. http://www.netlib.org/ode/vode.f
7. http://www.cs.berkeley.edu/~kdatta/lsode/lsode.html
8. van den Houwen PJ, Sommeijer BP (1993) Analysis of parallel diagonally implicit iteration of Runge-Kutta methods. Appl Numer Math 11:169–188
9. http://db.cwi.nl/rapporten/abstract.php?abstractnr=892
10. de Swart JJB, Blom JG (1995) Experiences with sparse matrix solvers in parallel ODE software. Technical Report Rept. NM-R9520 1995, Centrum voor Wiskunde en Informatica (CWI), Amsterdam. Accessible at http://db.cwi.nl/rapporten/abstract.php?abstractnr=892
11. http://hsl.rl.ac.uk/archive/hslarchive.html
12. Duff IS, Reid JK (1979) Some design features of a sparse matrix code. ACM Trans Math Softw 5:18–35

13. Zlatev Z, Wasniewski J, Schaumburg K (1981) Y12M. Solution of large and sparse systems of linear algebraic equations. Documentation and subroutines. In: Goos G, Hartmanis J (eds) Lecture notes in computer science, vol 121. Springer, Berlin
14. http://www.netlib.org
15. Petzold L (1983) A description of DASSL - a differential/algebraic system solver. In: Stepleman RS, Carver M, Peskin R, Ames WF, Vichnevetsky R (eds) Scientific computing, volume 1, IMACS transactions on scientific computation, 10th IMACS world congress on systems simulation and scientific computation, Montreal, August 1982. North Holland Publishing Company, Amsterdam, pp 65–68
16. Brenan KE, Campbell SL, Petzold LR (1996) Numerical solution of initial-value problems in differential-algebraic equations. SIAM, Philadelphia
17. http://www.engineering.ucsb.edu/~cse/software.html
18. http://wwwbrr.cr.usgs.gov/projects/GWC_coupled/phreeqc/
19. http://www.mgnet.org/mgnet-codes-pltmg.html
20. Alden JA, Compton RG (1996) The multigrid method, MGD1: an efficient and stable approach to electrochemical modelling. The simulaton of double electrode problems. J Electroanal Chem 415:1–12
21. Altas I, Dym J, Gupta MM, Manohar RP (1998) Multigrid solution of automatically generated high order discretizations for the biharmonic equation. SIAM J Sci Comput 19:1575–1585
22. McCormick S (1989) Breaking the matrix speed limit. Nature 337:205
23. Steffen B, Rousar I (1995) Numerical methods for electrochemical modelling - I. Electrochim Acta 40:379–386
24. http://www.orcad.com
25. Deng Z, Lin X (1999) Simulation of ultra-fast cyclic voltammetric curve by the exponentially expand network method. Chin J Anal Chem 27:1376–1380 (in Chinese, Engl. abstract)
26. Deng ZX, Tong ZH, Lin XQ (2004) Global evaluation of linear sweep voltammetric responses with electroactive species confined at the electrode surface. J Electroanal Chem 568:235–245
27. Deng ZX, Lin XQ, Tong ZH (2003) Exponentially expanded grid network approach EEGNA. An efficient way for the simulation of stiff electrochemical problems. Chin J Chem 21:1137–1145
28. Deng ZX, X-Q-Lin, Tong ZH (2004) Universal electrochemical/chemical simulator based on an exponentially expanding grid network approach. Chin J Chem 22:719–726
29. Klinger J, Conway BE, Angerstein-Kozlowska H (1978) Procedures for computer simulation of kinetics of electrochemical surface processes. Comput Chem 2:117–129
30. Kaczmarski K, Mazzotti M, Stortic G, Morbidelli M (1997) Modeling fixed-bed adsorption columns through orthogonal collocations on moving finite elements. Comput Chem Eng 21:641–660
31. Sanecki P, Kaczmarski K (1999) The voltammetric reduction of some benzenesulfonyl fluorides, simulation of the ECE mechanism and determination of the potential variation of the transfer coefficient by using the compounds with two reducible groups. J Electroanal Chem 471:14–25
32. Sanecki P (2001) A numerical modelling of voltammetric reduction of substituted iodobenzenes reaction series. A relationship between reductions in the consecutive-mode multistep system and a multicomponent system. Determination of the potential variation of the elementary charge transfer coefficient. Comput Chem 25:521–539
33. Sanecki P, Skitał P (2002) The cyclic voltammetry simulation of a competition between stepwise and concerted dissociative electron transfer, the modeling of alpha apparent variability, the relationship between apparent and elementary kinetic parameters. Comput Chem 26:297–311
34. Sanecki P, Skitał P (2002) A comparison of the multistep consecutive reduction mode with the multicomponent system reduction mode in cyclic voltammetry. Comput Chem 26:333–340
35. Sanecki P, Amatore C, Skitał P (2003) The problem of the accuracy of electrochemical kinetic parameter determination for the ECE reaction mechanism. J Electroanal Chem 546:109–121

36. Sanecki P, Skitał P, Kaczmarski K (2006) Numerical modelling of ECE-ECE and parallel EE-EE mechanisms in cyclic voltammetry. Reduction of 1,4-benzenedisulfonyl difluoride and 1,4-naphthalenedisulfonyl difluoride. Electroanalysis 18:981–991
37. Sanecki PT, Skitał PM, Kaczmarski K (2010) The mathematical models of the stripping voltammetry metal deposition/dissolution process. Electrochim Acta 55:1598–1604
38. Skitał PM, Sanecki PT (2012) The experimental verification of mathematical two plate model describing the metal deposition/dissolution process. Russ J Electrochem 48:797–803
39. Skitał PM (2014) The mathematical modelling of the palladium deposition/dissolution process by cyclic voltammetry method. Int J Electrochem Sci 8:2589–2602
40. Skitał PM, Sanecki PT, Kaczmarski K (2010) The mathematical model of the stripping voltammetry hydrogen evolution/dissolution process on Pd layer. Electrochim Acta 55:5604–5609
41. Sanecki PT, Skitał PM, Kaczmarski K (2006) An integrated two phases approach to $Zn^{2+}$ ions electroreduction on Hg. Electroanalysis 18:595–604
42. Bianchi F, Ferrigno R, Girault HH (2000) Finite element simulation of an electroosmotic-driven flow division at a T-junction of microscale dimensions. Anal Chem 72:1987–1993
43. Ferrigno R, Brevet PF, Girault HH (1997) Finite element simulation of the chronoamperometric response of recessed and protruding microdisc electrodes. Electrochim Acta 42:1895–1903
44. Ferrigno R, Brevet PF, Girault HH (1997) Finite element simulation of the amperometric response of recessed and protruding microband electrodes in flow channels. J Electroanal Chem 430:235–242
45. Ferrigno R, Josserand J, Brevet PF, Girault HH (1998) Coplanar interdigitated band electrodes for electrosynthesis. Part 5: finite element simulation of paired reactions. Electrochim Acta 44:587–595
46. Ferrigno R, Girault HH (2000) Finite element simulation of electrochemical ac diffusional impedance. Application to recessed microdiscs. J Electroanal Chem 492:1–6
47. Josserand J, Morandini J, Lee HJ, Ferrigno R, Girault HH (1999) Finite element simulation of ion transfer reactions at a single micro-liquid|liquid interface supported on a thin polymer film. J Electroanal Chem 468:42–52
48. Salaun P, Josserand J, Morandini J, Girault HH, Buffle J (2004) Numerical simulations of linear scan anodic stripping voltammetry at a modified square array of hemispherical microelectrodes located in a thin-layer cell. J Electroanal Chem 566:147–158
49. http://www.comsol.com
50. Dickinson EJF, Ekström H, Fontes E (2014) COMSOL Multiphysics®: finite element software for electrochemical analysis. A mini-review. Electrochem Commun 40:71–74
51. Carneiro-Neto EB, Sikora MS, Pereira EC, Lopes MC (2014) Probing the numerical convergence of a commercial finite element software in electrochemical simulations. Electrochemistry (Japan) 82:966–973
52. http://www.ansys.com
53. Iranzo A, Muñoz M, Rosa F, Pino J (2010) Numerical model for the performance prediction of a PEM fuel cell. Model results and experimental validation. Int J Hydrogen Energy 35:11533–11550
54. Oh K, Chippar P, Ju H (2014) Numerical study of thermal stresses in hightemperature proton exchange membrane fuel cell (HT-PEMFC). Int J Hydrogen Energy 39:2785–2794
55. Atempa-Rosiles P, Dáz-Cruz M, Cervantes-Tobón A, González-Velázquez JL, Godínez-Salcedo JG, Rodríguez-Arias YA, Macías-Salinas R (2014) Simulation of turbulent flow of a rotating cylinder electrode and evaluation of its effect on the surface of steel API 5L X-56 during the rate of corrosion in brine added with kerosene and $H_2S$. Int J Electrochem Sci 9:4805–4815
56. Salazar PF, Kumar S, Cola BA (2014) Design and optimization of thermo-electrochemical cells. J Appl Electrochem 44:325–336

57. Bieniasz LK (1992) ELSIM - a user-friendly PC program for electrochemical kinetic simulations. Version 1.0 - solution of integral equations for linear scan and cyclic voltammetry. Comput Chem 16:11–14
58. Bieniasz LK (1993) ELSIM - a PC program for electrochemical kinetic simulations. Version 2.0 - solution of the sets of kinetic partial differential equations in one-dimensional geometry, using finite difference and orthogonal collocation methods. Comput Chem 17:355–368
59. Bieniasz LK (1997) ELSIM - a problem-solving environment for electrochemical kinetic simulations. Version 3.0 - solution of governing equations associated with interfacial species, independent of spatial coordinates or in one-dimensional space geometry. Comput Chem 21:1–12
60. http://www.cyf-kr.edu.pl/~nbbienia/elsim3ad.html
61. Bieniasz LK (2015) Modelling electroanalytical experiments by the integral equation approach. Springer, Heidelberg
62. Strutwolf J, O'Sullivan CK (2007) Microstructures by selective desorption of self-assembled monolayer from polycrystalline gold electrodes. Electroanalysis 19:1467–1475
63. Rudolph M, Reddy DP, Feldberg SW (1994) A simulator for cyclic voltammetry responses. Anal Chem 66:589A–600A
64. http://www.bioanalytical.com
65. Ketter JK, Forry SP, Wightman RM, Feldberg SW (2004) Use of DigiSim to model cyclic voltammetric and photonic responses in electrogenerated chemiluminescent systems. Electrochem Solid-State Lett 7:E18–E22
66. Britz D (1995) [untitled]. Anal Chem 67:600A–601A. Review of DigiSim
67. http://www.elchsoft.com
68. http://www.gamry.com/products/digielch-electrochemical-simulation-software
69. Gosser DK Jr (1993) Cyclic voltammetry. VCH, New York, Weinheim
70. Britz D (1994) [untitled]. Anal Chem 66:792A–793A. Review of David K. Gosser, "Cyclic Voltammetry: Simulation and Analysis of Reaction Mechanisms"
71. Speiser B (1994) Review: cyclic voltammetry. Simulation and analysis of reaction mechanisms by D.K. Gosser. J Electroanal Chem 374:280–282
72. http://electrochemistry.net/science/chemistry/electrochemistry/software.htm
73. Speiser B (1989) EASIEST - Ein Programmsystem zur Simulation von und Parameterbestimmung aus elektroanalytischen Experimenten. In: Gauglitz G (ed) Softwareentwicklung in der Chemie 3. Springer, Berlin, pp 321–332
74. Speiser B (1990) EASIEST - a program system for electroanalytical simulation and parameter estimation. I. Simulation of cyclic voltammetric and chronoamperometric experiments. Comput Chem 14:127–140
75. Speiser B (1991) Electroanalytical simulations. Orthogonal collocation simulation of fast second-order chemical reactions coupled to an electron transfer with a heterogeneous equivalent formulation. Anal Chim Acta 243:301–310
76. Speiser B (1991) Chemical information from electroanalytical data. Part 2. Determination of the rate and equilibrium constants of a chemical reaction preceding a reversible electron transfer from cyclic voltammetric data. J Electroanal Chem 301:15–35
77. http://www.echem.uni-tuebingen.de/echem/software/EChem++/
78. Ludwig K, Rajendran L, Speiser B (2004) ECHEM++ – an object oriented problem solving environment for electrochemistry. Part 1. A C++ class collection for electrochemical excitation functions. J Electroanal Chem 568:203–214. See Erratum, ibid. 571:119 (2004)
79. Ludwig K, Rajendran L, Speiser B (2004) Erratum to "Echem++ - an object oriented problem solving environment for electrochemical excitation functions", [J. Electroanal. Chem. 568 (2004) 203–214]. J Electroanal Chem 571:119
80. Ludwig K, Speiser B (2004) EChem++ - an object-oriented problem solving environment for electrochemistry. 2. The kinetic facilities of ECCO - a compiler for (electro-)chemistry. J Chem Inf Comput Sci 44:2051–2060

81. Ludwig K, Speiser B (2006) EChem++ - an object-oriented problem solving environment for electrochemistry: Part 4. Adaptive multilevel finite elements applied to electrochemical models. Algorithm and benchmark calculations. J Electroanal Chem 588:74–87
82. Ludwig K, Speiser B (2007) EChem++ - an object-oriented problem solving environment for electrochemistry. Part 5. A differential-algebraic approach to the error control of adaptive algorithms. J Electroanal Chem 608:91–101
83. Ludwig K, Morales I, Speiser B (2007) EChem++ - an object-oriented problem solving environment for electrochemistry. Part 6. Adaptive finite element simulations of controlled-current electrochemical experiments. J Electroanal Chem 608:102–110
84. Sapozhnikova EP, Bogdan M, Speiser B, Rosenstiel W (2006) EChem++ - an object-oriented problem solving environment for electrochemistry: 3. Classification of voltammetric signals by the fuzzy ARTMAP neural network with respect to reaction mechanisms. J Electroanal Chem 588:15–26
85. Benthin S, Speiser B (2012) EChem++ - an object-oriented problem solving environment for electrochemistry. Part 7: simulation of equilibrium electron transfer processes with implicit Dirichlet boundary conditions. J Eletroanal Chem 682:147–157
86. http://opensource.org/licenses/gpl-license.php
87. http://sourceforge.net/projects/echempp
88. http://www.kanazawa-bidai.ac.jp/~momo/qrcv/QRCV.html
89. http://lem.ch.unito.it/chemistry/esp_manual.html
90. http://www.condecon.com
91. http://transient.mkolar.org/
92. Sanchez G, Codina G, Aldaz A (1991) A voltammetry experiment by digital simulation. J Chem Educ 68:489–490
93. Svir IB, Oleinick AI, Klimenko AV (2001) 'ECL-PACKAGE' - software for electrochemiluminescence simulation at microelectrodes. J Electroanal Chem 513:119–125
94. Penar J, Persona A, Stawinski A (1993) Computer programs for teaching the principles of the methods of rotating disc and rotating ring-disc electrodes in electrochemical kinetics investigations. Pol J Chem 67:529–540
95. Villa CM, Chapman TW (1995) Simulation of complex electrochemical reaction systems. Ind Eng Chem Res 34:3445–3453
96. Bieniasz LK (1996) A reaction compiler for electrochemical kinetics. Comput Chem 20:403–418
97. Luo W, Feldberg SW, Rudolph M (1994) Ensuring self-consistent assignment of thermodynamic parameters in simulations of electrochemical-chemical systems. J Electroanal Chem 368:109–113
98. Ludwig K, Speiser B (2006) ERRATA. EChem++ - an object-oriented problem solving environment for electrochemistry. 2. The kinetic facilities of Ecco - a compiler for (electro-)chemistry. J Chem Inf Model 46:2762
99. Nelder JA, Mead R (1965) A simplex method for function minimization. Comput J 7:308–313
100. Caceci MS, Cacheris WP (1984) Fitting curves to data. The Simplex algorithm is the answer. Byte 9:340–362
101. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical recipes in Fortran. The art of scientific computing, 2nd edn. Cambridge University Press, Cambridge
102. Hildebrand FB (1974) Introduction to numerical analysis, 2nd edn. Dover Publications, New York
103. Cheney W, Kincaid D (1985) Numerical mathematics and computing. Brooks/Cole, Belmont, CA
104. Levenberg K (1944) A method for the solution of certain nonlinear problems in least squares. Q Appl Math 2:164–168
105. Marquardt DW (1963) An algorithm for least-squares estimation of nonlinear parameters. J Soc Ind Appl Math 11:431–441
106. Bott AW, Feldberg SW, Rudolph M (1996) Fitting experimental cyclic voltammetry data with theoretical simulations using DigiSim 2.1. Curr Sep 15:67–71

107. Lavagnini I, Pastore P, Magno F (1989) Comparison of the Simplex, Marquardt, and extended and iterated extended Kalman filter procedures in the estimation of parameters from voltammetric curves. Anal Chim Acta 223:193–204
108. Strutwolf J (1995) Digitale Simulation elektrochemischer Systeme: Untersuchungen zeitabhängiger Phänomene an rotierenden Scheibenelektroden und Analyse von Cyclovoltammogrammen durch direkte Simulation. Ph.D. thesis, Universität Bielefeld, Bielefeld
109. Phillips GR, Eyring EM (1988) Error estimation using the sequential Simplex method in nonlinear least squares data analysis. Anal Chem 60:738–741
110. Brumby S (1989) Exchange of comments on the simplex algorithm culminating in quadratic convergence and error estimation. Anal Chem 61:1783–1786
111. Brumby S (1992) ESR spectrum simulation: the simplex algorithm with quadratic convergence and error estimation. Appl Spectrosc 46:176–178

# Erratum to: Digital Simulation in Electrochemistry

## Fourth Edition

**Dieter Britz, Jörg Strutwolf**

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8

**DOI 10.1007/978-3-319-30292-8_18**

### Chapter 13: Migrational Effects

On page 365, the term “Electicität” in reference [6] was replaced with “Electricität”. On page 366, the author name “BLorenz” in reference [34] was replaced with “Lorentz”.

### Appendix B : Transforming the Diffusion Equation into Curvilinear Coordinates

On page 453 there were errors in the expressions (B.40) and (B.41).

$$\frac{\partial Z}{\partial \varGamma} = \frac{\partial Z}{\partial \varGamma} \frac{1}{\sin\theta \cosh \varGamma} \tag{B.40}$$

Corrected as

$$\frac{\partial C}{\partial Z} = \frac{\partial C}{\partial \Gamma} \frac{1}{\sin\theta \cosh \Gamma} \tag{B.40}$$

The updated original online version for this book can be found at
DOI 10.1007/978-3-319-30292-8

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8_18

$$I = \frac{\pi}{2} \int_0^{\pi/2} \cos\theta \left.\frac{\partial Z}{\partial \mathit{\Gamma}}\right|_{\mathit{\Gamma}=0} \mathrm{d}\theta. \tag{B.41}$$

Corrected as

$$I = \frac{\pi}{2} \int_0^{\pi/2} \cos\theta \left.\frac{\partial C}{\partial \Gamma}\right|_{\Gamma=0} \mathrm{d}\theta. \tag{B.41}$$

# Appendix A
# Tables and Formulae

## A.1 First Derivative Approximations

In Table A.1, the coefficients are multiplied by $m$ (given in the first coefficients column), so as to get whole numbers.

## A.2 Current Approximations

These are simply obtained by using the appropriate coefficients in Table A.1, for the $y'_1(n)$ form chosen. For example, the two-point form is (3.23)

$$G \approx \frac{1}{H}(-C_0 + C_1) \tag{A.1}$$

while the three-point formula is (3.24),

$$G \approx \frac{1}{2H}(-3C_0 + 4C_1 - C_2) \ . \tag{A.2}$$

Note that all these have been cast as the function $\mathcal{G}(C, n, H)$, defined in Chap. 3, page 45.

## A.3 Second Derivative Approximations

Table A.2 shows the coefficients for these. As in Table A.1, the coefficients are multiplied by $m$ (given in the first coefficients column), so as to get whole numbers.

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8

**Table A.1** $m\beta$ (Eq. 3.14) for multi-point first derivatives on equally spaced points

| | $m$ | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ | $y_6$ | $y_7$ | Order |
|---|---|---|---|---|---|---|---|---|---|
| $y'_1(2)$ | 1 | $-1$ | 1 | | | | | | $h$ |
| $y'_2(2)$ | 1 | $-1$ | 1 | | | | | | $h$ |
| $y'_1(3)$ | 2 | $-3$ | 4 | $-1$ | | | | | $h^2$ |
| $y'_2(3)$ | 2 | $-1$ | 0 | 1 | | | | | $h^2$ |
| $y'_3(3)$ | 2 | 1 | $-4$ | 3 | | | | | $h^2$ |
| $y'_1(4)$ | 6 | $-11$ | 18 | $-9$ | 2 | | | | $h^3$ |
| $y'_2(4)$ | 6 | $-2$ | $-3$ | 6 | $-1$ | | | | $h^3$ |
| $y'_3(4)$ | 6 | 1 | $-6$ | 3 | 2 | | | | $h^3$ |
| $y'_4(4)$ | 6 | $-2$ | 9 | $-18$ | 11 | | | | $h^3$ |
| $y'_1(5)$ | 12 | $-25$ | 48 | $-36$ | 16 | $-3$ | | | $h^4$ |
| $y'_2(5)$ | 12 | $-3$ | $-10$ | 18 | $-6$ | 1 | | | $h^4$ |
| $y'_3(5)$ | 12 | 1 | $-8$ | 0 | 8 | $-1$ | | | $h^4$ |
| $y'_4(5)$ | 12 | $-1$ | 6 | $-18$ | 10 | 3 | | | $h^4$ |
| $y'_5(5)$ | 12 | 3 | $-16$ | 36 | $-48$ | 25 | | | $h^4$ |
| $y'_1(6)$ | 60 | $-137$ | 300 | $-300$ | 200 | $-75$ | 12 | | $h^5$ |
| $y'_2(6)$ | 60 | $-12$ | $-65$ | 120 | $-60$ | 20 | $-3$ | | $h^5$ |
| $y'_3(6)$ | 60 | 3 | $-30$ | $-20$ | 60 | $-15$ | 2 | | $h^5$ |
| $y'_4(6)$ | 60 | $-2$ | 15 | $-60$ | 20 | 30 | $-3$ | | $h^5$ |
| $y'_5(6)$ | 60 | 3 | $-20$ | 60 | $-120$ | 65 | 12 | | $h^5$ |
| $y'_6(6)$ | 60 | $-12$ | 75 | $-200$ | 300 | $-300$ | 137 | | $h^5$ |
| $y'_1(7)$ | 60 | $-147$ | 360 | $-450$ | 400 | $-225$ | 72 | $-10$ | $h^6$ |
| $y'_2(7)$ | 60 | $-10$ | $-77$ | 150 | $-100$ | 50 | $-15$ | 2 | $h^6$ |
| $y'_3(7)$ | 60 | 2 | $-24$ | $-35$ | 80 | $-30$ | 8 | $-1$ | $h^6$ |
| $y'_4(7)$ | 60 | $-1$ | 9 | $-45$ | 0 | 45 | $-9$ | 1 | $h^6$ |
| $y'_5(7)$ | 60 | 1 | $-8$ | 30 | $-80$ | 35 | 24 | $-2$ | $h^6$ |
| $y'_6(7)$ | 60 | $-2$ | 15 | $-50$ | 100 | $-150$ | 77 | 10 | $h^6$ |
| $y'_7(7)$ | 60 | 10 | $-72$ | 225 | $-400$ | 450 | $-360$ | 147 | $h^6$ |

The notation $y'_i(n)$ means the approximation at point $i$ using $n$ points numbered $1 \ldots n$

## A.4 Unequal Intervals

For arbitrarily spaced point sequences, the very general subroutines `FORNBERG` and `FORN`, referred to in Appendix E, can provide both the derivatives and the coefficients. However, the reader may wish to restrict the expressions to those involving only up to four points (for which there are some good arguments, see Chap. 8, Sect. 8.4). This can be coupled with current approximations using up to four points. For this number of points, the expressions are not unreasonably long, and we present them here. These should in principle be slightly more accurate than those derived from `FORN`, as these use approximate methods, whereas the following are exact solutions.

**Table A.2** Coefficients for some chosen multi-point second derivatives on equally spaced points

| | $m$ | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ | $y_6$ | $y_7$ | Order |
|---|---|---|---|---|---|---|---|---|---|
| $y_1''(3)$ | 1 | 1 | −2 | 1 | | | | | $h$ |
| $y_2''(3)$ | 1 | −1 | −2 | 1 | | | | | $h^2$ |
| $y_3''(3)$ | 1 | 1 | −2 | 1 | | | | | $h$ |
| $y_1''(4)$ | 1 | 2 | −5 | 4 | −1 | | | | $h^2$ |
| $y_2''(4)$ | 1 | 1 | −2 | 1 | 0 | | | | $h^2$ |
| $y_3''(4)$ | 1 | 0 | 1 | −2 | 1 | | | | $h^2$ |
| $y_4''(4)$ | 1 | 1 | 4 | −5 | 2 | | | | $h^2$ |
| $y_1''(5)$ | 12 | 35 | −104 | 114 | −56 | 11 | | | $h^3$ |
| $y_2''(5)$ | 12 | 11 | −20 | 6 | 4 | −1 | | | $h^3$ |
| $y_3''(5)$ | 12 | −1 | 16 | −30 | 16 | −1 | | | $h^4$ |
| $y_4''(5)$ | 12 | −1 | 4 | 6 | −20 | 11 | | | $h^3$ |
| $y_5''(5)$ | 12 | 11 | −56 | 114 | −104 | 35 | | | $h^3$ |
| $y_1''(6)$ | 12 | 45 | −154 | 214 | −156 | 61 | −10 | | $h^4$ |
| $y_2''(6)$ | 12 | 10 | −15 | −4 | 14 | −6 | 1 | | $h^4$ |
| $y_3''(6)$ | 12 | −1 | 16 | −20 | 16 | −1 | 0 | | $h^4$ |
| $y_4''(6)$ | 12 | 0 | −1 | 16 | −30 | 16 | −1 | | $h^4$ |
| $y_5''(6)$ | 12 | 1 | −6 | 14 | −4 | −15 | 10 | | $h^4$ |
| $y_6''(6)$ | 12 | −10 | 61 | −156 | 214 | −154 | 45 | | $h^4$ |
| $y_1''(7)$ | 180 | 812 | −3132 | 5265 | −5080 | 2970 | −972 | 137 | $h^5$ |
| $y_2''(7)$ | 180 | 137 | −147 | −255 | 470 | −285 | 93 | −13 | $h^5$ |
| $y_3''(7)$ | 180 | −13 | 228 | −420 | 200 | 15 | −12 | 2 | $h^5$ |
| $y_4''(7)$ | 180 | 2 | −27 | 270 | −490 | 270 | −27 | 2 | $h^6$ |
| $y_5''(7)$ | 180 | 2 | −12 | 15 | 200 | −420 | 228 | −13 | $h^5$ |
| $y_6''(7)$ | 180 | −13 | 93 | −285 | 470 | −225 | −147 | 137 | $h^5$ |
| $y_7''(7)$ | 180 | 137 | −972 | 2970 | −5080 | 5265 | −3112 | 812 | $h^5$ |

As in Chap. 3, the notations $u_i'(n)$ and $u_i''(n)$ are for a number $n$ of positions $x_1 \dots x_n$, at which some function values, respectively $u_1 \dots u_n$ are defined, and the derivatives refer to point $i$ out of the $n$. We also have a set of displacements $h_k = x_k - x_i$, in each case the zero displacement $h_i$ missing from the set. Some of the following formulae have been given previously by Gavaghan [1] and Rudolph [2] but with different notation and different convention for the displacements. The method used to derive these is that described in Chap. 3 on page 56.

Table A.3 shows the first derivative coefficients, while Table A.4 those for second derivatives.

**Table A.3** First derivative approximations $u_i'(n)$ for arbitrarily spaced $n$ points ($n = 3, 4$) at positions $x_1 \dots x_n$, each point at an offset $h_k = x(k) - x(i)$ with respect to the reference point at index $i$

| Form | Parameter | Expression |
|---|---|---|
| $u_1'(3)$ | $\Delta$ | $h_2 h_3 (h_3 - h_2)$ |
| | $\beta_1$ | $-(\beta_2 + \beta_3)$ |
| | $\beta_2$ | $h_3^2/\Delta$ |
| | $\beta_3$ | $-h_2^2/\Delta$ |
| $u_2'(3)$ | $\Delta$ | $h_1 h_3 (h_3 - h_1)$ |
| | $\beta_1$ | $h_3^2/\Delta$ |
| | $\beta_2$ | $-(\beta_1 + \beta_3)$ |
| | $\beta_3$ | $-h_1^2/\Delta$ |
| $u_3'(3)$ | $\Delta$ | $h_1 h_2 (h_2 - h_1)$ |
| | $\beta_1$ | $h_2^2/\Delta$ |
| | $\beta_2$ | $-h_1^2/\Delta$ |
| | $\beta_3$ | $-(\beta_1 + \beta_2)$ |
| $u_1''(4)$ | $\Delta$ | $h_2 h_3^2 h_4^2 (h_4 - h_3) - h_3 h_2^2 h_4^2 (h_4 - h_2) + h_4 h_2^2 h_3^2 (h_3 - h_2)$ |
| | $\beta_1$ | $-(\beta_2 + \beta_3 + \beta_4)$ |
| | $\beta_2$ | $h_3^2 h_4^2 (h_4 - h_3)/\Delta$ |
| | $\beta_3$ | $h_2^2 h_4^2 (h_2 - h_4)/\Delta$ |
| | $\beta_4$ | $h_2^2 h_3^2 (h_3 - h_2)/\Delta$ |
| $u_2''(4)$ | $\Delta$ | $h_1 h_3^2 h_4^2 (h_4 - h_3) - h_3 h_1^2 h_4^2 (h_4 - h_1) + h_4 h_1^2 h_3^2 (h_3 - h_1)$ |
| | $\beta_1$ | $h_3^2 h_4^2 (h_4 - h_3)/\Delta$ |
| | $\beta_2$ | $-(\beta_1 + \beta_3 + \beta_4)$ |
| | $\beta_3$ | $h_1^2 h_4^2 (h_1 - h_4)/\Delta$ |
| | $\beta_4$ | $h_1^2 h_3^2 (h_3 - h_1)/\Delta$ |
| $u_3''(4)$ | $\Delta$ | $h_1 h_2^2 h_4^2 (h_4 - h_2) - h_1^2 h_2 h_4^2 (h_4 - h_1) + h_1^2 h_2^2 h_4 (h_2 - h_1)$ |
| | $\beta_1$ | $h_2^2 h_4^2 (h_4 - h_2)/\Delta$ |
| | $\beta_2$ | $h_1^2 h_4^2 (h_1 - h_4)/\Delta$ |
| | $\beta_3$ | $-(\beta_1 + \beta_2 + \beta_4)$ |
| | $\beta_4$ | $h_1^2 h_2^2 (h_2 - h_1)/\Delta$ |
| $u_4''(4)$ | $\Delta$ | $h_1 h_2^2 h_3^2 (h_3 - h_2) - h_1^2 h_2 h_3^2 (h_3 - h_1) + h_1^2 h_2^2 h_3 (h_2 - h_1)$ |
| | $\beta_1$ | $h_2^2 h_3^2 (h_3 - h_2)/\Delta$ |
| | $\beta_2$ | $h_1^2 h_3^2 (h_1 - h_3)/\Delta$ |
| | $\beta_3$ | $h_1^2 h_2^2 (h_2 - h_1)/\Delta$ |
| | $\beta_4$ | $-(\beta_1 + \beta_2 + \beta_3)$ |

The symbol $\Delta$ denotes the determinant of the generating matrix. All three-point forms are $O(h^2)$ and all four-point forms are $O(h^3)$

**Table A.4** Second derivative approximations $u_i''(n)$ for arbitrarily spaced $n$ points ($n = 3, 4$) at positions $x_1 \ldots x_n$, each point at an offset $h_k = x(k) - x(i)$ with respect to the reference point at index $i$

| Form | Parameter | Expression |
|---|---|---|
| $u_1''(3)$ | $\Delta$ | $h_2 h_3 (h_3 - h_2)$ |
| | $\alpha_1$ | $-(\alpha_2 + \alpha_3)$ |
| | $\alpha_2$ | $-2h_3/\Delta$ |
| | $\alpha_3$ | $2h_2/\Delta$ |
| $u_2''(3)$ | $\Delta$ | $h_1 h_3 (h_3 - h_1)$ |
| | $\alpha_1$ | $-2h_3/\Delta$ |
| | $\alpha_2$ | $-(\alpha_1 + \alpha_3)$ |
| | $\alpha_3$ | $2h_1/\Delta$ |
| $u_3''(3)$ | $\Delta$ | $h_1 h_2 (h_2 - h_1)$ |
| | $\alpha_1$ | $-2h_2/\Delta$ |
| | $\alpha_2$ | $2h_1/\Delta$ |
| | $\alpha_3$ | $-(\alpha_1 + \alpha_2)$ |
| $u_1''(4)$ | $\Delta$ | $h_2 h_3^2 h_4^2 (h_4 - h_3) - h_3 h_2^2 h_4^2 (h_4 - h_2) + h_4 h_2^2 h_3^2 (h_3 - h_2)$ |
| | $\alpha_1$ | $-(\alpha_2 + \alpha_3 + \alpha_4)$ |
| | $\alpha_2$ | $2h_3 h_4 (h_3^2 - h_4^2)/\Delta$ |
| | $\alpha_3$ | $2h_2 h_4 (h_4^2 - h_2^2)/\Delta$ |
| | $\alpha_4$ | $2h_2 h_3 (h_2^2 - h_3^2)/\Delta$ |
| $u_2''(4)$ | $\Delta$ | $h_1 h_3^2 h_4^2 (h_4 - h_3) - h_3 h_1^2 h_4^2 (h_4 - h_1) + h_4 h_1^2 h_3^2 (h_3 - h_1)$ |
| | $\alpha_1$ | $2h_3 h_4 (h_3^2 - h_4^2)/\Delta$ |
| | $\alpha_2$ | $-(\alpha_1 + \alpha_3 + \alpha_4)$ |
| | $\alpha_3$ | $2h_1 h_4 (h_4^2 - h_1^2)/\Delta$ |
| | $\alpha_4$ | $2h_1 h_3 (h_1^2 - h_3^2)/\Delta$ |
| $u_3''(4)$ | $\Delta$ | $h_1 h_2^2 h_4^2 (h_4 - h_2) - h_1^2 h_2 h_4^2 (h_4 - h_1) + h_1^2 h_2^2 h_4 (h_2 - h_1)$ |
| | $\alpha_1$ | $-2h_2 h_4 (h_4^2 - h_2^2)/\Delta$ |
| | $\alpha_2$ | $2h_1 h_4 (h_4^2 - h_1^2)/\Delta$ |
| | $\alpha_3$ | $-(\alpha_1 + \alpha_2 + \alpha_4)$ |
| | $\alpha_4$ | $-2h_1 h_2 (h_2^2 - h_1^2)/\Delta$ |
| $u_4''(4)$ | $\Delta$ | $h_1 h_2^2 h_3^2 (h_3 - h_2) - h_1^2 h_2 h_3^2 (h_3 - h_1) + h_1^2 h_2^2 h_3 (h_2 - h_1)$ |
| | $\alpha_1$ | $-2h_2 h_3 (h_3^2 - h_2^2)/\Delta$ |
| | $\alpha_2$ | $2h_1 h_3 (h_3^2 - h_1^2)/\Delta$ |
| | $\alpha_3$ | $-2h_1 h_2 (h_2^2 - h_1^2)/\Delta$ |
| | $\alpha_4$ | $-(\alpha_1 + \alpha_2 + \alpha_3)$ |

The symbol $\Delta$ denotes the determinant of the generating matrix. All three-point forms are $O(h)$ and all four-point forms are $O(h^2)$

## A.5 Jacobi Roots for Orthogonal Collocation

Tables A.5, A.6, and A.7 provides the roots of the Jacobi polynomials used as node points in orthogonal collocation, for some values of $N$. Values for $X = 0 \quad (i = 0)$ and $X = 1 \quad (i = N + 1)$ (0 and 1, resp.) are not included. The roots were computed using the subroutine `JCOBI`, modified from the original of Villadsen and Michelsen [3], discussed in Appendix E, using for a given $N$ the call

```
    CALL JCOBI (N+1, N, 0, 0, 0.0_dbl, 0.0_dbl, ...)
```

Roots for higher $N$, if required, can be computed using this subroutine.

**Table A.5** Jacobi polynomial roots, $N = 3 \ldots 6$

| $N = 3$ | $N = 4$ | $N = 5$ | $N = 6$ |
|---|---|---|---|
| 0.11270166537926 | 0.06943184420297 | 0.04691007703067 | 0.03376524289842 |
| 0.50000000000000 | 0.33000947820757 | 0.23076534494716 | 0.16939530676687 |
| 0.88729833462074 | 0.66999052179243 | 0.50000000000000 | 0.38069040695840 |
|  | 0.93056815579703 | 0.76923465505284 | 0.61930959304160 |
|  |  | 0.95308992296933 | 0.83060469323313 |
|  |  |  | 0.96623475710158 |

**Table A.6** Jacobi polynomial roots, $N = 7 \ldots 10$

| $N = 7$ | $N = 8$ | $N = 9$ | $N = 10$ |
|---|---|---|---|
| 0.02544604382862 | 0.01985507175123 | 0.01591988024619 | 0.01304673574141 |
| 0.12923440720030 | 0.10166676129319 | 0.08198444633668 | 0.06746831665551 |
| 0.29707742431130 | 0.23723379504184 | 0.19331428364970 | 0.16029521585049 |
| 0.50000000000000 | 0.40828267875218 | 0.33787328829810 | 0.28330230293538 |
| 0.70292257568870 | 0.59171732124782 | 0.50000000000000 | 0.42556283050918 |
| 0.87076559279970 | 0.76276620495816 | 0.66212671170190 | 0.57443716949082 |
| 0.97455395617138 | 0.89833323870681 | 0.80668571635030 | 0.71669769706462 |
|  | 0.98014492824877 | 0.91801555366332 | 0.83970478414951 |
|  |  | 0.98408011975381 | 0.93253168334449 |
|  |  |  | 0.98695326425859 |

**Table A.7** Jacobi polynomial roots, $N = 11 \dots 14$

| $N = 11$ | $N = 12$ | $N = 13$ | $N = 14$ |
|---|---|---|---|
| 0.01088567092697 | 0.00921968287664 | 0.00790847264071 | 0.00685809565159 |
| 0.05646870011595 | 0.04794137181476 | 0.04120080038851 | 0.03578255816821 |
| 0.13492399721298 | 0.11504866290285 | 0.09921095463335 | 0.08639934246512 |
| 0.24045193539659 | 0.20634102285669 | 0.17882533027983 | 0.15635354759416 |
| 0.36522842202383 | 0.31608425050091 | 0.27575362448178 | 0.24237568182092 |
| 0.50000000000000 | 0.43738329574427 | 0.38477084202243 | 0.34044381553606 |
| 0.63477157797617 | 0.56261670425573 | 0.50000000000000 | 0.44597252564633 |
| 0.75954806460341 | 0.68391574949909 | 0.61522915797757 | 0.55402747435367 |
| 0.86507600278702 | 0.79365897714331 | 0.72424637551822 | 0.65955618446394 |
| 0.94353129988405 | 0.88495133709715 | 0.82117466972017 | 0.75762431817908 |
| 0.98911432907303 | 0.95205862818524 | 0.90078904536665 | 0.84364645240584 |
| | 0.99078031712336 | 0.95879919961149 | 0.91360065753488 |
| | | 0.99209152735929 | 0.96421744183179 |
| | | | 0.99314190434841 |

## A.6 Rosenbrock Constants

The Rosenbrock method is described for *ode*s in Chap. 4 and for electrochemical simulations, that is, DAEs, in Chap. 9. There are four variants, two of them second-order with respect to the time interval, and two of them third-order, that are considered in these chapters. Although only two variants recommend themselves, the constants for all four are given here. For the notation and the meaning of the variant names, see these chapters. The notation is in some cases not that of the (cited) sources. Constants that are left out can be taken as zero.

RO2, Ref. [4], see Sect. 4.12

$\gamma = 1 - \frac{1}{2}\sqrt{2}$

$a_{21} = \frac{1}{2}(\sqrt{2} - 1); \quad \alpha_2 = a_{21}$

$m_1 = 0; \quad m_2 = 1$

ROS2, Ref. [5]

$\gamma = 1.707106781186547$

$\gamma_1 = \gamma; \quad \gamma_2 = -\gamma$

$\alpha_1 = 0; \quad \alpha_2 = 1$

$a_{21} = 0.5857864376269050$

$c_{21} = -1.171572875253810$

$m_1 = 0.8786796564403575; \quad m_2 = 0.2928932188134525$

ROWDA3, Refs. [6, 7]

$\gamma = 0.4358665215084590$

$\gamma_1 = \gamma$

$\gamma_2 = 0.6044552840655590$

$\gamma_3 = 6.379788799344883$

$\alpha_2 = 0.7; \quad \alpha_3 = 0.7$<br>
$a_{21} = 1.605996252195329$
$a_{31} = a_{21}; \quad a_{32} = 0$
$c_{21} = 0.8874044410657833$
$c_{31} = 23.98747971635036$
$c_{32} = 5.263722371562129$
$m_1 = 2.236727045296590$
$m_2 = 2.250067730969644$
$m_3 = -0.2092514044390320$

ROS3P, Refs. [5, 8]

$\gamma = 0.7886751345948129$
$\gamma_1 = \gamma$
$\gamma_2 = -0.2113248654051871$
$\gamma_3 = -1.077350269189626$
$a_{21} = 1.267949192431123; \quad a_{31} = a_{21}$
$c_{21} = -1.607695154586736$
$c_{31} = -3.464101615137755$
$c_{32} = -1.732050807568877$
$m_1 = 2$
$m_2 = 0.5773502691896258$
$m_3 = 0.4226497308103742$

## References

1. Gavaghan DJ (1998) An exponentially expanding mesh ideally suited to the fast and efficient simulation of diffusion processes at microdisc electrodes. 1. Derivation of the mesh. J Electroanal Chem 456:1–12
2. Rudolph M (2002) Digital simulation on unequally spaced grids. Part 1. Critical remarks on using the point method by discretisation on a transformed grid. J Electroanal Chem 529:97–108
3. Villadsen J, Michelsen ML (1978) Solution of differential equation models by polynomial approximation. Prentice-Hall, New Jersey
4. Rosenbrock H (1962/1963) Some general implicit processes for the numerical solution of differential equations. Comput J 5:329–330
5. Lang J (2001) Adaptive multilevel solution of nonlinear parabolic PDE systems. Springer, Berlin
6. Roche M (1988) Rosenbrock methods for differential algebraic equations. Numer Math 52:45–63
7. Bieniasz LK (1999) Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme. J Electroanal Chem 469:97–115
8. Lang J, Verwer J (2001) ROS3P - an accurate third-order Rosenbrock solver designed for parabolic problems. BIT Numer Math 41:731–738

# Appendix B
# Transforming the Diffusion Equation into Curvilinear Coordinates

## B.1 Introduction

The method can be found in Morse and Feshbach [1, p. 115]. The description is simpler than in Bland [2], although the two are in principle the same method. Boas [3] details the way the method comes about. It rests on a general expression of the *pde* in a new coordinate system, and expressions for the scaling factors for the displacement of a point in the original coordinate system expressed as a corresponding displacement in the new system. In this way, transformations into a new coordinate system can be performed with ease, in contrast to the way Verbrugge and Baker did it [4], which is very laborious.

In Cartesian coordinates, if a given point is moved a small distance in all three coordinates, the net displacement d$s$ is given by

$$\mathrm{d}s^2 = \mathrm{d}x^2 + \mathrm{d}y^2 + \mathrm{d}z^2\ . \tag{B.1}$$

In another coordinate system with coordinates $q_1, q_2, q_3$, the same displacement can be expressed as

$$\mathrm{d}s^2 = h_1^2\,\mathrm{d}q_1^2 + h_2^2\,\mathrm{d}q_2^2 + h_3^2\,\mathrm{d}q_3^2 \tag{B.2}$$

where the $h$'s are scaling factors to be determined from the transformation. They are determined in the following manner. For transformation into the new coordinate system $(q_1, q_2, q_3)$, we can write

$$\mathrm{d}x = \frac{\partial x}{\partial q_1}\mathrm{d}q_1 + \frac{\partial x}{\partial q_2}\mathrm{d}q_2 + \frac{\partial x}{\partial q_3}\mathrm{d}q_3 \tag{B.3}$$

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8

and so on for d$y$ and d$z$. Substitution in (B.1) results in an expression in the three new differentials, with scaling factors $h_n$, $n = 1, 2, 3$, given by [1]

$$\begin{aligned} h_n^2 &= \left(\frac{\partial x}{\partial q_n}\right)^2 + \left(\frac{\partial y}{\partial q_n}\right)^2 + \left(\frac{\partial z}{\partial q_n}\right)^2 \\ &= \left(\left(\frac{\partial q_n}{\partial x}\right)^2 + \left(\frac{\partial q_n}{\partial y}\right)^2 + \left(\frac{\partial q_n}{\partial z}\right)^2\right)^{-1} \end{aligned} \tag{B.4}$$

and the final Laplacian by

$$\nabla^2 C = \frac{1}{h_1 h_2 h_3} \sum_n \frac{\partial}{\partial q_n} \left(\frac{h_1 h_2 h_3}{h_n^2} \frac{\partial C}{\partial q_n}\right) , \tag{B.5}$$

which is the key to all solutions.

Starting from a curvilinear coordinate system $(p_1, p_2, p_3)$ with its own scaling factors $(g_1, g_2, g_3)$, the new equation for d$s^2$ has scaling factors given by (B.4) but each one multiplied by its corresponding prior $g_n$. This comes into play when starting from cylindrical coordinates where we have the existing scaling factor $R$, as will be seen below.

## B.2 A Simple Example: Cartesian to Cylindrical

This serves as a simple example, and establishes the base for the transformations of the equation for the disk system. Coordinates $(X, Y, Z)$ to $(R, \phi, Z)$ ($\phi$ being the rotational angle around the vertical axis) are mapped into new coordinates

$$\begin{aligned} X &= R \cos\phi \\ Y &= R \sin\phi \\ Z &= Z \end{aligned} \tag{B.6}$$

(note that $Z$, pointing upwards, is not changed). Going directly to the scaling factors, using (B.4) obtains

$$\begin{aligned} h_1^2 &= \left(\frac{\partial X}{\partial R}\right)^2 + \left(\frac{\partial Y}{\partial R}\right)^2 + \left(\frac{\partial Z}{\partial R}\right)^2 \\ &= \cos^2(\phi) + \sin^2(\phi) + 0 \qquad (B.7) \\ &= 1 , \qquad (B.8) \end{aligned}$$

$$\begin{aligned} h_2^2 &= \left(\frac{\partial X}{\partial \phi}\right)^2 + \left(\frac{\partial Y}{\partial \phi}\right)^2 + \left(\frac{\partial Z}{\partial \phi}\right)^2 \\ &= R^2 \sin^2(\phi) + R^2 \cos^2(\phi) + 0 \qquad \text{(B.9)} \\ &= R^2 \qquad \text{(B.10)} \end{aligned}$$

and

$$\begin{aligned} h_3^2 &= \left(\frac{\partial X}{\partial Z}\right)^2 + \left(\frac{\partial Y}{\partial Z}\right)^2 + \left(\frac{\partial Z}{\partial Z}\right)^2 \\ &= 0 + 0 + 1 \qquad \text{(B.11)} \\ &= 1\ . \end{aligned}$$

These yield the displacement equation

$$\mathrm{d}s^2 = \mathrm{d}R^2 + R^2 \mathrm{d}\phi^2 + \mathrm{d}Z^2 \qquad \text{(B.12)}$$

which will serve as a base for the disk transformations with $g_2 = R$. The Laplacian for this system is then

$$\nabla^2 C = \frac{1}{R}\left[\frac{\partial}{\partial R}\left(R\frac{\partial C}{\partial R}\right) + \frac{\partial}{\partial \phi}\left(\frac{1}{R}\frac{\partial C}{\partial \phi}\right) + \frac{\partial}{\partial Z}\left(R\frac{\partial C}{\partial Z}\right)\right] \qquad \text{(B.13)}$$

which develops to

$$\nabla^2 C = \frac{1}{R}\left(R\frac{\partial^2 C}{\partial R^2} + \frac{\partial C}{\partial R} + \frac{1}{R}\frac{\partial^2 C}{\partial \phi^2} + R\frac{\partial^2 C}{\partial Z^2}\right) \qquad \text{(B.14)}$$

or, finally, in view of $\partial C/\partial \phi = \partial^2 C/\partial \phi^2 = 0$,

$$\nabla^2 C = \frac{\partial^2 C}{\partial R^2} + \frac{1}{R}\frac{\partial C}{\partial R} + \frac{\partial^2 C}{\partial Z^2}\ . \qquad \text{(B.15)}$$

In general, in the following, the final equation in the new coordinates $(\theta, \phi, \Gamma)$ will be represented in the form

$$\nabla^2 C = \frac{1}{F}\left(a_\theta \frac{\partial^2 C}{\partial \theta^2} + b_\theta \frac{\partial C}{\partial \theta} + a_\Gamma \frac{\partial^2 C}{\partial \Gamma^2} + b_\Gamma \frac{\partial C}{\partial \Gamma}\right) \qquad \text{(B.16)}$$

so it remains to identify the coefficients (the coefficients for derivatives in $\phi$ (disk) or $y$ (band) are zero). Also, for the development it is helpful to write the general form of the equation to be derived as

$$\nabla^2 C = \frac{1}{h_1 h_2 h_3}\left[\frac{\partial}{\partial \theta}\left(u\frac{\partial C}{\partial \theta}\right) + \frac{\partial}{\partial \Gamma}\left(w\frac{\partial C}{\partial \Gamma}\right)\right]\ . \qquad \text{(B.17)}$$

(there is a $v$-term for $\phi$ or $y$ but that whole term is zero and therefore left out). The three terms $u$, $v$ and $w$ are evident from (B.5).

## B.3 Transformations for the Band Electrode

A concrete example is the transformation from Cartesian to curvilinear coordinates for the band, using two chosen transformations. The new coordinates are $(\theta, Y, \Gamma)$.

### *B.3.1 Cartesian to MWA*

Here we have $X$ across the band, $Y$ along the (relatively long) band length (along which there are no gradients) and $Z$ pointing up, normal to the band. The MWA transformations are

$$\begin{aligned} X &= \cos\theta \cosh\Gamma \\ Y &= Y \\ Z &= \sin\theta \sinh\Gamma \end{aligned} \tag{B.18}$$

and (B.4) easily leads to

$$\begin{aligned} h_1 &= \sqrt{\sin^2\theta + \sinh^2\Gamma} \\ h_2 &= 1 \\ h_3 &= \sqrt{\sin^2\theta + \sinh^2\Gamma}\ . \end{aligned} \tag{B.19}$$

and so immediately (noting that $u = w = 1$)

$$\nabla^2 C = \frac{1}{\sin^2\theta + \sinh^2\Gamma} \left\{ \frac{\partial^2 C}{\partial\theta^2} + \frac{\partial^2 C}{\partial\Gamma^2} \right\}\ . \tag{B.20}$$

### *B.3.2 Extension to VB*

MWA itself is not very efficient, as noted in Chap. 12, and it is advantageous to extend it to VB, using the method of Verbrugge and Baker [4] (their (A6) and (A7)) (this is much easier than going directly to VB!). We need only transform the above $\Gamma$ to the new $\gamma$,

$$\Gamma = \frac{\gamma}{1-\gamma}\ . \tag{B.21}$$

and the converse

$$\gamma = \frac{\Gamma}{1+\Gamma} \tag{B.22}$$

and replace $\mathrm{d}^2/\mathrm{d}\Gamma^2$ with the corresponding expression in $\gamma$. We note

$$\frac{\mathrm{d}\gamma}{\mathrm{d}\Gamma} = \frac{1}{(1+\Gamma)^2}\ . \tag{B.23}$$

We have for some $u$

$$\frac{\mathrm{d}u}{\mathrm{d}\Gamma} = \frac{\mathrm{d}\gamma}{\mathrm{d}\Gamma}\frac{\mathrm{d}u}{\mathrm{d}\gamma} \tag{B.24}$$

and using (B.21) and (B.23), this becomes

$$\frac{\mathrm{d}u}{\mathrm{d}\Gamma} = \frac{1}{(1+\Gamma)^2}\frac{\mathrm{d}u}{\mathrm{d}\gamma} = (1-\gamma)^2\frac{\mathrm{d}u}{\mathrm{d}\gamma}\ . \tag{B.25}$$

Differentiating again

$$\frac{\mathrm{d}^2u}{\mathrm{d}\Gamma^2} = \frac{\mathrm{d}}{\mathrm{d}\Gamma}\left(\frac{\mathrm{d}u}{\mathrm{d}\Gamma}\right) = \frac{\mathrm{d}\gamma}{\mathrm{d}\Gamma}\left(\frac{\mathrm{d}}{\mathrm{d}\gamma}\left(\frac{\mathrm{d}u}{\mathrm{d}\Gamma}\right)\right) \tag{B.26}$$

which in view of (B.23) and (B.25) becomes

$$\begin{aligned}\frac{\mathrm{d}^2u}{\mathrm{d}\Gamma^2} &= (1-\gamma^2)\frac{\mathrm{d}}{\mathrm{d}\gamma}\left((1-\gamma^2)\frac{\mathrm{d}u}{\mathrm{d}\gamma}\right)\\ &= (1-\gamma^2)\left(\frac{\mathrm{d}(1-\gamma)^2}{\mathrm{d}\gamma}\frac{\mathrm{d}u}{\mathrm{d}\gamma} + (1-\gamma)^2\frac{\mathrm{d}^2u}{\mathrm{d}\gamma^2}\right)\\ &= (1-\gamma)^4\frac{\mathrm{d}^2u}{\mathrm{d}\gamma^2} - 2(1-\gamma)^3\frac{\mathrm{d}u}{\mathrm{d}\gamma}\end{aligned} \tag{B.27}$$

now all in terms of $\gamma$.

## B.4 Disk Electrode Transformations

In all cases, we proceed from the cylindrical equation for the disk, and using (B.12) as start, there is a prior $g_2 = R$ (the other two $g$ factors are unity).

The equations for MWA are

$$\begin{aligned} R &= \cos\theta \cosh\mathit{\Gamma} \\ \phi &= \phi \\ Z &= \sin\theta \sinh\mathit{\Gamma} \end{aligned} \tag{B.28}$$

($\phi$ being the rotation angle, along which there are no gradients). The scale factors are

$$\begin{aligned} h_1 &= \sqrt{\sin^2\theta + \sinh^2\mathit{\Gamma}} \\ h_2 &= \cos\theta \cosh\mathit{\Gamma} \\ h_3 &= \sqrt{\sin^2\theta + \sinh^2\mathit{\Gamma}} \end{aligned} \tag{B.29}$$

(note that $h_1$ and $h_3$ are the same as for the band with MWA; the different $h_2$ term is of course due to the multiplication with $g_2 = R$ from (B.12)). So, letting $\mathit{\Delta} = \sqrt{\sin^2\theta + \sinh^2\mathit{\Gamma}}$, we have

$$h_1h_2h_3 = \mathit{\Delta}\cos\theta\cosh\mathit{\Gamma}\ , \tag{B.30}$$

$$u = \cos\theta\cosh\mathit{\Gamma}\ , \tag{B.31}$$

$$v = \frac{\mathit{\Delta}^2}{\cos\theta\cosh\mathit{\Gamma}}\ , \tag{B.32}$$

$$w = \cos\theta\cosh\mathit{\Gamma}\ , \tag{B.33}$$

leading to

$$\frac{\partial C}{\partial T} = \frac{1}{\sin^2\theta + \sinh^2\mathit{\Gamma}}\left[\frac{\partial^2 C}{\partial\theta^2} - \tan\theta\frac{\partial C}{\partial\theta} + \frac{\partial^2 C}{\partial\mathit{\Gamma}^2} + \tanh\mathit{\Gamma}\frac{\partial C}{\partial\mathit{\Gamma}}\right]\ . \tag{B.34}$$

Again, this can to advantage be extended to VB using the same equations (B.21) and (B.22), resulting in the form already presented in Table 12.1 on page 283 (where $\gamma$ is written as $\mathit{\Gamma}$).

## B.5 Transforming the Current

For the UMDE, the dimensionless current $I$ is given by the integral

$$I(T) = \frac{\pi}{2}\int_0^1 R\frac{\partial C}{\partial Z}\bigg|_{Z=0}\mathrm{d}R \tag{B.35}$$

in cylindrical coordinates and needs to be transformed along with the *pde*, depending on the transformation used. This is done as follows. $R$ is simply substituted by its transformation in terms of $\theta$ and $\mathit{\Gamma}$. Then one writes

$$\mathrm{d}R = \frac{\mathrm{d}R}{\mathrm{d}\theta}\mathrm{d}\theta + \frac{\mathrm{d}R}{\mathrm{d}\mathit{\Gamma}}\mathrm{d}\mathit{\Gamma} \tag{B.36}$$

and for the derivative we have going from $Z$ to $\mathit{\Gamma}$,

$$\frac{\partial C}{\partial Z} = \frac{\partial C}{\partial \mathit{\Gamma}}\frac{\partial \mathit{\Gamma}}{\partial Z} \,. \tag{B.37}$$

These are evaluated and substituted, and setting $\mathit{\Gamma} = 0$ the expressions given in Sect. 12.3.3.4 are obtained.

For example, for transforming for the MWA coordinates, repeated here,

$$\begin{aligned} R &= \cos\theta \cosh \mathit{\Gamma} \ , \\ Z &= \sin\theta \sinh \mathit{\Gamma} \ , \end{aligned} \tag{B.38}$$

this results in

$$\mathrm{d}R = -\sin\theta \cosh \mathit{\Gamma}\ \mathrm{d}\theta + \cos\theta \sinh \mathit{\Gamma}\ \mathrm{d}\mathit{\Gamma} \tag{B.39}$$

and

$$\frac{\partial C}{\partial Z} = \frac{\partial C}{\partial \Gamma}\frac{1}{\sin\theta \cosh \Gamma} \tag{B.40}$$

and substituting these and noting that $\mathit{\Gamma} = 0$, which eliminates the term in $\mathrm{d}\mathit{\Gamma}$, and for $\mathit{\Gamma} = 0$, $\cosh \mathit{\Gamma} = 1$, and noting that we must switch the integration limits which go from $0 \ldots 1$ along $R$ to $\pi/2 \ldots 0$ along $\theta$, we obtain

$$I = \frac{\pi}{2}\int_0^{\pi/2} \cos\theta \left.\frac{\partial C}{\partial \Gamma}\right|_{\Gamma=0} \mathrm{d}\theta \ . \tag{B.41}$$

Similarly for the other transformations.

## References

1. Morse PM, Feshbach H (1953) Methods in theoretical physics, 2 vols. McGraw-Hill, New York
2. Bland DR (1961) Solutions of Laplace's equation. Routledge and Kegan Paul, London
3. Boas ML (1966) Mathematical methods in the physical sciences. Wiley, New York, NY
4. Verbrugge MW, Baker DR (1992) Transient diffusion and migration to a disk electrode. J Phys Chem 96:4572–4580

# Appendix C
# Some Mathematical Proofs

## C.1 Consistency of the Sequential Method

As described in Chap. 5, the simulation of a first order homogeneous chemical reaction (*hcr*) coupled to diffusion such as the Reinert-Berg mechanism (5.11) we have the governing equation

$$\frac{\partial C}{\partial T} = \frac{\partial^2 C}{\partial X^2} - KC \tag{C.1}$$

and for the explicit point method it discretises to

$$C_i' = C_i + \lambda\,(C_{i-1} - 2C_i + C_{i+1}) - K\delta T C_i \tag{C.2}$$

containing, on the right-hand side, a term for diffusion and one for the *hcr*. The sequential method calculates first an intermediate new concentration, for which only diffusion has caused a change,

$$C_i^* = C_i + \lambda\,(C_{i-1} - 2C_i + C_{i+1}) \tag{C.3}$$

and then allows the *hcr* to act on this

$$C_i' = C_i^* - K\delta T C_i^* \tag{C.4}$$

giving, after combining the two,

$$C_i' = (1 - K\delta T)\,\{C_i + \lambda\,(C_{i-1} - 2C_i + C_{i+1})\}\ . \tag{C.5}$$

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8

The proof of the consistency of this procedure [1] goes as follows. Multiply (C.1) by $e^{KT}$:

$$\frac{\partial C}{\partial T}\,e^{KT} = e^{KT}\frac{\partial^2 C}{\partial X^2} - KCe^{KT}\ . \tag{C.6}$$

Now, since

$$\frac{\partial}{\partial T}\left(Ce^{KT}\right) = \frac{\partial C}{\partial T}\,e^{KT} + KCe^{KT}\ , \tag{C.7}$$

(C.6) can be written as

$$\frac{\partial}{\partial T}\left(Ce^{KT}\right) = e^{KT}\frac{\partial^2 C}{\partial X^2}\ . \tag{C.8}$$

The left-hand side can be approximated as the forward difference (at point $i$ along $X$)

$$\frac{\partial}{\partial T}\left(Ce^{KT}\right) \approx \frac{C_i' e^{K(T+\delta T)} - Ce^{KT}}{\delta T} \tag{C.9}$$

(recalling that $C_i$ is $C_i(T)$ and that $C_i' = C_i(T + \delta T)$) and combining this with (C.8) and discretising the second derivative as usual (Eq. (5.2), page 90), this leads to

$$C_i' e^{K(T+\delta T)} = C_i e^{KT} + \lambda e^{KT}\left(C_{i-1} - 2C_i + C_{i+1}\right) \tag{C.10}$$

which becomes, upon dividing throughout by $e^{K(T+\delta T)}$ and rearranging,

$$C_i' = e^{-K\delta T}\left\{C_i + \lambda\left(C_{i-1} - 2C_i + C_{i+1}\right)\right\}\ . \tag{C.11}$$

This converges to Eq. (C.5) as $K\delta T \rightarrow 0$. The sequential method is therefore mathematically consistent and this is the reason that it works rather well, within the limits of the approximation $e^{-K\delta T} \approx 1 - K\delta T$.

## C.2 The Feldberg Start for BDF

The simple start for BDF, as adopted by Mocak and Feldberg [2] coupled with the subsequent subtraction of half a time interval, as described in Chap. 4, Sect. 4.8.1.1, was found [3, 4] to be mathematically consistent. This is proved as follows.

The proof is given for a general *ode*. It also applies to a system of *ode*s and thus to the system of equations resulting from the discretisation of a *pde*. Let the equation to be solved be

$$u' = f(u), \quad u(0) = u_0 \tag{C.12}$$

with $f(u)$ being some unspecified function. This is solved using BDF (see Chap. 4). The contention is that after a number $n$ time steps each of length $h$ the time, which should be equal to $nh$, has in fact been shifted by a fraction $s$ of the time step length $h$, that is, by $s \times h$, and that this converges, after a number of steps, to $-0.5h$, justifying the Feldbergian correction.

The numerical solution yields a sequence of approximations to $u(t)$ at the times $t = h, 2h, \ldots$ denoted, respectively as $u_1, u_2, \ldots$. For convenience, we write $f_n$ to denote $f(u_n)$.

In what follows here, in order to be consistent with [5] (and normal computer science usage), BDF is described on a number of levels, rather than the number of points in time. Thus, the symbol $k$ now refers to levels, and is less by 1 than the $k$ used in other parts of this book. So, three-point BDF corresponds to $k = 2$, etc.

First, the expression to be solved is developed, for a simple case. We seek a solution to the *ode* (C.12), using 2-level BDF. The BDF expression at the $n$th step is

$$\frac{u_{n-2} - 4u_{n-1} + 3u_n}{2h} = f(u_n) \tag{C.13}$$

or

$$u_n = -\tfrac{1}{3}u_{n-2} + \tfrac{4}{3}u_{n-1} + \tfrac{2}{3}hf(u_n)\ . \tag{C.14}$$

For the very first step, $n = 1$, $u_{-1}$ is lacking, and with the simple start (see Chap. 4, Sect. 4.8.1) one simply substitutes $u_0$. The result is then

$$u_1 = u_0 + \tfrac{2}{3}hf_1 \tag{C.15}$$

(writing $f_1$ instead of $f(u_1)$). This is clearly equivalent to a BI step of length $\frac{2}{3}h$, and thus, one-third of an interval has been "lost", and $s_1 = -\frac{1}{3}$.

At step 2, using (C.13) and substituting for $u_1$ as from (C.15), the equation is

$$u_2 = u_0 + \tfrac{8}{9}hf_1 + \tfrac{2}{3}hf_2 \tag{C.16}$$

and we now have a total advance of $\frac{14}{9}h$, or shift $s_2 = -\frac{4}{9}$.

At step $n - 1$, let the expression be

$$u_{n-1} = u_0 + p_1hf_1 + p_2hf_2 + \cdots + p_{n-1}hf_{n-1} \tag{C.17}$$

and denote the total advance (given in units of $h$) in $t$ as

$$a_{n-1} = \sum_{i=1}^{n-1} p_i\ . \tag{C.18}$$

At the next step, we have a new series

$$u_n = u_0 + q_1hf_1 + q_2hf_2 + \cdots + q_nhf_n \tag{C.19}$$

(with $q_n = 2/3$) and a new sum of advances

$$a_n = \sum_{i=1}^{n} q_i \; . \tag{C.20}$$

From (C.13), we note that generation of $u_n$ is a linear combination of the $p$ and $q$ sequences, with the added last term, $\frac{2}{3}hf_n$. Thus, we have the recursive expression for the advances,

$$a_n = -\tfrac{1}{3}a_{n-2} + \tfrac{4}{3}a_{n-1} + \tfrac{2}{3} \; . \tag{C.21}$$

There is no advance before the first step, so that

$$a_i = 0, \quad i \le 0 \; . \tag{C.22}$$

By definition,

$$s_n = a_n - n \tag{C.23}$$

and this makes, given (C.22)

$$s_i = -i, \quad i \le 0 \; . \tag{C.24}$$

Substituting (C.23) into (C.21) finally yields the recursive expression for all $s_n$,

$$3s_n - 4s_{n-1} + s_{n-2} = 0 \tag{C.25}$$

with starting values for $s_n$ obtained from (C.24). The above treatment can be extended to higher-level forms, see below.

Up to this point, this has been presented in [3]. In that paper, this was then followed by computer calculations showing that for $k = 2, 3, 4$, the $s_n$ values converge to $-0.5$. There is, however, a mathematical proof [4].

The first few further recursive equations to be solved are

$$k{=}3: \quad 11s_n - 18s_{n-1} + 9s_{n-2} - 2s_{n-3} = 0 \tag{C.26}$$

$$k{=}4: \quad 25s_n - 48s_{n-1} + 36s_{n-2} - 16s_{n-3} + 3s_{n-4} = 0 \tag{C.27}$$

$$k{=}5: \quad 137s_n - 300s_{n-1} + 300s_{n-2} - 200s_{n-3} + 75s_{n-4} - 12s_{n-5} = 0 \tag{C.28}$$

$$k{=}6: \quad 147s_n - 360s_{n-1} + 450s_{n-2} - 400s_{n-3} + 225s_{n-4} - 72s_{n-5} + 10s_{n-6} = 0 \tag{C.29}$$

covering all the stable cases. The general form is

$$\alpha_0 s_n + \alpha_1 s_{n-1} + \alpha_2 s_{n-2} + \cdots + \alpha_k s_{n-k} = 0, \tag{C.30}$$

starting with values $s_0, s_{-1}$, etc., as given above in (C.24). Consider the generating function

$$\begin{aligned}
\alpha_0 \cdot \sum_{n=0}^{\infty} s_n t^n &= \sum_{n=0}^{k-1} \alpha_0 s_n t^n + \sum_{n=k}^{\infty} \alpha_0 s_n t^n \\
&= \sum_{n=0}^{k-1} \alpha_0 s_n t^n - \sum_{n=k}^{\infty} \sum_{i=1}^{k} \alpha_i s_{n-i} t^n \\
&= \sum_{n=0}^{k-1} \alpha_0 s_n t^n - \sum_{i=1}^{k} \alpha_i t^i \sum_{n=k-i}^{\infty} s_n t^n \\
&= \sum_{n=0}^{k-1} \alpha_0 s_n t^n - \sum_{i=1}^{k} \alpha_i t^i \sum_{n=0}^{\infty} s_n t^n + \sum_{i=1}^{k-1} \alpha_i t^i \sum_{n=0}^{k-i-1} s_n t^n \\
&= \sum_{i=0}^{k-1} \alpha_i t^i \sum_{n=0}^{k-i-1} s_n t^n - \sum_{i=1}^{k} \alpha_i t^i \sum_{n=0}^{\infty} s_n t^n
\end{aligned} \tag{C.31}$$

Thus

$$\sum_{n=0}^{\infty} s_n t^n = \frac{\sum_{i=0}^{k-1} \alpha_i t^i \sum_{n=0}^{k-i-1} s_n t^n}{\sum_{i=0}^{k} \alpha_i t^i} \tag{C.32}$$

We now want to find a single power series in $t$ to replace the expression on the right-hand side of Eq. (C.32), and its terms will then be equivalent with those of the left-hand side and yield the desired $s_n$. This is done by splitting Eq. (C.32) into partial fractions. An example follows here, namely the case $k = 2$ from Eq. (C.25) above. We now solve the recursive equations for $s_{n-1}$, for all $n \geq 2$. We have

$$\sum_{n=0}^{\infty} s_{n-1} t^n = \frac{\sum_{i=0}^{1} \alpha_i t^i \sum_{n=0}^{1-i} s_{n-1} t^n}{\sum_{i=0}^{2} \alpha_i t^i} = \frac{3s_{-1} - 4s_{-1}t + 3s_0 t}{3 - 4t + t^2}, \tag{C.33}$$

i.e.,

$$\sum_{n=0}^{\infty} s_{n-1}t^n = \frac{1}{2}\frac{-s_{-1}+3s_0}{1-t} + \frac{9}{2}\frac{s_{-1}-s_0}{3-t} \tag{C.34}$$

Hence, expanding as a power series in $t$,

$$\sum_{n=0}^{\infty} s_{n-1}t^n = \frac{1}{2}\sum_{n=0}^{\infty}\Big(-s_{-1}+3s_0+3(s_{-1}-s_0)\frac{1}{3^n}\Big)t^n \tag{C.35}$$

and $s_{n-1}$ is obtained by identifying the coefficients of the two sums. The solution to (C.25) is then

$$s_n = \frac{1}{2}\left(-s_{-1}+3s_0\right) + \frac{1}{2}\left(s_{-1}-s_0\right)\frac{1}{3^n}\,. \tag{C.36}$$

With increasing $n$, the second term on the right-hand side vanishes and together with Eq. (C.23) above (that is, $s_{-1} = 1$ and $s_0 = 0$); $s_n$ converges to $-1/2$. The same procedure applied to the higher $k$ values yields the following solutions:

$$k=3: \quad s_n = \frac{1}{6}\left(2s_{-2}-7s_{-1}+11s_0\right) + O(2.35^{-n}) \tag{C.37}$$

$$k=4: \quad s_n = \frac{1}{12}\left(-3s_{-3}+13s_{-2}-23s_{-1}+25s_0\right) + O(1.78^{-n}) \tag{C.38}$$

$$\begin{aligned} k=5: \quad s_n = {} & \frac{1}{60}\left(12s_{-4}-63s_{-3}+137s_{-2}-163s_{-1}+137s_0\right) \\ & +O(1.41^{-n}) \end{aligned} \tag{C.39}$$

$$\begin{aligned} k=6: \quad s_n = {} & \frac{1}{60}\left(-10s_{-5}+62s_{-4}-163s_{-3}+237s_{-2}-213s_{-1}\right. \\ & \left.+147s_0\right) + O(1.16^{-n}) \end{aligned} \tag{C.40}$$

The numbers 3, 2.35, 1.78, 1.41, 1.16 are the numerically smallest of the polynomial roots, all of which are shown in Table C.1, extending the range of $k$.

Now, $s_i = -i$ for $i \le 0$, so

$$\lim_{n\to\infty} s_n^{(k)} = -\frac{1}{2}\,. \tag{C.41}$$

for all five values of $k$, shown above, that is, $k \le 6$.

**Table C.1** Roots of the polynomials for $k = 1 \dots 8$

| k | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | | | | | | | |
| 2 | 1 | 3.00 | | | | | | |
| 3 | 1 | 2.35 | 2.35 | | | | | |
| 4 | 1 | 2.61 | 1.78 | 1.78 | | | | |
| 5 | 1 | 2.39 | 2.39 | 1.41 | 1.41 | | | |
| 6 | 1 | 2.46 | 2.11 | 2.11 | 1.16 | 1.16 | | |
| 7 | 1 | 2.35 | 2.35 | 1.85 | 1.85 | 0.98 | 0.98 | |
| 8 | 1 | 2.37 | 2.18 | 2.18 | 1.64 | 1.64 | 0.85 | 0.85 |

The general solution to Eq. (C.30) is of the form

$$s_n = \sum_{i=1}^{k} P_i(n)\lambda_i^{-n} \tag{C.42}$$

where the $\lambda_i$ coefficients are the complex roots of the polynomial

$$\alpha_0 + \alpha_1 t + \alpha_2 t^2 + \cdots + \alpha_k t^k = \alpha_0 \prod_{i=1}^{k} \left(1 - \frac{t}{\lambda_i}\right) \tag{C.43}$$

See [6, Chap. 4] for a proof of this.

The convergence for $k \leq 6$ is due to the fact that the roots of the five polynomials

$$3 - 4t + t^2 \tag{C.44}$$

$$11 - 18t + 9t^2 - 2t^3 \tag{C.45}$$

$$25 - 48t + 36t^2 - 16t^3 + 3t^4 \tag{C.46}$$

$$137 - 300t + 300t^2 - 200t^3 + 75t^4 - 12t^5 \tag{C.47}$$

$$147 - 360t + 450t^2 - 400t^3 + 225t^4 - 72t^5 + 10t^6 \tag{C.48}$$

all are numerically equal to or greater than 1.

The coefficients in the above polynomials are those for the BDF forms, given in Appendix A, as the last entry in each group in Table A.1.

Note that in the cases $k > 6$ there appears at least one polynomial root which is numerically smaller than 1. The form of the general solution (C.42) therefore implies that the values $s_n$ then do not converge to a finite value, as $n$ tends towards infinity. This is in accord with the known instability of BDF for $k > 6$ [7].

Table C.2 shows a few shifts $s_n$ for some k, and the convergence is clearly seen.

Convergence is slower for higher $k$, as is also implied by the values of the polynomial roots in Table C.1.

**Table C.2** $s_n$ values

| n | $k = 2$ | $k = 3$ | $k = 4$ |
|---|---|---|---|
| 1 | −0.333 | −0.455 | −0.520 |
| 2 | −0.444 | −0.562 | −0.598 |
| 3 | −0.481 | −0.548 | −0.520 |
| 4 | −0.494 | −0.519 | −0.470 |
| 5 | −0.498 | −0.503 | −0.474 |
| 6 | −0.499 | −0.499 | −0.494 |
| 7 | −0.500 | −0.499 | −0.504 |
| 8 | −0.500 | −0.499 | −0.504 |
| 9 | −0.500 | −0.500 | −0.501 |
| 10 | −0.500 | −0.500 | −0.499 |

## C.3 Similarity of the Exponential Expansion and Transformation Functions

In Chap. 7, two ways of implementing unequal intervals were described. These were the Seeber and Stefani [8] and Feldberg [9] approach, in which exponentially expanding boxes are placed along the $X$-axis (Eq. (7.16)), and the transformation method (Eq. (7.3)). Here it will be shown that they are approximately equivalent, and the relation between their respective expansion parameters will be given. The exponential expansion consists of a starting (first) box length of length $H_1$. Subsequent boxes are then defined such that box number $i$ has length $\beta^{i-1} H_1$. See Fig. C.1 for this and the points spacing. Points spacing uses the transformation relations

$$Y = \ln(1 + aX) \tag{C.49}$$

and the reverse

$$X = (e^Y - 1)/a \, . \tag{C.50}$$

We have at any point,

$$\begin{aligned} X_n &= H_1 + H_1\beta + H_1\beta^2 + \cdots + H_1\beta^{n-1} \\ &= H_1 \sum_{k=0}^{n-1} \beta^k \qquad\qquad (C.51) \\ &= H_1 \sum_{k=0}^{n-1} \exp(k \ln \beta) \; . \end{aligned}$$

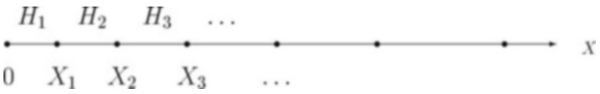


**Fig. C.1** Points and box unequal spacing

For large $n$, the sum approaches the integral and we have

$$X_n = H_1 \int_0^n \exp(k \ln \beta)\, \mathrm{d}k \tag{C.52}$$

and this is readily integrated to yield

$$X_n = \frac{H_1}{\ln(\beta)} \left(\exp(n \ln \beta) - 1\right) \ . \tag{C.53}$$

This is of the form of (C.50), thus establishing the equivalence. Also, we can read off the relation between the parameters, being

$$a \equiv \ln \beta / H_1 \tag{C.54}$$

and also since $Y_n = n\delta Y$,

$$\delta Y \equiv \ln \beta \ . \tag{C.55}$$

It is noted that the derivation is an approximation, resting on the approximate equality of the sum and integral, and holding better for larger $n$. However, it might be useful, given the information of the values of, say, $H_1$ and $\beta$ in a paper using the box method, to be able to translate it into the corresponding values of $a$ and $\delta Y$.

The second way of establishing the parameter relations focusses on three points (encompassing two boxes), $X_{n-1}, X_n, X_{n+1}$ obtained from (C.50). If they also obey the box expansion formula, then a definite expression for $\beta$ should be obtained from the ratio of the two box lengths,

$$\beta = \frac{X_{n+1} - X_n}{X_n - X_{n-1}} \tag{C.56}$$

which, substituting from (C.50), and reducing, becomes

$$\beta = \frac{e^{\delta Y} - 1}{1 - e^{-\delta Y}} \tag{C.57}$$

which reduces to

$$\beta = e^{\delta Y} \tag{C.58}$$

the same as (C.55). Then using

$$\delta Y = \ln(1 + aH_1) = \ln \beta \qquad \text{(C.59)}$$

and the approximation $\ln(1 + aH_1) \approx aH_1$ holding for $aH_1 << 1$ (which will usually be the case), relation (C.54). If $aH_1 << 1$ does not hold, then the more correct relation will be

$$\beta = 1 + aH_1 \; . \qquad \text{(C.60)}$$

## References

1. Ružić I, Britz D (1991) Consistency proof of the sequential algorithm for the digital simulation of systems involving first-order homogeneous kinetics. Acta Chem Scand 45:1087–1089
2. Mocak J, Feldberg SW (1994) The Richtmyer modification of the fully implicit finite difference algorithm for simulations of electrochemical problems. J Electroanal Chem 378:31–37
3. Britz D (2001) Consistency proof of Feldberg's simple BDF start in electrochemical digital simulation. J Electroanal Chem 515:1–7
4. Britz TJ, Britz D (2003) Mathematical proof of the consistency of Feldberg's simple BDF start in electrochemical digital simulations. J Electroanal Chem 546:123–125
5. Britz D (2003) Higher-order spatial discretisations in digital simulations. Algorithm for any multi-point first- or second derivative on an arbitrarily spaced grid. Electrochem Commun 5:195–198
6. Stanley RP (1997) Enumerative combinatorics, vol 1. Cambridge University Press, Cambridge, New York
7. Cryer CW (1972) On the instability of high order backward-difference multistep methods. BIT Numer Math 12:17–25
8. Seeber R, Stefani S (1981) Explicit finite difference method in simulating electrode processes. Anal Chem 53:1011–1016
9. Feldberg SW (1981) Optimization of explicit finite-difference simulation of electrochemical phenomena utilizing an exponentially expanded space grid. Refinement of the Joslin-Pletcher algorithm. J Electroanal Chem 127:1–10

# Appendix D
# Finding $\Gamma_{max}$

We proceed from $(R, Z)$ space, where it is easy to define an envelope that encloses the diffusion field to a good approximation. This is the length $L$, defined, for times $T$ other than very large (see below), by

$$L = 6\sqrt{T} \ . \tag{D.1}$$

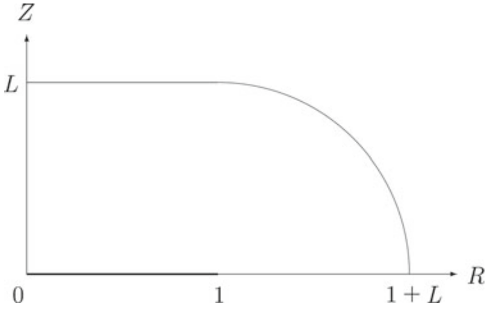
For the Cottrell system, this is the distance, beyond which changes greater than $10^{-4}$ relative to the bulk concentration are no longer observed. Figure D.1 shows this line. In the range $0 \le R \le 1$, it is simply the line

$$Z = L \ , \tag{D.2}$$

and for $1 < R \le 1 + L$, it is a quarter-circle defined by

$$Z^2 + (R-1)^2 = L^2 \ . \tag{D.3}$$

**Fig. D.1** Diffusion limit line



D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8

As will be seen, the line generates a corresponding line of $\Gamma$ values by inversion of a given transformation; and one chooses the maximum value, to be on the safe side. That value lies at the point $(R,Z) = (1+L,\ 0)$, as will now be shown.

To find the maximum $\Gamma$ value, we first find the maximum value along the straight line segment, defined by (D.2). This equation is substituted in the equation for $Z$ in the given transformation. For example, for the MWA transformation, this leads to

$$\sinh(\Gamma) = \frac{L}{\sin(\theta)}\ . \tag{D.4}$$

This is maximum for a minimum $\theta$, meaning the point $(1, L)$, at which the line joins the quarter-circle. It remains to search that arc for $\Gamma$ values. Equation (D.3) is substituted by the transformed expressions. The **VB** limit is obtained by

$$\begin{aligned}\Gamma'_m &= \operatorname{arccosh}(1+L)\\ &= \ln\left(1+L+\sqrt{(1+L)^2-1}\right)\ ,\end{aligned} \tag{D.5}$$

and then

$$\Gamma_{max(\mathrm{VB})} = \frac{\Gamma'_m}{1+\Gamma'_m}\ . \tag{D.6}$$

The above implies that $\Gamma_{max}$ generally increases indefinitely with $T$. However, this ignores the fact of a steady state at the UMDE at long times, so for long times, $\Gamma_{max}$ might be an overestimate. The choice of $L$ in (D.1) is made on the basis of the Cottrell experiment at a planar shrouded electrode, and defines the point where the concentration deviates from that in the bulk by no more than some small number, such as $10^{-4}$ (which leads to the factor 6). At the steady state for the UMDE, we have the analytical solution for the concentration profile [1, 2]:

$$C = 1 - \frac{2}{\pi}\arcsin\left(\frac{2}{\sqrt{Z^2+(1+R)^2}+\sqrt{Z^2+(1-R)^2}}\right) \tag{D.7}$$

and if we substitute $C = 0.9999$ in this equation, we obtain a curve very close to a quarter-circle, with a maximum $\Gamma$ value corresponding to the point $(R,Z) = \frac{2\times10^4}{\pi}$. This means that for roughly $T > 1000$, little is to be gained by not including the whole $\Gamma$ range.

## References

1. Crank J, Furzeland RM (1977) The treatment of boundary singularities in axially symmetric problems containing discs. J Inst Maths Appl 20:355–370
2. Saito Y (1968) A theoretical study on the diffusion current at the stationary electrodes of circular and narrow band types. Rev Polarog (Jpn) 15:177–187

# Appendix E
# Procedure and Program Examples

Here some modules, procedures and whole programs are described, that may be useful to the reader, as they have been to the authors. They are all in Fortran 90/95 and invoke a generally useful module, that will be used in all procedures and programs in the examples, and another module useful for programs using a Rosenbrock variant. The source texts (except for the two modules) are not all reproduced here, but can be downloaded from the web site http://extras.springer.com.

## E.1 Example Modules

### *E.1.1 Module* `STUFF`

In this module, the two `kind`s of `real` values, respectively single precision (6 decimals or a little better) and what used to be called double precision (here defined as 14 decimals or a little better) are given the names `sgl` and `dbl`, and the constant `pi` is made a parameter. As well, the (`real`) parameter `small` is set up. This is useful in those cases where something is tested for being very small.

```
module STUFF
! General-purpose module.
  implicit none
  integer, parameter         :: sgl=selected_real_kind(6),&
                                dbl=selected_real_kind(14)
  real(kind=dbl), parameter  :: small = 1.0E-08
  real(kind=dbl), parameter  :: pi = 3.14159265358979
end module STUFF
```

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs in Electrochemistry, DOI 10.1007/978-3-319-30292-8

### E.1.2 Module ROSTUFF

Rosenbrock methods require a set of constants, and as these are needed in several subroutines in a given program, it is convenient to gather them in a module. Experiments have shown that the two Rosenbrock variants, ROS2 and ROWDA3 (see Chap. 9), are about the best in the present context, responding without oscillations. Only these two have been included in the module. Readers wanting the constants for the two other methods RO2 and ROS3P can find them in Appendix A, in the same unified notation as these two preferred variants.

```
module ROSTUFF
! Module for the Rosenbrock coefficients.
  use STUFF;   implicit none
  real(kind=dbl) :: gamma, gamma1, gamma2, gamma3, &
                    alpha1,  alpha2,  alpha3,      &
                    a21, a31, a32,                 &
                    c21, c31, c32, m1, m2, m3
CONTAINS
  subroutine ROCOEFFS (order)
  ! Sets the Rosenbrock coeffs for orders 2 or 3,
  ! accessing the module ROSTUFF where they are.
  ! Order 2 is ROS2, order 3 is ROWDA3.
  use STUFF;  implicit none
  integer :: order

  gamma1 = 0 ! Zero defaults
  gamma2 = 0
  gamma3 = 0
  alpha1 = 0
  alpha2 = 0
  alpha3 = 0
  a21 = 0
  a31 = 0
  a32 = 0
  c21 = 0
  c31 = 0
  c32 = 0
  m1 = 0
  select case (order)
    case (2)
      gamma = 1.707106781186547_dbl
      gamma1 = 0
      gamma2 = - gamma
      alpha2 = 1
      a21 = 0.5857864376269050_dbl
      c21 = - 1.171572875253810_dbl
      m1  = 0.8786796564403575_dbl
      m2  = 0.2928932188134525_dbl
    case (3)
      gamma = 0.435866521508459_dbl
      gamma1 = gamma
      gamma2 = 0.6044552840655588_dbl
      gamma3 = 6.3797887993448800_dbl
      alpha2 = 0.7_dbl
      alpha3 = 0.7_dbl
      a21 = 1.605996252195329_dbl
      a31 = a21
      a32 = 0
      c21 = 0.8874044410657823_dbl
      c31 = 23.98747971635035_dbl
      c32 = 5.263722371562130_dbl
      m1  = 2.236727045296589_dbl
      m2  = 2.250067730969645_dbl
      m3  = -0.209251404439032_dbl
    end select
  end subroutine ROCOEFFS
end module ROSTUFF
```

## E.2 Procedures

A few functions and subroutines that recur in simulation programs, are presented here, not in alphabetic order.

### *E.2.1 File Names Routine*

The routine FILSPC conveniently reads in a file name and opens the file. For personal historical reasons (DB), logical unit numbers (LUNs) either equal to or ending with 3 are input files to be read from, while those ending with 4 are output files, to be written into. Thus there are a possible ten of each of these two types. The files are all ASCII.

### *E.2.2 The Error Functions*

In test programs, where the numerical solution is compared with the analytical solution, the latter often involves the error function erf or the complementary error function erfc. The latter could be obtained simply by subtracting erf from unity but a better approximation is obtained by the direct algorithm. The two routines, `ERF` and `ERFC`, were given to one of us (DB) by a colleague, who probably obtained them from an IBM collection. They have been adapted to Fortran 90/95 by the authors, and use the module `STUFF`. The comments in capitals are the original comments.

### *E.2.3 Current Approximations*

The current value is obtained from the concentrations at any time as the dimensionless quantity $G$, the gradient $\mathrm{d}C/\mathrm{d}X$ at $X = 0$ as an $n$-point forward difference (see Appendix A, or Sect. 3.4). For concentrations represented at equal intervals, this is conveniently computed by the function `G0FUNC`.

`G0FUNC` can be inverted to calculate $C_0$, given $G$; this is useful in the formulation of derivative boundary conditions. For example, to take a simple case, in chronopotentiometry, one has constant $G$ and computes $C_0$ from that and the concentration profile. The function `C0FUNC` does this job. Both $G$ and $H$ are needed, but they appear as the product $GH$, which is passed to the function.

Both these functions call on `G0BETA` for the coefficients. It is also useful in other contexts, such as the setting up of boundary value calculations for coupled systems.

For unequal intervals, there are equivalent functions `G0FORN` and `C0FORN`, both using the Fornberg algorithm [1]. These two need the $X$ array to operate.

### *E.2.4 Matrix Inversion*

In some programs such as `CV_CAT` and the migration programs in Chap. 13, matrix inversion is needed. This is best done by using LU decomposition, as described in Press et al. [2]. The subroutine `MATINV` does this, inverting the passed matrix in place; that is, the matrix is returned inverted. If one does not want it altered, the other routine `MAT_INV` returns the inverse in a new matrix, leaving the original one unchanged. In fact, `MATINV` calls `MAT_INV`. Both assume a square matrix, of the exact size given, so the best way to call them is by using a section of that size, for example

```
call MATINV (mat(1:N,1:N), N)
```

For matrices up to 4×4, inversion is done "manually" for speed. The two subroutines `DEC` and `SOL` are the usual LU decomposition routines, of which there are a number, freely available.

### *E.2.5* `MINMAX`

This subroutine is useful in linear sweep simulations, for calculating peak (or trough) current values and where they occur, from a trio of currents, in which the second is either larger, or smaller, than the other two. A parabola is fitted to the three points,

$$y = a_0 + a_1 x + a_2 x^2 \tag{E.1}$$

and the parabola's maximum or minimum is computed, as well as its position. The positions are assumed (by the subroutine) to be, respectively, at −1, 0 and +1, so that it is up to the calling program unit to scale the minmax position. Needless to say, $X$ must be equally spaced.

For arbitrarily spaced intervals, we require procedures for first and second derivatives, and some other subroutines, as well as routines for setting the sequence of points according to the expansion desired.

### *E.2.6* `EE_FAC`

There is a need in some situations to generate a series of points, with the intervals between them exponentially expanding from a base value. This can be both in time or in space. It is often most convenient (see Chap. 7) to start with the base interval, $H_1$ (which is also the first value $X_1$ after zero), the last value, $L$, and the number of points $N$ to be generated in that range. The expansion factor $\gamma$ then makes

$$h_i = \gamma h_{i-1} \tag{E.2}$$

or, as in Chap. 7,

$$x_i = \frac{\gamma^i - 1}{\gamma - 1}\ . \tag{E.3}$$

The task is then to find the $\alpha$ that fits the requirements. This can easily be done using a binary search, and the function `EE_FAC` is provided for this.

### *E.2.7* `DAMPED_EXPANSION`

This routine computes the base $\gamma_1$ for a damped exponentially expanding sequence, where $\gamma$ is damped for increasing $X$, as described in Chap. 7, page 130. It then computes and returns, besides this $\gamma_1$ value, the position sequence $X$.

### E.2.8 SV_FAC

There is another sequence of point positions of potential use, the S&V sequence, discussed in Chap. 7, page 129. This sequence is described by the recursion

$$h_i = h_{i-1}(1 + \alpha H_{i-1}/H_1) \; . \tag{E.4}$$

As with the exponentially expanding sequence, we start with a first interval $H_1$, decide on a furthest point $L$ and how many points there should be in the sequence. The function SV_FAC computes the expansion factor, here called $\alpha$. The task is then to find the $\alpha$ that fits the requirements. See page 129 for why this might be considered interesting.

### E.2.9 Gradient Routine FORNBERG and FORN

In many 2D programs, there are derivative boundary conditions requiring gradient approximations in the form of the weighting coefficients. These are best obtained by the routine FORNBERG which, unlike the simpler FORN, returns the derivative as well as the weighting coefficients, which then can be applied in the discretisation expressions. FORN only returns the coefficients. FORNBERG naturally requires the values at the points in order to operate.

### E.2.10 Current Integration on an Unequally Gridded Surface

In the case of the ultramicroelectrodes such as the disk electrode, it is necessary to integrate over the surface, and sometimes there will be unequally spaced points along the surface, as for example, in direct discretisation on an unequal grid in the example program UME_DIRECT. The routine U_TRAP does a trapezium integration, usually sufficiently accurate, given the inaccuracies in the gradients. It integrates local current densities, precalculated by using the routine G0FORN. If better accuracy is wanted, the Simpson-like U_SIMP can be used.

### E.2.11 Reference Fluxes and Errors

For 2D simulations testing new algorithms, and where there are no proper analytical solutions it is convenient to have reliable reference flux values to compare with. We have provided some of these: UMDE_REF_FLUX for the ultramicrodisk electrode, BAND_REF_FLUX for the band, and CYL_REF_FLUX for the cylinder electrode.

These are described in detail in [3–5]. Two of them are accompanied by subroutines that compute the error in simulated flux values, calling on these routines for comparison, `UMDE_ERROR` and `BAND_ERROR`. We have not had occasion to produce a routine for errors in the current at a cylinder but extension of the other routines to this is evident.

A recent paper by Bieniasz [6] presented a method for computing machine-precision current values for the ultramicroband electrode, using MATLAB, but not the values themselves. The method requires heavy computing, and a follow-up paper provides equally accurate polynomial approximations in the form of coefficients [7].

### *E.2.12 JCOBI*

In Chap. 9, the method orthogonal collocation is described. It makes use of certain Jacobi polynomials, whose roots become the node points $X$, at which concentrations are defined. The subroutine `JCOBI` is an adaptation of the subroutine reproduced in the book by Villadsen and Michelsen [8], converting it to Fortran 90 and making use of the module `STUFF`. There is a number of options in the subroutine. In using the subroutine to generate Tables A.5, A.6, and A.7, the recommendations of Whiting and Carr [9] were followed, setting both parameters $\alpha$ and $\beta$ to zero, and not including the boundary points indexed zero and $N + 1$. See the book by Villadsen and Michelsen for the details.

### *E.2.13 I1I2*

The small routine `I1I2`, although performing a very simple job, has been useful to the authors in simulations. When using multi-point derivative approximations on a window of $m$ points somewhere along the whole stretch indexed with $1 \ldots N$, it is clearly necessary to know the window start and end indices, especially if it is close to one of the ends of the whole stretch, where the approximation will be asymmetrical. This entails two `IF` statements and the routine relieves the programmer of the tedium of writing this at every occurrence in the program.

## E.3 Example Programs

### *E.3.1 Program `COTT_EX`*

This program simulates the Cottrell experiment, as discussed in Chap. 5. The output, upon running this for `NT`=100 and `lambda`=0.45 and three-point current approximation, is

```
   iT      T        G   log10(error)
    1    0.010    4.025    -0.54
    2    0.020    4.100    -1.56
    4    0.040    2.876    -1.71
    8    0.080    2.005    -2.28
   16    0.160    1.411    -3.16
   32    0.320    0.998    -3.26
   64    0.640    0.706    -3.34
  100    1.000    0.564    -3.48
```

Ordinarily, there would be some header information, echoing the input data and data derived from it. Also, the prompts produced by the program have been deleted.

### *E.3.2 Program* CHRONO_EX

This program is much the same as COTT_EX, but with the derivative boundary condition, see Chap. 5 for background information. Note that $C_0$ is calculated at the start, so that it conforms to the rest of the initial concentration profile, so as to satisfy the boundary condition. This calculation is repeated after every run of the innermost loop in which all concentrations $C_1, \dots C_N$ are recalculated. In this way, the old concentrations always include the proper $C_0$ value. This must always be the case, no matter what the boundary conditions are.

This produces, again for NT=100 and lambda=0.45, the (trimmed) output

```
   iT      T       C(0)  C(0)(analyt) log10(err)
    1    0.010    0.887    0.900          -1.88
    2    0.020    0.849    0.859          -2.00
    4    0.040    0.792    0.800          -2.10
    8    0.080    0.711    0.717          -2.22
   16    0.160    0.596    0.600          -2.37
   32    0.320    0.431    0.434          -2.52
   64    0.640    0.198    0.200          -2.67
  100    1.000   -0.002    0.000          -2.77
```

### *E.3.3 Program* CV_EX

This is a simple simulation of a CV experiment, using the explicit method EX, and assuming a quasireversible reaction,

$$A + e^- \rightleftharpoons B \tag{E.5}$$

with dimensionless heterogeneous rate constant $K_0$, as defined in (2.28). If this constant is input as $K_0 > 1000$, the system is assumed reversible and the Nernst boundary condition is applied. See Sect. 5.5 for the details of how to apply the boundary conditions. Here, an important point needs to be made. When carrying out a step forward in time using method EX, the old $C$-arrays are used to explicitly generate the new arrays at the new time. Again, the old $C$-arrays must include the boundary values corresponding to the arrays, so that the boundary conditions hold.

The program draws on several subroutines and functions, already described, such as `G0FUNC`, `G0BETA` and `MINMAX` to calculate peak and trough currents and at what potentials they occur.

This produces the following example output in a particular run:

```
CV_EX
Lambda   =     0.450
H        =     0.149
nT per p =   100
N        =   278
K0       =  1001.000
 1190 points written into plot file.
Top current and -p =  0.4462 -1.1083
Bot current and -p = -0.3334  1.1326
```

Being an explicit method, the program uses 278 points in space, even though $H$ is rather large at 0.149. The peak current is surprisingly accurate (to all four decimals) but the peak potential is not (it should be $-1.1090$ for the $K_0$ value set here at 1001 to force the Nernst boundary condition). Generally, one finds that the peak potential is a more sensitive indicator of how well a given simulation method works.

### *E.3.4 Program `COTT_CN`*

This program does the same work as the earlier one, `COTT_EX`, but uses Crank–Nicolson (with equal intervals). It also includes the choice of $M$ Pearson substeps within the first step, to damp the oscillations, as discussed in Chap. 8, Sect. 8.5.1.

Using similar parameters as for the earlier program, but making use of CN's stability with respect to $\lambda$, that parameter is set to 3, giving more points in the $X$-range. Here is a sample output from a run (again omitting the dialog part):

```
CN Cottrell simulation.
NT     =   100
Lambda =     3.00
N      =   104 pts along X.
10 Pearson substeps within first step.

   iT       T       Gsim    log10(err)
    2     0.020     3.878     -1.56
```

```
      4       0.040       2.802     -2.16
      8       0.080       1.992     -2.87
     16       0.160       1.410     -3.84
     32       0.320       0.997     -4.16
     64       0.640       0.705     -4.15
    100       1.000       0.564     -4.27
```

Note the improved accuracy, using 10 Pearson steps, despite using only 100 steps in time. The explicit program gave rise to errors in the third decimal at the end of the simulation, but here they lie at around the fourth or better.

### *E.3.5 Program CHRONO_CN*

A chronopotentiometry program, using CN, is shown here, again, as with the above COTT_CN, with equal intervals. The two programs are in fact very similar, differing only in the boundary conditions in the CN routine, and the initialisation. As before, old known concentrations always include a conforming $C_0$.

Note also that although the Pearson option has been included, it is not really needed here. CN does not oscillate with the constant current start.

A sample output follows, again using $\lambda = 3$ as for Cottrell above.

```
CN chronopot. simulation.
NT     =   100
Lambda =     3.00
N      =   104 pts along X.
 1 Pearson substeps within first step.

   iT        T         C(0)      log10(err)
     2      0.020       0.859     -3.65
     4      0.040       0.800     -3.45
     8      0.080       0.717     -3.46
    16      0.160       0.600     -3.66
    32      0.320       0.434     -3.85
    64      0.640       0.200     -4.03
   100      1.000       0.000     -4.14
```

As with Cottrell, note the accuracy compared with CHRONO_EX. Note also that a single "Pearson" step has been used, meaning no subdivision of the first step.

### *E.3.6 Program CHRONO_CN_HERM*

Another version of chronopotentiometry using CN, but here using Hermitian gradient approximations of the 2(2) and 2(3) form, see Chap. 3, page 46.

## E.3.7 Program LSV_CN

The following shows a CN program with unequal intervals, simulating a single LSV sweep for a reversible system. Apart from writing out the $(G, p)$ results for plotting, the program also detects the peak value and the potential at which it peaks. The boundary condition part is described in Chap. 6, page 110.

An example output, apart from the data file for plotting, was

```
LSV_CN_UN:
nT     =   100 per p-unit
pstart =    12.000
pstop  =   -12.000
X(1)   =     0.00100
N      =    50
Xlim   =    29.394
G, using     5 points.
gamma  =     1.18811 (found by iteration)

G-peak of    0.4471 found at p =   -1.1092
```

This program uses exponentially expanding intervals in $X$, and we started here with a first interval of 0.001 and demanded 50 points across the diffusion space. This set the $\gamma$ value as seen above. The peak current is a little off (it should be 0.4463) but the peak potential is quite good (the exact value is $-1.1090$).

## E.3.8 Program COTT_EXTRAP

This is an example of a Cottrell simulation using second-order extrapolation based on the BI (Laasonen) method and unequal intervals. Three-point spatial discretisation is used here.

## E.3.9 A Nonlinear System: Programs for the Birk/Perone Reaction

One of the problems mentioned in Chap. 8 is that of second-order homogeneous chemical reactions, which give rise to nonlinear terms in the transport equations. One such system is the Birk and Perone reaction [10, 11], in which a light flash produces an electroactive substance in solution, which decays with a second-order reaction while it is electrolysed. If CN is used to simulate this, the term in $C_i^2$ can be linearised to a good, second-order approximation. If one does not choose or is prevented from linearisation, a Newton approach, as described in that chapter, must be used. The three programs below are examples of both approaches as well as the Rosenbrock method. These programs not only illustrate the handling of nonlinear equations, but also of the time-dependent outer boundary values, due to the decaying substrate. There is an analytical solution for this value and its time derivative, and both are made use of in the programs. The system simulated is a Cottrell experiment (potential jump) on the decaying substance using CN and Rosenbrock. The first program, BP_LIN, uses linearisation, the second, BP_NONLIN, the Newton method. The two programs produce almost, but not quite the same, results, the Newton version being slightly more accurate. Both programs make automatic use of the Pearson start, by subdividing the first step in that number of substeps that gives a unity value to $\delta T/H_1^2$. In this way, negative concentrations are avoided.

A third program, BP_ROS, uses two Rosenbrock methods, discussed in Chaps. 4 and 9, which handle nonlinear equations without iteration or linearisation, and thus suggest themselves for the solution of nonlinear systems. The program provides the choice of two Rosenbrock methods. It works about as well as the above two and illustrates the Rosenbrock approach. The program also makes use, besides the usual STUFF, of the special Rosenbrock module ROSTUFF, see above. The module contains the subroutine ROCOEFFS, which sets the Rosenbrock constants. Only the Rosenbrock variants ROS2 and ROWDA3 are allowed for in the program. The inconsistency at $T = 0$, mentioned in Sect. 9.4 page 203, is overcome by the simple trick of setting $C_0$ to zero initially.

### *E.3.10 EC Reaction, Cyclic Voltammetry:* **CV_EC**

The program CV_EC uses CN to simulate the rather simple EC reaction as described in Chap. 8. There are no complications here. The output is in the form of a data file for plotting. It makes use of the simple subroutine MATINV to solve the small 2 × 2 system of equations for the boundary values. This could have been done directly but when we already have this routine, why not use it? Note that the $a'$ coefficients are different for the two species but are constant throughout the simulation and are therefore precalculated. This system is not coupled, so that the (scalar) Thomas algorithm can be used. Figure E.1 shows the result of some runs of this program. The fat curve is for $K = 0$, that is, plain reversible CV without a chemical reaction, and the numbers marked on the curves show the $K$ values input to the program. As expected, as $K$ increases, the negative-going peak (at the top) shifts in the positive direction and the trough (on the reverse sweep) becomes smaller, to disappear entirely for large $K$.

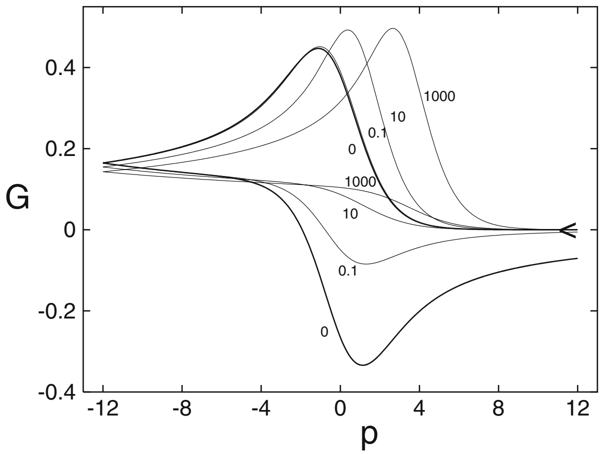


**Fig. E.1** CV of the EC reaction, *K* values as marked

### *E.3.11 CV of the EC' Reaction: Program* `CV_CAT`

The coupled system arising from the EC' or catalytic system, described in Chap. 8, was programmed using CN and the block-Thomas method. That is, the two concentration vectors were gathered into a vector of two-element vectors, and the usual coefficients in the discrete system of equations become a number of coefficient matrices. For this system, the **A** matrices are constant over the whole simulation and could be precomputed. In fact, one finds that one does not need these but rather the inverse of the $\mathbf{A}'$ matrices that are the result of reducing the system to two variables (vectors) for each row; and it is these that are precomputed and stored. The program outputs a data file for plotting. The routine `MATINV` is used more extensively here. Figure E.2 shows a family of CV curves, the fat curve again being that for the plain reversible case, and the others with rate constants $K$ as marked. As $K$ increases, the curve becomes increasingly sigmoid-shaped, with a plateau of height equal to $\sqrt{K}$.

### *E.3.12 LSV Simulation with iR Drop and Capacitance: Program* `LSV4IRC`

The program `LSV4IRC` is a simulation of a reversible reaction with input values of $\rho$ (dimensionless uncompensated resistance) and $\gamma_c$ (dimensionless double layer capacity). Unequal intervals are used, with asymmetric four-point second spatial derivatives, and second-order extrapolation in the time direction. The nonlinear set of six equations for the boundary values is solved by Newton–Raphson iteration. Some results are seen in Chap. 11.

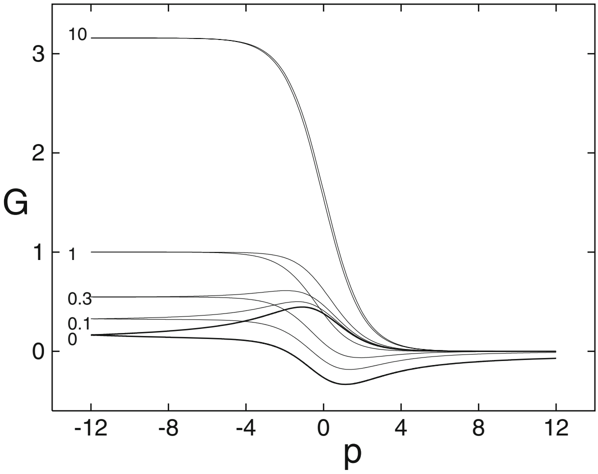


**Fig. E.2** CV of the catalytic (EC$'$) reaction, $K$ values as marked

### *E.3.13 Program* `UMDE_DIRECT`

This program simulates a Cottrellian potential jump at a UMDE, using three-point BDF for the time integration (starting it with a single BI step), and $m$-point asymmetric spatial discretisation ($m$ variable), on an $(R, Z)$ grid with exponentially expanding intervals expanding upwards in $Z$, and away from the electrode edge at $R = 1$ in both directions. This is discussed in Chap. 12, Sect. 12.3.2. The sparse solver `MA28` is used.

### *E.3.14 Program* `UMDE_VB`

This is a version of the disk simulator but using a grid in the conformal space given by the Verbrugge/Baker transformation, as discussed in Chap. 12, page 279.

### *E.3.15 Program* `UMDE_ARRAY`

This program is patterned after `UMDE_DIRECT`, with the difference that the $R$ axis is limited by the diffusion domain radius $R_d$, at some given multiple of the disk radius, as discussed in Chap. 12, Sect. 12.5. Effectively, at this radius, there is a cylindrical wall around the electrode, across which there is no flux, so we have a zero gradient boundary condition there.

The program simulates an LSV experiment for a reversible electrode reaction. The first output file contains dimensionless potential/current values, and the second output file contains the final concentration field. The program was used to calculate the LSVs shown in Fig. 12.15 of Sect. 12.5. A typical input file to start the program is

```
out1.dat
cprofile.dat
18, -18, 50       #Pstart, Pstop, nT
0.207957, 2.82    #pscan, Rd
100, 30, 70       #nZ, nA,  nR
5E-5, 5E-5, 2.82  #dZ1, dR1
5                 #m
20                #fa
```

The number of points of the finite difference approximations to the derivatives of the *pde* and boundary conditions is given by the parameter `m`. Here, `m`=5 implies the use of symmetrical $u_3''(5)$ and $u_3'(5)$ formulae for inner grid points and asymmetric five-point formulae close to and at the boundaries. The parameter `pscan` is the dimensionless scan rate for a UMDE, as defined by Eq. (12.28). The factor `fa` is for memory allocation of the sparse matrix solver `MA28`. The input list given above was used to simulate the voltammogram for array $A_4$ in Fig. 12.15.

### E.3.16 Programs LIQU_JUNC, RPC, CURDE

The three example programs for migration as described in detail in Chap. 13 are included in the collection.

### E.3.17 Program CHANNEL_BAND

This program does a steady state simulation of the current at a narrow band electrode at the bottom of a channel with laminar flow of electrolyte through it, as discussed in Chap. 14, page 376. It is done as a march along $X$, the direction of flow.

## References

1. Fornberg B (1988) Generation of finite difference formulas on arbitrarily spaced grids. Math Comput 51:699–706
2. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1986) Numerical recipes in Fortran. The art of scientific computing, 1st edn. Cambridge University Press, Cambridge

3. Britz D, Poulsen K, Strutwolf J (2004) Reference values of the diffusion-limited current at a disk electrode. Electrochim Acta 50:107–113. See Erratum, ibid. 53:8101 (2008)
4. Britz D, Poulsen K, Strutwolf J (2005) Reference values of the diffusion-limited chronoamperometric current at a microband electrode. Electrochim Acta 51:333–339. See Erratum, ibid. 53:7805 (2008)
5. Britz D, Østerby O, Strutwolf J (2010) Reference values of the chronoamperometric response at cylindrical and capped cylindrical electrodes. Electrochim Acta 55:5629–5635
6. Bieniasz LK (2015) Theory of potential step chronoamperometry at a microband electrode: complete explicit semi-analytical formulae for the Faradaic current density and the Faradaic current. Electrochim Acta 178:25–33
7. Bieniasz LK (2016) Highly accurate, inexpensive procedures for computing theoretical chronoamperometric currents at single straight electrode edges and at single microband electrodes. J Electroanal Chem 760:71–79
8. Villadsen J, Michelsen ML (1978) Solution of differential equation models by polynomial approximation. Prentice-Hall, New Jersey
9. Whiting LF, Carr PW (1977) A simple, fast numerical method for the solution of a wide variety of electrochemical diffusion problems. J Electroanal Chem 81:1–20
10. Britz D, Kastening B (1974) On the electrochemical observation of a second-order decay of radicals generated by flash photolysis or pulse radiolysis. J Electroanal Chem 56:73–90
11. Birk JR, Perone SP (1968) Electrochemical studies of rapid photolytic processes. A theoretical and experimental evaluation of potentiostatic analysis in flash photolysed solutions. Anal Chem 40:496–500

# Index

D. Britz, J. Strutwolf, *Digital Simulation in Electrochemistry*, Monographs
in Electrochemistry, DOI 10.1007/978-3-319-30292-8