Computational Methods in Applied Sciences

Jorge Magalhães-Mendes
David Greiner *Editors*

Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management

ECCOMAS
European Community on Computational Methods in Applied Sciences

Springer

Computational Methods in Applied Sciences

Volume 39

Series editor

E. Oñate
CIMNE
Edificio C-1, Campus Norte UPC
Gran Capitán, s/n
08034 Barcelona, Spain
onate@cimne.upc.edu

More information about this series at http://www.springer.com/series/6899

Jorge Magalhães-Mendes · David Greiner
Editors

Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management

Springer

Editors
Jorge Magalhães-Mendes
Departamento de Engenharia Civil
ISEP, Instituto Politecnico Porto
Porto
Portugal

David Greiner
Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numericas en Ingenieria—SIANI
Universidad de Las Palmas de Gran Canaria
Las Palmas de Gran Canaria
Spain

ISSN 1871-3033
Computational Methods in Applied Sciences
ISBN 978-3-319-20405-5 ISBN 978-3-319-20406-2 (eBook)
DOI 10.1007/978-3-319-20406-2

Library of Congress Control Number: 2015943032

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Preface

It's best to have failure happen early in life.
It wakes up the Phoenix bird in you so you rise from the ashes.

Anne Baxter

We all hope for breakthrough rebirth moments.

Dane Cook

Evolutionary Algorithms and Metaheuristics are used extensively and with growing interest in wide areas of applied sciences and engineering for solving real application problems of interest in industry and society.

Particularly, in recent years, different MiniSymposium/Special Thematic Sessions focused on the area of "Civil Engineering and Construction Management" promoted by the editors of this book have attracted the attention of the scientific community in different European Community on Computational Methods in Applied Sciences (ECCOMAS) related conferences. Among them we should mention:

- "Applications in Structural and Civil Engineering Optimum Design", MiniSymposium at the 10th International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems EUROGEN 2013, ECCOMAS Thematic Conference, Las Palmas de Gran Canaria, Spain, October 2013 (organized jointly also with Rajan Filomeno Coelho, Universite Libre de Bruxelles, Belgium).
- "Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management", MiniSymposium at the 11th World Congress on Computational Mechanics WCCM—5th European Conference on Computational Mechanics ECCM, IACM—ECCOMAS, Barcelona, Spain, July 2014.
- "Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management", Special Thematic Session at the Congress on Numerical Methods in Engineering organized by Sociedad Española de Métodos

Numéricos en Ingeniería SEMNI and Associação Portuguesa de Mecânica Teórica, Aplicada e Computacional APMTAC (Spanish and Portuguese Societies, respectively, integrated in ECCOMAS), Lisboa, Portugal, June–July 2015.
- "Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management", MiniSymposium at the European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS 2016, Crete Island, Greece, June 2016.

Among the participants of these meetings, some selected contributions which constitute modified, extended, and improved versions of research have been collected in this volume. It presents up-to-date material on the state of the art in Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management from European contributors, being mainly oriented for researchers and postgraduate students who are familiar with the fundamentals and wish to study or to advance the state of the art on the field, although practicing engineers could benefit as well from it as there is a recent tendency of including evolutionary computation/metaheuristics as optimization tools in commercial design codes.

The book consists of 7 chapters (ordered alphabetically by first author surname), where problems of Civil Engineering are handled by using Evolutionary Algorithms and Metaheuristics as global optimization tools: structural mechanics problems are covered in Chaps. 1, 2 and 4–6, while in Chap. 3 a construction management problem is solved and in Chap. 7 an optimum design methodology of top-edge devices on noise barriers is introduced; the use of surrogate modeling/metamodels and other proposals oriented to reduce the number of real evaluations of the fitness function are taken into account in several chapters (1, 4 and 5); Chaps. 1 and 3 deal with multi-objective optimization, while Chaps. 2 and 4–7 solve single-objective optimization problems.

In Chap. 1, R. Filomeno Coelho et al., propose a complete metamodel-assisted optimization procedure to deal with mixed variables (including discrete, integer, or categorical data), which are required in complex civil engineering structural problems, by using a multi-objective evolutionary algorithm, a multiple kernel regression model, and an efficient online enrichment of the metamodel during the optimization.

In Chap. 2, D. Greiner et al. compare the truss structural optimum design problems of fully stressed design and minimum constrained weight, when using discrete cross-section type bar sizing. An analysis of whole search space in a simple truss test case is included, and optimization behavior of evolutionary algorithms with multiple population sizing and mutation rates is compared.

J. Magalhães-Mendes presents in Chap. 3 a new hybrid genetic algorithm for the time-cost optimization problem with application in construction projects. The approach was developed in Visual Basic language, applied to test problems reported on the literature and compared with other approaches.

Chapter 4 by J. Orkisz and M. Glowacki is devoted to efficiency increase of evolutionary algorithms for large nonlinear constrained optimization problems with applications to mechanics, which include: smoothing and balancing, adaptive

step-by-step mesh refinement, and a' posteriori error analysis and related techniques. It includes their application in residual stresses analysis in elastic-plastic bodies under cyclic loadings.

In Chap. 5, R. Paz et al. propose new optimization strategies based on genetic algorithms combined with surrogate models to reduce as much as possible the number of finite element method simulations in an additive manufacturing application, allowing minimization of weight by using internal cellular and lattice structures.

Chapter 6, by D. Ribeiro et al. deals with calibration methodologies of finite element numerical dynamic models of railway bridges, which imply the resolution of an optimization problem (solved using a genetic algorithm), including residuals associated to natural frequencies and mode shapes. It is applied to the calibration of dynamic models of two railway bridges in the northern line of Portuguese railways with excellent agreement between numerical and experimental responses of the bridges' decks.

The book closes with a procedure for improving the acoustic efficiency of top-edge devices on noise barriers by using the boundary element method (BEM) and genetic algorithms (Chap. 7, by R. Toledo et al.). Both thickness and non-thickness bodies are able to be modeled with a complementary formulation to the classical BEM, presenting here numerical results validating the formulation. Applications to quadratic residue diffuser design and to waterwheel-top barrier design are successfully obtained.

Evolutionary Algorithms and Metaheuristics are becoming an increasing key role in optimum design in many applied sciences and engineering fields, and this book focused on Civil Engineering and Construction Management applications is intended to increase this field interest and foster research in this particular area among the international and ECCOMAS community.

The book editors would like to express their deep appreciation to all contributors for the time and effort devoted to the completion of their contributions to this volume. Finally, the editors would like to thank the personnel of Springer for their most valuable support during the publication process.

Porto, Portugal — Jorge Magalhães-Mendes
Las Palmas de Gran Canaria, Spain — David Greiner

Contents

Chapter 1
On-line Metamodel-Assisted Optimization with Mixed Variables

Rajan Filomeno Coelho, Manuel Herrera, Manyu Xiao and Weihong Zhang

Abstract The optimization of complex civil engineering structures remains a major scientific challenge, mostly because of the high number of calls to the finite element analysis required by the complete design process. To achieve a significant reduction of this computational effort, a popular approach consists in substituting the high-fidelity simulation by a lower-fidelity regression model, also called a metamodel. However, most metamodels (like kriging, radial basis functions, etc.) focus on continuous variables, thereby neglecting the large amount of problems characterized by discrete, integer, or categorical data. Therefore, in this chapter, a complete metamodel-assisted optimization procedure is proposed to deal with mixed variables. The methodology includes a multi-objective evolutionary algorithm and a multiple kernel regression model, both adapted to mixed data, as well as an efficient on-line enrichment of the metamodel during the optimization. A structural benchmark test case illustrates the proposed approach, followed by a critical discussion about the generalization of the concepts introduced in this chapter for metamodel-assisted optimization.

R. Filomeno Coelho (✉)
Building, Architecture & Town Planning (BATir) Department,
Brussels School of Engineering, Université libre de Bruxelles (ULB),
Avenue F.D. Roosevelt, 50 (CP 194/2), 1050 Brussels, Belgium
e-mail: rfilomen@ulb.ac.be

M. Herrera
Department of Civil and Environmental Engineering, Skempton Building,
South Kensington Campus Imperial College London, SW7 2AZ London, UK
e-mail: a.herrera-fernandez@imperial.ac.uk

M. Xiao
Department of Applied Mathematics, Northwestern Polytechnical University,
127 Youyi West Road, Xi'an, Shaanxi 710072, People's Republic of China
e-mail: manyuxiao@nwpu.edu.cn

W. Zhang
Engineering Simulation and Aerospace Computing, School of Mechanical Engineering,
Northwestern Polytechnical University, 127 Youyi West Road, Xi'an, Shaanxi 710072,
People's Republic of China
e-mail: zhangwh@nwpu.edu.cn

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*,
Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2_1

Fig. 1.1 Nijmegen city bridge "De Oversteek" during its construction (September 2013)

Keywords Genetic algorithms · Mixed variables · Categorical variables · Multiple kernel regression · Support vector regression

1.1 Mixed Variables in Civil Engineering

The need for stronger, safer, and greener structures built at shortened delays and at a competitive cost pushes the art of construction to favor elegant lightweight structures, as the city bridge of Nijmegen (The Netherlands) designed by L. Ney and C. Poulissen (see Fig. 1.1),[1] for which an optimization study was conducted by the first author of this chapter.

Nowadays, the design of such remarkable structures is greatly improved by numerical methods, in particular through an efficient combination of evolutionary algorithms to explore the design space, and general regression models (also called metamodels) to avoid a systematic call to the structural finite element analysis [11, 15]. Seen through a simulation-based perspective, in a classical structural optimization process the simulations are integrated within an optimization iterative loop, as depicted in Fig. 1.2. The scientific and technical challenge resides in carrying out a good balance between the use of the high-fidelity (i.e. finite elements) and the low-fidelity (i.e. regression) models, in order to find the best compromise between accuracy and CPU time.

Before such numerical considerations, the first step requires the definition of relevant design variables. As a general rule, the parameterization of civil engineering structures (bridges, dams, buildings, etc.) involves several types of variables representing respectively the geometry (sizing of the elements, overall shape and topology), the materials used, and the boundary conditions (supports). Mathematically, these variables can be classified as follows:

[1]Credits: http://commons.wikimedia.org/.

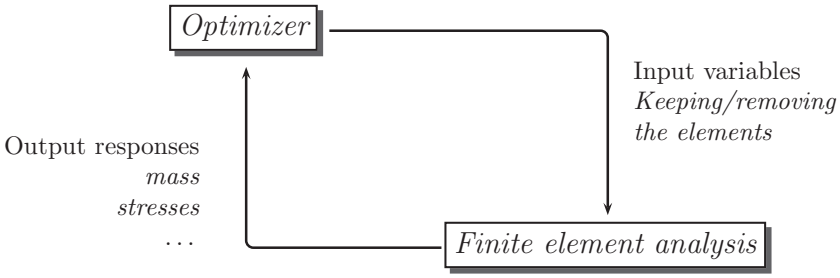

Fig. 1.2 Classical structural optimization process

- *continuous* variables are real numbers defined within an interval $[x^{\min}, x^{\max}] \subseteq \mathbb{R}$;
- *discrete* variables are continuous variables only available among a discrete set. For instance, the cross-section area of a beam profile is an intrinsically continuous parameter defined in square meters, but whose availability might be limited to a discrete sampling taken from a catalog of beam profiles $\{A^{(1)}, A^{(2)}, \ldots, A^{(n)}\}$. This means however that values not available in catalogs can still be computed if necessary, and then rounded off;
- *integer* variables are strictly defined in $\mathbb{N}$. Contrary to discrete variables, intermediate values have no physical meaning. For example, the number of holes drilled in a plate to reduce its weight can be equal to 3 or 4, but any value between these two integer numbers does not correspond to a physical design;
- *categorical* variables represent the remaining non-numerical parameters. Although largely neglected in the optimization literature, they have a huge practical interest in civil engineering, since they can represent for instance the choice of a material ($\{steel, aluminum, \ldots\}$), the type of connection in the assembly of frames ($\{rigid,\ semi\text{-}rigid,\ articulated\}$), the shape of a cross-section ($\{$ O , ■ , I ,... $\}$), etc. If they are endowed with a predefined ranking (e.g. a size $\{S, M, L, XL\}$, a qualitative appreciation $\{weak, normal, strong\}$), they are referred to as *ordinal* variables; otherwise, without intrinsic ordering, they are purely *nominal*.

As mentioned above for the case of discrete variables, a common practice consists in treating non-continuous variables as real numbers, performing an approximation and/or optimization task, and eventually rounding off the solution. The very simple example below will show the danger of using such techniques to deal with intrinsically non-continuous problems.

Let us consider the following three-variable minimization problem:

$$\begin{aligned} \min_{x_1,x_2,x_3} \quad & f(x_1,x_2,x_3) \equiv x_1 + x_2 + x_3 \\ \text{subject to:} \quad & \begin{cases} g(x_1,x_2,x_3) \equiv x_1 + x_2 + x_3 - 10 \geq 0 \\ h(x_1,x_2,x_3) \equiv x_1 - x_2 = 0 \\ x_1, x_2, x_3 \in \{1,2,3,4,5\} \end{cases} \end{aligned} \tag{1.1}$$

Figure 1.3 shows the design space and a few solutions of problem (1). On one hand, $\mathbf{x}^* = (3, 3, 4)$ is a discrete optimum characterized by an objective function value equal to $f(\mathbf{x}^*) = 10$. On the other hand, $\mathbf{x}^{**} = (3.6, 3.6, 2.8)$ is a continuous solution of problem of (1.1) without considering the last constraint on the discrete nature of the variables, and characterized by the same value at the optimum $f(\mathbf{x}^{**}) = 10$; however, by rounding off a posteriori the corresponding design variables to the nearest integer values $\mathbf{x}^{**}_{\text{rounded off}}(4, 4, 3)$, all constraints are now satisfied but the optimality has been lost ($f(\mathbf{x}^{**}_{\text{rounded off}}) = 11$).

This simple application demonstrates the need for considering the discrete, integer, or categorical nature of the variables directly in the optimization phase. Of course, *mixed-integer programming*, implying the relaxation of discrete/integer variables [17], is a powerful approach to handle such problems, and is widely used in combinatorial optimization. Nevertheless, its efficiency depends largely on the mathematical properties of the functions involved (linearity, convexity, etc.), which are usually not guaranteed in civil engineering problems. Besides, recent works on the symmetry of optimal designs in sizing and topology optimization proved that the solution of symmetric problems is always symmetric with continuous variables, but might be asymmetric when discrete variables are involved [19]. All these observations should convince the reader of the importance of treating mixed variables appropriately.

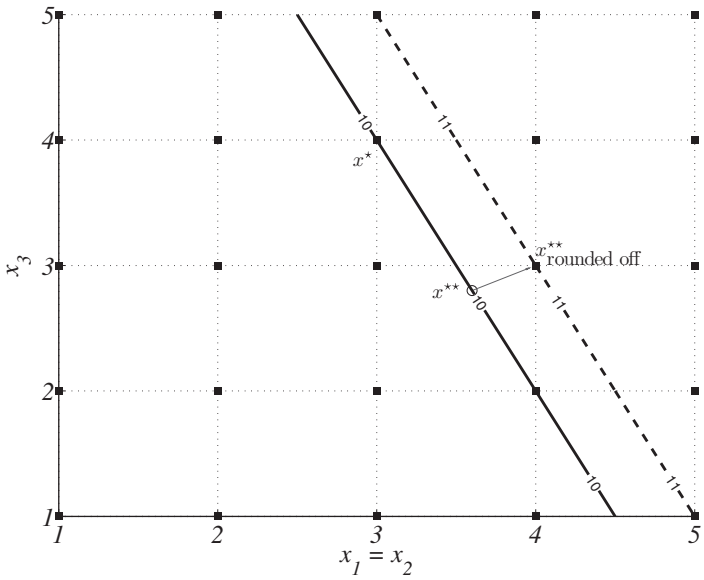

Fig. 1.3 Three-variable example showing the danger of using continuous optimization followed by rounding off the solution for discrete or integer optimization. The design space is depicted in 2D, thereby satisfying automatically the equality constraint $x_1 = x_2$. The black squares represent the discrete design space. While x^* constitutes a valid feasible solution, $x^{**}_{\text{rounded off}}$ obtained by rounding off a valid continuous solution loses the optimality criterion

In this chapter, the emphasis will be put on the simultaneous presence of continuous and categorical variables. After a discussion on the representation of mixed variables (Sect. 1.2), a multiple kernel regression metamodel is described (Sect. 1.3), followed by an on-line metamodel-assisted optimization procedure applied to the design of a rigid frame (Sect. 1.4), and by the conclusions (Sect. 1.5).

1.2 Representation of Mixed Variables

Before diving into the regression and optimization procedures, it is worth considering how mixed data can be mingled together.

Being familiar with genetic algorithms, or simply with computer science in general, the first idea to deal with different types of data is to adopt a *binary coding*. According to the range of the available data (or precision for real numbers), a binary conversion can be operated as shown in Fig. 1.4.

However, it is often preferable to propose a representation closer to the variable types [13]. In this case, a real-number array can be obtained, as illustrated in Fig. 1.5.

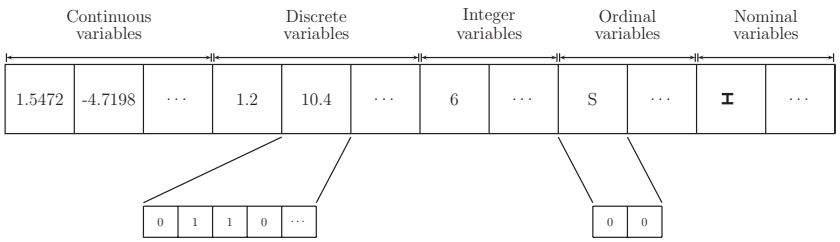

Fig. 1.4 Mixed data converted into an array of binary digits

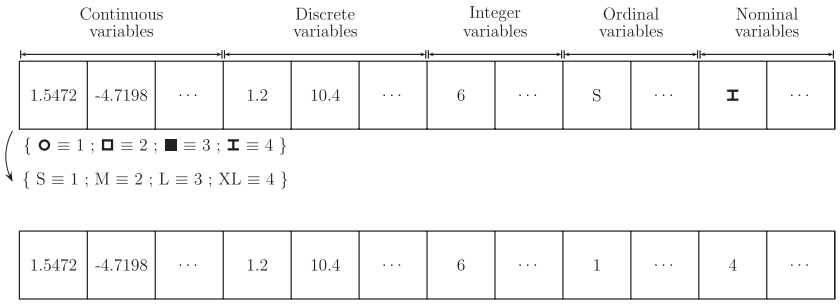

Fig. 1.5 Mixed data converted into an array of real numbers

Table 1.1 Mapping of categorical attributes onto a real number (a binary vector, respectively)

Nominal variable	Mapping 1		Mapping 2	
	Real	Binary	Real	Binary
○	1	(0,0)	2	(0,1)
□	2	(0,1)	4	(1,1)
■	3	(1,0)	1	(0,0)
工	4	(1,1)	3	(1,0)

Both mapping operators are equally valid, but might lead to a different behavior of the approximation/optimization tools without specific care

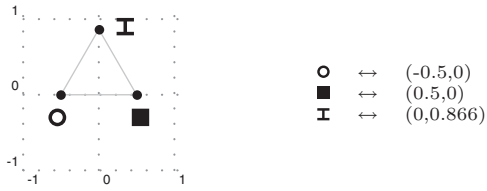

Fig. 1.6 Representation of a nominal variable with three attributes in the 2D space by a standard regular simplex

However, both approaches require an arbitrary mapping of the categorical variables. As shown in Table 1.1, different mappings lead to different conversions, which—without specific care—might lead to a different behavior of the approximation/optimization tools, viz. to a different output prediction (approximation) or optimal solution (optimization).

Therefore, to alleviate this shortcoming, the basic idea of the *regular simplex* method is to assume that any pair of levels of a categorical variable are separated by the same distance (see Fig. 1.6). To achieve this, each level of an n-level variable is associated with a distinct vertex of a regular simplex in $(n-1)$ dimensional space [12, 14]. For simplicity, the Euclidean distance between levels is assumed to be 1. For example, if x^{categ} can take values in a set of $n_{attr} = 3$ possible attributes $\{○ ; ■ ; 工\}$, these attributes can be drawn in a $(n_{attr} - 1)$ space in such a way that each attribute is converted to the vertex coordinates of a standard regular simplex. By construction, all potential values are thus equally distant [4].

Numerical validation performed on several analytical benchmark test cases [8] showed that a real-simplex mapping (i.e. real conversion for continuous, discrete, integer, and ordinal variables, and regular simplex for the nominal variables) is a sound and competitive conversion technique.

The next sections will describe how such mapping techniques can be seamlessly integrated within approximation and optimization procedures.

1.3 Approximation by On-line Multiple Kernel Regression

1.3.1 Multiple Kernel Regression and Support Vector Regression (SVR)

As discussed in the introduction, a reliable and efficient structural optimization process should harmoniously combine an optimization algorithm with an approximate model in order to reach a feasible solution at an affordable computational time. This is the purpose of *metamodel-assisted optimization*, also referred to as *surrogate-based optimization*.

The main steps to devise a surrogate-based optimization algorithm are the following [6]:

1. the variables to be optimized are selected, often due to their importance, as determined by preliminary experiments;
2. a series of initial designs are analyzed by means of the high-fidelity simulation (in structural design, the simulation usually consists in a finite element analysis). The set of designs are selected according to a pre-defined sampling scheme, for example through Latin hypercube sampling [22];
3. a metamodel (e.g. kriging, radial basis function networks, artificial neural networks, polynomial response surfaces, etc.) is used to build a low-fidelity model;
4. the optimization is performed using the low-fidelity model;
5. the results of the optimization are post-processed in order to keep the best point(s). According to the infill criterion selected, new designs are assessed by the high-fidelity simulation, and the corresponding results are added to the existing database to improve the reliability of the metamodel. The process can go back to step 3 and the optimization-approximation cycling is repeated until the stopping criterion is reached (convergence or maximum number of cycles attained).

An off-line optimization process would stop just after step 4, the best solution(s) found being re-assessed by the high-fidelity simulation for verification purposes. In the approach advocated by the authors, an *on-line learning* process is proposed (see step 5). It consists in updating the metamodel both by adding new information and by updating the regression parameters. Thus, this on-line model is updated by the results of an optimization process adapting itself to work better in areas close to the optimum (or optima in multi-objective optimization) found at each cycle (see Fig. 1.7).

First, the metamodel for mixed variables must be presented. In this study, a *kernel*-based approach is followed. Common kernel-based learning methods [20] use an implicit mapping of the input data into a high dimensional feature space defined by a kernel function, i.e. a function K returning the inner product $\langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$ between the images of two data points $\mathbf{x}$, $\mathbf{x}'$ in the feature space (see Table 1.2). The choice of the map ϕ aims at converting the nonlinear relations into

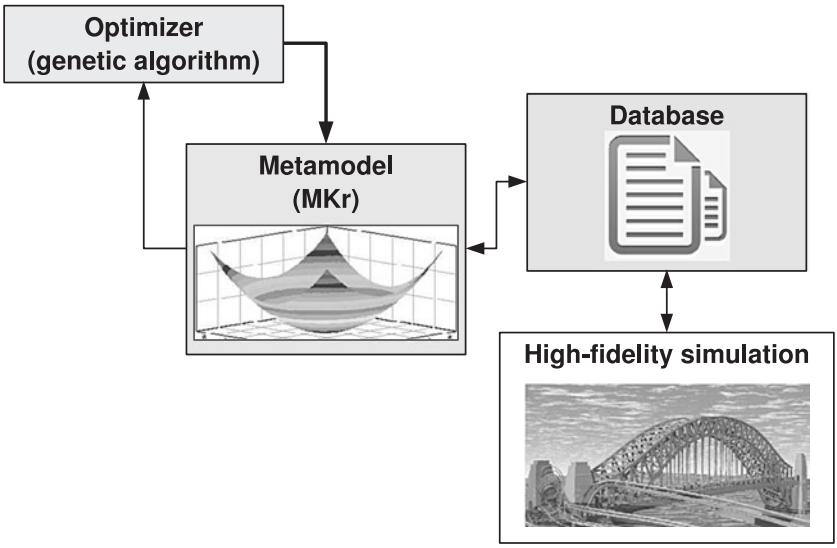

Fig. 1.7 On-line optimization with high-fidelity (= structural finite element analysis) and low-fidelity (= metamodel by multiple kernel regression) simulations

Table 1.2 Short list of some common kernel functions

Name	Expression
Gaussian	$K(\mathbf{x},\mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\sigma^2}\right)$
ANOVA	$K(\mathbf{x},\mathbf{x}') = \sum \exp\left(-\sigma\left(\mathbf{x}^k - \mathbf{x}'^k\right)^2\right)^d$
Linear	$K(\mathbf{x},\mathbf{x}') = \mathbf{x}^T\mathbf{x}' + c$
Polynomial	$K(\mathbf{x},\mathbf{x}') = (\alpha\mathbf{x}^T\mathbf{x}' + c)^d$
Rational quadratic	$K(\mathbf{x},\mathbf{x}') = 1 - \frac{\|\mathbf{x}-\mathbf{x}'\|^2}{\|\mathbf{x}-\mathbf{x}'\|^2+c}$

linear ones. The learning then takes place in the feature space and the learning algorithm can be expressed so that the data points only appear inside dot products with other points. This is often referred to as the "kernel trick" [21].

The use of kernel methods is well adapted to the problem of data integration as it enables multiple types of data to be converted into a common usable format, using one of the representations mentioned in Sect. 1.2. These can be combined eventually with a weighted summation and used as training data for a classical support vector regression (SVR) scheme.

A detailed explanation of SVR is outside the scope of this chapter, but its main principles are summarized hereafter. The key characteristic of SVR is that it allows to specify a margin ε within which we are ready to accept errors in the sample data without affecting the prediction quality. The SVR predictor is defined by the points lying outside the region formed by the band of width $\pm\varepsilon$ around the regression (see Eq. 1.2). Those vectors are the so-called *support vectors*.

$$\hat{f}(\mathbf{x}) = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle + b \tag{1.2}$$

The goal is to find a function $\hat{f}(x)$ that deviates at most by ε from the observed output y_i for the regression based on the training data, and minimizing at the same time the model complexity (see Eq. 1.3):

$$\begin{aligned} &\min_{\mathbf{w},b} \ \frac{1}{2}||\mathbf{w}||^2 \\ &\text{subject to: } y_i - \langle \mathbf{w}, \phi(x_i) \rangle - b \leq \varepsilon \\ &\qquad\qquad \langle \mathbf{w}, \phi(x_i) \rangle + b - y_i \leq \varepsilon \end{aligned} \tag{1.3}$$

The constraints in Eq. 1.3 assume that $\hat{f}(\mathbf{x})$ exists for all y_i with precision $\pm\varepsilon$. Nevertheless, the solution may actually not exist or it would be possible to achieve better predictions if outliers were allowed. Consequently, *slack variables* ξ^+ and ξ^- are introduced:

$$\xi^+ = \hat{f}(x_i) - y(x_i) > \varepsilon \tag{1.4}$$

$$\xi^- = y(x_i) - \hat{f}(x_i) > \varepsilon \tag{1.5}$$

and the objective function and constraints for SVR are

$$\begin{aligned} &\min_{\mathbf{w},b} \ \frac{1}{2}||\mathbf{w}||^2 + C\frac{1}{n}\sum_{i=1}^{n}(\xi_i^+ + \xi_i^-) \\ &\text{subject to: } \ y_i - \langle \mathbf{w}, \phi(x_i) \rangle - b \leq \varepsilon + \xi_i^+, \\ &\qquad\qquad \langle \mathbf{w}, \phi(x_i) \rangle + b - y_i \leq \varepsilon + \xi_i^-, \\ &\qquad\qquad \xi_i^+, \xi_i^- \geq 0 \ i = 1, \ldots, n \end{aligned} \tag{1.6}$$

where n is the number of training patterns and C is a trade-off parameter between model complexity and training error. Additionally, ξ^+ and ξ^- are slack variables for exceeding the target value by more than ε and for being below the target value by more than ε, respectively. This method of tolerating errors is known as ε *-insensitive*.

The usual SVR implementations use a single mapping function ϕ, and hence a single kernel function K. If a data set has a locally varying distribution, using a single kernel may not catch up correctly the varying distribution. Kernel fusion can help to deal with this problem [2]. Recent applications [10] and developments based on support vector machines have shown that using multiple kernels instead of a single one can enhance interpretation of the decision function and improve classifier performance. By the use of different kernels we can address problems from different data nature too. It will reveal beneficial in the perspective of mixed variable programming [1, 7].

We will adopt the weighted sum fusion with the following mapping functions:

$$\Phi(\mathbf{x}) = [\sqrt{\mu_1}\phi_1(\mathbf{x}), \sqrt{\mu_2}\phi_2(\mathbf{x}), \ldots, \sqrt{\mu_M}\phi_M(\mathbf{x})] \tag{1.7}$$

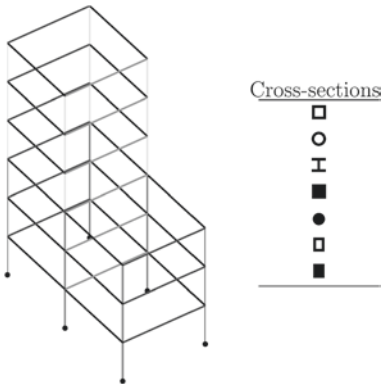

Fig. 1.8 Example of the design of a rigid frame

where $\mu_1, \mu_2, \ldots, \mu_M$ are weights of component functions. Now, the regression problem includes the optimization of two parts. One part is the regression hyperplane $f(\mathbf{x})$ and the other part is the weight vector $\mathbf{m} = [\mu_1, \mu_2, \ldots, \mu_M]$. The idea is to address these two parts of the optimization process in one step, based on the parametric dependence idea.

The resulting multi-kernel, expressed by Eq. 1.8:

$$\begin{aligned}\tilde{K}(x_i, x_j) &= \langle \Phi(x_i), \Phi(x_j) \rangle \\ &= \mu_1 \langle \phi_1(x_i), \phi_1(x_j) \rangle + \mu_2 \langle \phi_2(x_i), \phi_2(x_j) \rangle \\ &\quad + \cdots + \mu_M \langle \phi_M(x_i), \phi_M(x_j) \rangle \\ &= \mu_1 K_1(x_i, x_j) + \mu_2 K_2(x_i, x_j) + \cdots + \mu_M K_M(x_i, x_j) \\ &= \sum_{s=1}^{M} \mu_s K_s(x_i, x_j)\end{aligned} \tag{1.8}$$

is the weighted sum of M kernel functions, constituting a new kernel function. We can solve the regression hyperplane by plugging this multi-kernel into the expression of the SVR regression surface, as shown in Eq. 1.9:

$$\hat{f}(\mathbf{x}) = b + \sum_{i=1}^{n} (\alpha_i^+ - \alpha_i^-) \tilde{K}(x_i, \mathbf{x}). \tag{1.9}$$

1.3.2 On-line Multiple Kernel Regression

In metamodel-assisted optimization, the metamodel is not defined once for all, but is likely to be updated whenever new information from the high-fidelity simulation is made available. Therefore, a metamodel-updating on-line technique is mandatory.

Most of the kernel-based algorithms cannot be used to operate on-line due to a number of difficulties such as time and memory complexities (because of the

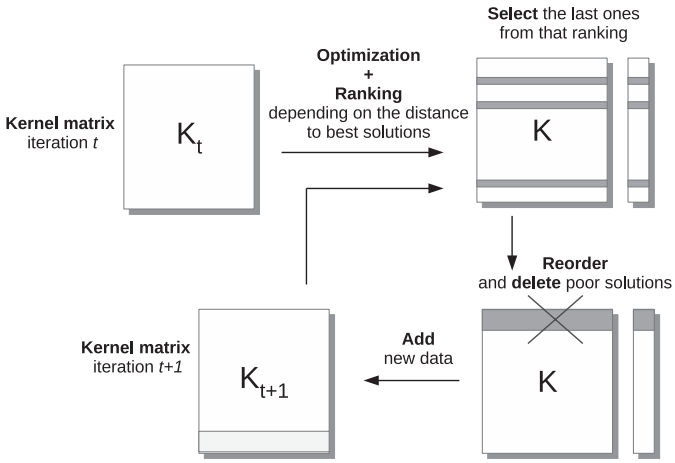

Fig. 1.9 Windowing strategy

growing kernel matrix), and due to the need to avoid over-fitting. However, a few recent experiments were successfully conducted in this sense [9]. For example, a kernel-based recursive least-squares algorithm implementing a fixed size "sliding-window" technique was proposed in [24].

In this chapter, an extension of this methodology to the use of multiple kernels for mixed variables is proposed. Moreover, the windowing process is embedded here into an optimization process. Some additional improvements have thus been included, the main one being to discard the data far from the optimum at each iteration of the process (see Fig. 1.9).

Practically, in the *sliding window* approach proposed here, only the last N pairs of the stream are selected to perform the multi-kernel regression. When a new observed pair $\{x_{n+1}, y_{n+1}\}$ is obtained, the kernel matrix $K_j^{(n)}$ is first down-sized by extracting the contribution from x_{n-N} (see Eq. 1.10):

$$\breve{K}_j^{(n)} = \begin{pmatrix} K_j^{(n)}(2,2) & \cdots & K_j^{(n)}(2,N) \\ \vdots & \ddots & \vdots \\ K_j^{(n)}(N,2) & \cdots & K_j^{(n)}(N,N) \end{pmatrix} \tag{1.10}$$

and then the $K_j^{(n)}$ dimension is augmented again by importing the data input x_{n+1} to obtain the kernel expressed in Eq. 1.11.

$$K_j^{(n+1)} = \begin{pmatrix} \breve{K}_j^{(n)} & K_j(\mathbf{X}_n, x_{n+1}) \\ K_j(x_{n+1}, \mathbf{X}_n) & K_j(x_{n+1}, x_{n+1}) + \lambda \end{pmatrix} \tag{1.11}$$

where $\mathbf{X}_n = (x_{n-N+1}, \ldots, x_n)^T$ and λ is a correction factor.

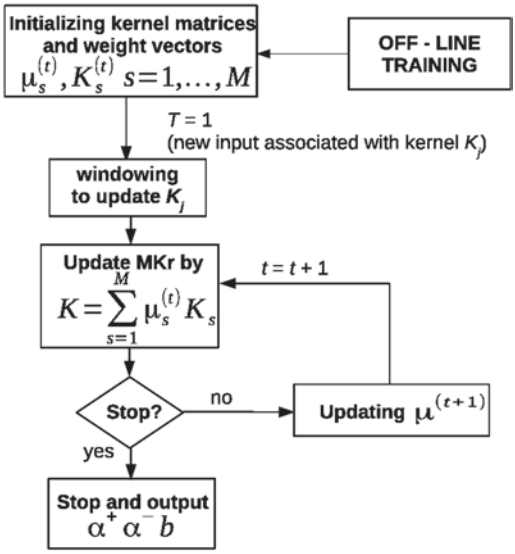

Fig. 1.10 On-line multiple kernel regression: tuning of the regression parameters

Next, the kernel matrices are summed again (see Fig. 1.10) and their weights μ are updated too. Afterwards, the weights and parameters are tuned by a multi-start trust region algorithm for the off-line part [18] (a fast version of this algorithm is included to update the values of the weights at every step of the on-line process).

In the next section, this on-line metamodeling procedure will be coupled to a multi-objective evolutionary algorithm, and applied to a structural optimization test case.

1.4 On-line Metamodel-Assisted Optimization

In order to illustrate the efficiency of on-line regression within an optimization scheme, a structural design optimization problem will be considered (see Fig. 1.8). The numerical application consists in the multi-objective design optimization of a three-dimensional rigid frame with respect to categorical and continuous variables (see Fig. 1.8 [16]). The problem is formulated as follows:

$$\begin{cases} \min\limits_{\mathbf{x}} \mathbf{f}(\mathbf{x}) = \left\{ \begin{array}{c} f_1 \equiv \text{mass} \\ \\ f_2 \equiv \log(\text{compliance}) \end{array} \right\} \\ \text{subject to: } \mathbf{x} = \{c_1, c_2, c_3, c_4, c_5, l_1, l_2, l_3, l_4, l_5\}, \\ \qquad c_i \in \{ \square\ ;\ \circ\ ;\ \text{I}\ ;\ \blacksquare\ ;\ \bullet\ ;\ \square\ ;\ \blacksquare\ \}, i = 1\ldots,5, \\ \qquad l_i \in [0.09, 0.11], i = 1\ldots,5. \end{cases} \tag{1.12}$$

Evolutionary algorithms, thanks to the flexibility of their data structure, are well adapted to deal with mixed variables [23]. The multi-objective optimizer used in this study is the second version of the Non-dominated Sorting Genetic Algorithm (NSGA-II) [3], where the probabilities of simulated binary crossover and mutation are respectively set to 0.9 and 0.5, and the distribution index for simulated binary crossover (η_c) and mutation (η_m) are respectively set to 10 and 20. Its implementation has been modified to tackle nominal variables by adapting the evolutionary operators as follows [5]:

- *crossover*: for each nominal variable and at the user-defined probability of crossover, the operation consists in swapping the values of the parents provided a randomly generated number is above 0.5;
- *mutation*: for each nominal variable and for the user-defined probability of mutation, the operation consists in changing the value of the variable randomly among the set of attributes.

The population of the evolutionary algorithm is set to 200 individuals. In the metamodel-assisted optimization, an initial database of 2,000 samples is needed. Then, at each cycle (i.e. after each optimization with the low-fidelity model), the best 60 designs (according to the NSGA-II ranking criterion) are calculated by means of the high-fidelity finite element analysis, and added to the training database to eventually update the metamodel. The numerical results show (surprisingly) a better coverage of the Pareto front than with the high-fidelity model alone, apparently due to a smoothing of the objective functions predicted by the metamodels (see Fig. 1.11), leading in this case to an improved behavior of the optimizer. Besides, these excellent results are obtained for a reduced number of calls to the finite element (FE) program: at the final (25th) generation, 3,500 FE runs are needed for the on-line process, to be compared to the 5,000 FE runs when the FE program is called systematically.

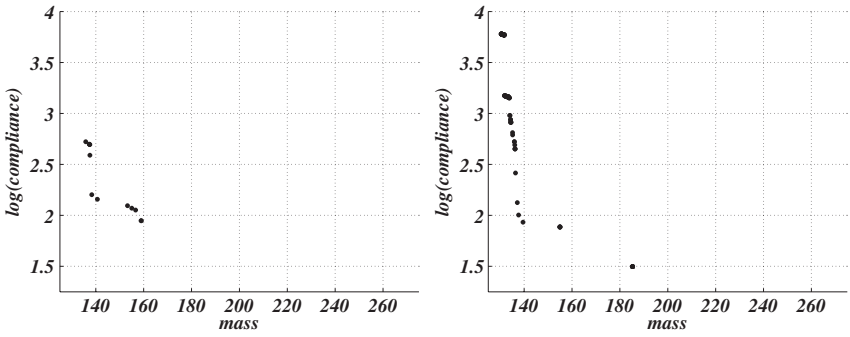

Fig. 1.11 Comparison between high-fidelity optimization process ($\Leftrightarrow$ without using a metamodel, (*left*) and metamodel-assisted on-line optimization (*right*), at the 25th iteration/cycle

1.5 Conclusions

In this chapter, mixed-variate metamodel-assisted optimization has been introduced. Based on the experience of the authors and peers in the field of evolutionary optimization with mixed variables, the reader should keep in mind the following points:

- for both approximation and optimization tasks, the inner nature of the variables should be taken into account. In particular, rounding off continuous optimal solutions to discrete values should be avoided or used with utmost care;
- categorical data should be handled by means of appropriate coding techniques like the regular simplex mapping used in classification and clustering;
- a multi-kernel function is well adapted to deal with variables of different types. Although presented here in the context of support vector regression, it is believed by the authors that the general idea could be fruitfully used in other families of metamodels (for instance with radial basis function networks, which also make use of kernels);
- an efficient on-line procedure to update the metamodels can be coupled seamlessly to an evolutionary algorithm. However, although not treated in detail in this chapter, the main issue consists in defining a proper interaction between the metamodel and the optimizer, by means of suitable *infill criteria* [6]. These criteria define which points should be assessed by the high-fidelity model and added to the database. New points can typically be the best points obtained so far, the points endowed with the highest prediction error of the metamodels (hence requiring a re-sampling to improve the accuracy of the low-fidelity model), or a combination of both paradigms.

Future prospects in this field include the handling of all kinds of variables (continuous, discrete, integer, and categorical altogether), and its validation on several benchmark test cases.

Acknowledgments This work has been supported by Innoviris (Brussels-Capital Region, Belgium) through a BB2B project entitled “Multicriteria optimization with uncertainty quantification applied to the building industry”.

The authors also acknowledge support by the Basic Project Foundation of Northwestern Polytechnical University (JC20120241), and by the National Natural Science Foundation of China (Grants No. 11302173, 51275424, and 11432011).

References

1. Abramson M, Audet C, Dennis DE Jr (2007) Filter pattern search algorithms for mixed variable constrained optimization problems. Pac J Optim 3(3):477–500
2. Christmann A, Hable R (2012) Consistency of support vector machines using additive kernels for additive models. Comput Stat Data Anal 56(4):854–873
3. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182–197

4. Filomeno Coelho R (2014) Metamodels for mixed variables based on moving least squares—Application to the structural analysis of a rigid frame. Optim Eng 15(2):311–329
5. Filomeno Coelho R, Xiao M, Guglielmetti A, Herrera M, Zhang W (2015) Investigation of three genotypes for mixed variable evolutionary optimization. In: Greiner D et al (eds) Advances in evolutionary and deterministic methods for design, optimization and control in engineering and sciences. Computational methods in applied sciences, vol 36. Springer, New York, pp 309−319
6. Forrester AIJ, Keane AJ (2009) Recent advances in surrogate-based optimization. Prog Aerosp Sci 45(1–3):50–79
7. Hemker T, Fowler KR, Farthing MW, von Stryk O (2008) A mixed-integer simulation-based optimization approach with surrogate functions in water resources management. Optim Eng 9:341–360
8. Herrera M, Guglielmetti A, Xiao M, Filomeno Coelho R (2014) Metamodel-assisted optimization based on multiple kernel regression for mixed variables. Struct Multidiscip Optim 49(6):979–991
9. Hoi SCH, Jin R, Zhao P, Yang T (2013) Online multiple kernel classification. Mach Learn 90:289–316
10. Luts J, Molenberghs G, Verbeke G, Van Huffel S, Suykens JAK (2012) A mixed effects least squares support vector machine model for classification of longitudinal data. Comput Stat Data Anal 56(3):611–628
11. Machairas V, Tsangrassoulis A, Axarli K (2014) Algorithms for optimization of building design: A review. Renew Sustain Energy Rev 31:101–112
12. McCane B, Albert MH (2008) Distance functions for categorical and mixed variables. Pattern Recogn Lett 29(7):986–993
13. Michalewicz Z (1996) Genetic Algorithms + Data Structures = Evolution Programs. Springer, New York
14. Mortier F, Robin S, Lassalvy S, Baril CP, Bar-Hen A (2006) Prediction of Euclidean distances with discrete and continuous outcomes. Multivar Anal 97(8):1799–1814
15. Nguyen A-T, Reiter S, Rigo P (2014) A review on simulation-based optimization methods applied to building performance analysis. Appl Energy 113:1043–1058
16. Papadrakakis M, Lagaros ND (2002) Reliability-based structural optimization using neural networks and Monte Carlo simulation. Comput Methods Appl Mech Eng 191:3491–3507
17. Pardalos PM, Prokopyev OA, Busygin S (2006) Continuous approaches for solving discrete optimization problems. In: Appa G, Pitsoulis L, Williams HP (eds) Handbook on modelling for discrete optimization. International Series in Operations Research & Management Science. Springer Science+Business Media B.V., pp 39–60
18. Peri D, Tinti F (2012) A multistart gradient-based algorithm with surrogate model for global optimization. Commun Appl Ind Math 3(1)
19. Richardson JN, Adriaenssens S, Bouillard Ph, Filomeno Coelho R (2013) Symmetry and asymmetry of solutions in discrete variable structural optimization. Struct Multidiscip Optim 47(5):631–643
20. Schölkopf B, Smola AJ (2001) Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT Press, Cambridge
21. Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14(3):199–222
22. Swiler LP, Wyss GD (2004) A user's guide to Sandia's Latin hypercube sampling software: LHS UNIX library/standalone version. Technical report, Sandia National Laboratories, Albuquerque, New Mexico
23. Tang X, Bassir DH, Zhang W (2011) Shape, sizing optimization and material selection based on mixed variables and genetic algorithm. Optim Eng 12:111–128
24. van Vaerenbergh S, Vía J, Santamaría I (2006) A sliding-window kernel RLS algorithm and its application to nonlinear channel identification. In: IEEE international conference on acoustics, speech and signal processing—ICASSP 2006, vol 5, pp 789–792

Chapter 2
Comparing the Fully Stressed Design and the Minimum Constrained Weight Solutions in Truss Structures

David Greiner, José M. Emperador, Blas Galván and Gabriel Winter

Abstract The optimization structural design problems of Fully Stressed Design (FSD) and Minimum Constrained Weight (MCW) are compared in this work in a simple truss test case with discrete cross-section type bar sizing, where both optimum designs are coincident. An analysis of the whole search space is included, and the optimization behaviour of evolutionary algorithms are compared with multiple population sizing and mutation rates in both problems. Results of average, best and standard deviation metrics indicate the success and the robustness of the methodology, as well as the fastest and easiest behaviour when considering the FSD case.

Keywords Structural design · Truss optimization · Evolutionary algorithms · Fully stressed design · Minimum constrained weight

2.1 Introduction

The use of evolutionary algorithms/metaheuristics has allowed the resolution of the global optimum design of many engineering problems (see e.g. [2, 10]), and particularly, in the case of discrete cross-section bar structures since the first nineties of the twentieth century [1, 7, 8]. In this book chapter, it is handled a comparative and relational study of the search algorithm performance in two structural problems: first, the minimization of the constrained weight and, second, the obtainment of the fully stressed design. Results using the above mentioned global search methods in a simple truss structure considering some statistical metrics are

D. Greiner (✉) · J.M. Emperador · B. Galván · G. Winter
Institute of Intelligent Systems and Numerical Applications in Engineering SIANI,
Universidad de Las Palmas de Gran Canaria ULPGC, 35017 Las Palmas, Spain
e-mail: david.greiner@ulpgc.es

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*,
Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2_2

obtained. First, the structural handled problems are described in Sect. 2.2, then the test case is shown in Sect. 2.3. Section 2.4 presents results and discussion, and finally, the conclusions Sect. 2.5 ends this book chapter.

2.2 Structural Problems

Two optimization problems of bar structures with discrete section-types are fronted in this book chapter.

In first place, the problem of minimization of the constrained structural weight (MCW), which is related with the minimization of raw cost of the structure, is considered (Eq. 2.1) (e.g. see [4, 12]). It is the most common structural optimum design problem.

$$MCW = \sum_{i=1}^{Nbars} A_i \cdot l_i \cdot \rho_i \tag{2.1}$$

where A_i is cross-sectional area, l_i is length and ρ_i is specific weight, all corresponding to bar i; subjected under constraints of stresses, displacements and/or buckling. In this chapter only stresses constraints are taken into account, and treated as in [6].

In second place, the problem of achieving the fully stressed design (FSD) structure is considered (e.g., see [14]), which has been handled since the beginning of the 20th century. The FSD of a structure is defined as the design in which some location of every bar member in the structure is at its maximum allowable stress for at least one loading condition.

$$FSD = \sqrt{\sum_{i=1}^{Nbars} (\sigma_{MAX-i} - \sigma_{MAX-Ri})^2} \tag{2.2}$$

where $\sigma_{MAX\text{-}i}$ and $\sigma_{MAX\text{-}Ri}$ are the maximum stress and the maximum allowable stress, respectively, both corresponding to bar i.

Some relation between both previous problems, MCW and FSD, has been established, mainly in trusses structures where the material is allowed to work at its full potential due to the only existence of normal efforts, associated with the cross-sectional area [11, 13]. In this work, we show through the use of metaheuristic global optimization methods in discrete cross section-type trusses, that even in the possible case that both problems (MCW and FSD) share the same optimum solution, the search still has different characteristics and topology, which makes easier or harder to solve for the global search evolutionary algorithm.

2.3 Test Case

The test case used is a simple test case with truss bar structures based on one in [15] and [16], solving it with discrete cross-section types variables (as in [9]). The computational domain, loading and boundary conditions are shown in Fig. 2.1, with Load P = 4450 N. This test case has been solved also for simultaneous (that is, multiobjective optimization) minimization of weight and maximization of the reliability index in Greiner and Hajela [3].

Each bar corresponds to an independent variable. Table 2.1 shows the set of cross section types and their geometric properties (area, radius of gyration). Table 2.2 represents the search space of variables, including the lower and upper limit of each variable.

An own implemented truss bar structure stiffness matrix calculation has been used to evaluate the structural variables, where articulated nodes (that is

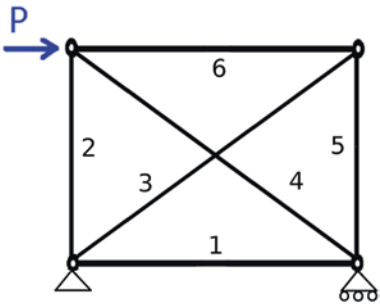

Fig. 2.1 Test case. Computational domain, loading and boundary conditions

Table 2.1 Cross-section types

Order	Cross-section	Area (cm^2)	Radius of gyration (cm)
1	C1	0.85	0.653
2	C2	0.93	0.652
3	C3	1.01	0.651
4	C4	1.09	0.650
5	C5	1.17	0.649
6	C6	1.25	0.648
7	C7	1.33	0.647
8	C8	1.41	0.646
9	C9	1.49	0.645
10	C10	1.57	0.644
11	C11	1.65	0.643
12	C12	1.73	0.642
13	C13	1.81	0.641
14	C14	1.89	0.640
15	C15	1.97	0.639
16	C16	2.05	0.638

Table 2.2 Search space of variables

Bar number	Bar variable	Cross-section type set
1	v1	From C1 to C16
2	v2	From C1 to C16
3	v3	From C1 to C16
4	v4	From C1 to C16
5	v5	From C1 to C16
6	v6	From C1 to C16

Table 2.3 Geometric parameters

	Value (m.)
Height (H)	0.9144
Width (W)	1.2190

Table 2.4 Material properties (Steel)

Parameter	Value
Density	7850 kg/m3
Young modulus	2.06×10^5 MPa
Maximum stress	276 MPa

non-resisting moment capabilities) are considered, elastic behaviour of steel is assumed, and no buckling effect is taken into account in these results. Table 2.3 shows the geometric parameters (height and width) of the structure. Table 2.4 exposes its material properties, -those of standard construction steel-.

In order to define the cross-section type sizing of each bar (that is the structural design), the quantities of interest are the values of the fitness function/s (minimum constrained weight and /or fully stress design) and the maximum stress of each bar.

2.4 Results and Discussion

2.4.1 Test Case Analysis

This section studies the relationship between the MCW problem and the FSD problem in our test case (as described in Sect. 2.3).

Therefore, the whole search space of the previous test case has been explored, evaluating both objectives: (a) Minimum Constrained Weight (MCW, in kg.) as shown in Eq. 2.1, and (b) Square root of the sum of squared stress differences of each bar with Fully Stressed Design (FSD) as shown in Eq. 2.2.

Values of the $2^{24} = 16{,}777{,}216$ designs (corresponding to a 6 bar $\times$ 4 bits/bar chromosome $=$ 24 bits) are obtained and shown in Fig. 2.2 (whole search space).

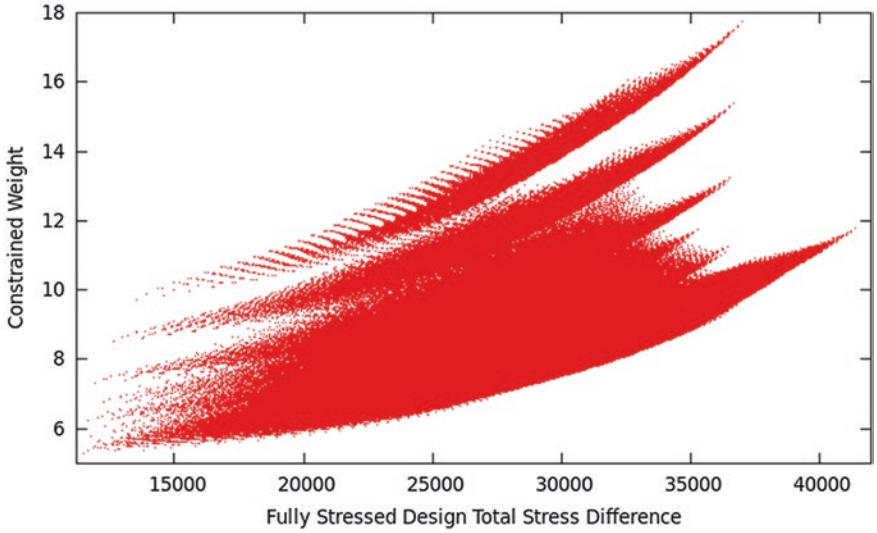

Fig. 2.2 Whole search space designs of test case

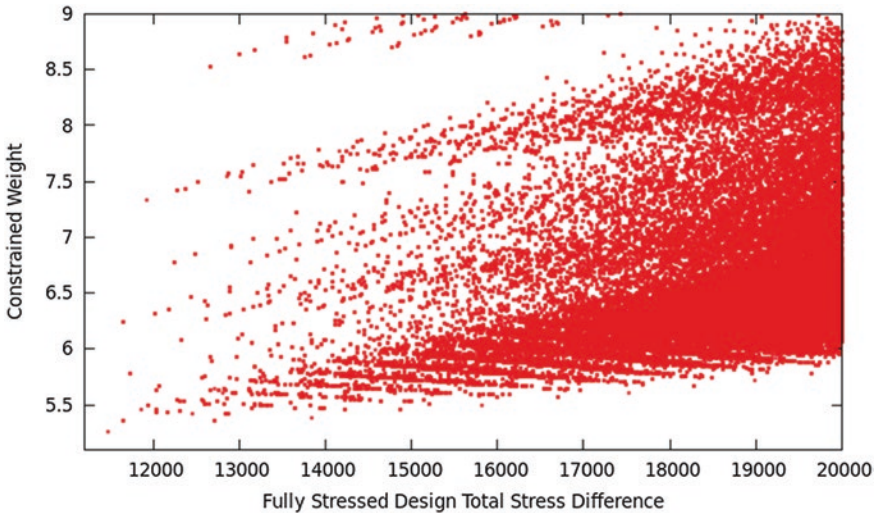

Fig. 2.3 Zommed vision over Fig. 2.2 (search space designs of test case)

In addition, a zoomed picture of the best solution designs are shown in Fig. 2.3. In this test case, the minimum of both fitness functions is a coincident design, the most left and bottom point in this Fig. 2.3. Moreover, calculating the population Pearson's correlation coefficient r between both objectives (MCW and FSD) gives a value $r = 0.71089$ (where 1.0 means a perfect linear relationship).

The detailed best fifty designs of each objective are shown in Table 2.5 (MCW optimum designs) and Table 2.6 (FSD optimum designs). In addition to the optimum design, which is shared by the problem of MCW and the problem of FSD, fifteen designs out of this list of fifty are included in both sets (all shared designs are highlighted in bold type in the tables).

Table 2.5 Minimum MCW designs (cross-section types as in Table 2.1)

Design order	FSD value	MCW value	Unconstrained weight	Bar 1	Bar 2	Bar 3	Bar 4	Bar 5	Bar 6
1st	**11480.7**	**5.26338**	**5.26338**	**C1**	**C1**	**C3**	**C3**	**C1**	**C1**
2nd	12709.5	5.3581	5.3208	C1	C1	C3	C3	C2	C1
3rd	12709.5	5.3581	5.3208	C1	C2	C3	C3	C1	C1
4th	**11645.8**	**5.35907**	**5.35907**	**C1**	**C1**	**C4**	**C3**	**C1**	**C1**
5th	**11645.8**	**5.35907**	**5.35907**	**C1**	**C1**	**C3**	**C4**	**C1**	**C1**
6th	13834.7	5.37822	5.37822	C1	C2	C3	C3	C2	C1
7th	**12449.8**	**5.41648**	**5.41648**	**C2**	**C1**	**C3**	**C3**	**C1**	**C2**
8th	12840.5	5.4165	5.4165	C1	C2	C3	C4	C1	C1
9th	12875.2	5.4165	5.4165	C1	C1	C3	C4	C2	C1
10th	12840.5	5.4165	5.4165	C1	C1	C4	C3	C2	C1
11st	12875.2	5.4165	5.4165	C1	C2	C4	C3	C1	C1
12nd	**12163.8**	**5.43563**	**5.43563**	**C2**	**C1**	**C4**	**C3**	**C1**	**C1**
13rd	**12038.5**	**5.43563**	**5.43563**	**C2**	**C1**	**C3**	**C4**	**C1**	**C1**
14th	**12038.5**	**5.43563**	**5.43563**	**C1**	**C1**	**C4**	**C3**	**C1**	**C2**
15th	**12163.8**	**5.43563**	**5.43563**	**C1**	**C1**	**C3**	**C4**	**C1**	**C2**
16th	**12312.0**	**5.43998**	**5.43563**	**C1**	**C1**	**C5**	**C2**	**C1**	**C2**
17th	**12312.0**	**5.43998**	**5.43563**	**C2**	**C1**	**C2**	**C5**	**C1**	**C1**
18th	**12036.9**	**5.45477**	**5.45477**	**C1**	**C1**	**C5**	**C3**	**C1**	**C1**
19th	**11861.9**	**5.45477**	**5.45477**	**C1**	**C1**	**C4**	**C4**	**C1**	**C1**
20th	**12036.9**	**5.45477**	**5.45477**	**C1**	**C1**	**C3**	**C5**	**C1**	**C1**
21st	14232.6	5.45478	5.45478	C2	C1	C3	C3	C3	C1
22nd	14232.6	5.45478	5.45478	C1	C3	C3	C3	C1	C2
23rd	14861.9	5.46076	5.43565	C1	C2	C3	C3	C3	C1
24th	14861.9	5.46076	5.43565	C1	C3	C3	C3	C2	C1
25th	13148.3	5.46882	5.39735	C2	C1	C3	C3	C2	C1
26th	13148.3	5.46882	5.39735	C1	C2	C3	C3	C1	C2
27th	13645.3	5.47391	5.47391	C3	C2	C2	C4	C1	C1
28th	13645.3	5.47391	5.47391	C1	C1	C4	C2	C2	C3
29th	13977.5	5.47392	5.47392	C1	C1	C3	C4	C3	C1
30th	13977.5	5.47392	5.47392	C1	C3	C4	C3	C1	C1
31st	13930.1	5.47392	5.47392	C1	C1	C4	C3	C3	C1
32nd	13930.1	5.47392	5.47392	C1	C3	C3	C4	C1	C1
33rd	13970.5	5.47392	5.47392	C1	C2	C4	C3	C2	C1

(continued)

Table 2.5 (continued)

Design order	FSD value	MCW value	Unconstrained weight	Bar 1	Bar 2	Bar 3	Bar 4	Bar 5	Bar 6
34th	13970.5	5.47392	5.47392	C1	C2	C3	C4	C2	C1
35th	13818.0	5.48927	5.37822	C1	C1	C3	C3	C3	C1
36th	13818.0	5.48927	5.37822	C1	C3	C3	C3	C1	C1
37th	13278.3	5.49303	5.49303	C4	C1	C2	C4	C1	C1
38th	13278.3	5.49303	5.49303	C1	C1	C4	C2	C1	C4
39th	13330.9	5.49305	5.49305	C2	C1	C4	C3	C2	C1
40th	13249.4	5.49305	5.49305	C2	C1	C3	C4	C2	C1
41st	13175.6	5.49305	5.49305	C1	C1	C4	C3	C2	C2
42nd	13175.6	5.49305	5.49305	C2	C2	C3	C4	C1	C1
43rd	13249.4	5.49305	5.49305	C1	C2	C4	C3	C1	C2
44th	13381.0	5.49305	5.49305	C2	C2	C2	C5	C1	C1
45th	13381.0	5.49305	5.49305	C1	C1	C5	C2	C2	C2
46th	13330.9	5.49305	5.49305	C1	C2	C3	C4	C1	C2
47th	13328.1	5.49305	5.49305	C2	C2	C4	C3	C1	C1
48th	13328.1	5.49305	5.49305	C1	C1	C3	C4	C2	C2
49th	15823.2	5.49307	5.49307	C1	C3	C3	C3	C3	C1
50th	**11944.9**	**5.49621**	**5.33993**	**C2**	**C1**	**C3**	**C3**	**C1**	**C1**
51th	**11944.9**	**5.49621**	**5.33993**	**C1**	**C1**	**C3**	**C3**	**C1**	**C2**

Table 2.6 Minimum FSD designs (cross-section types as in Table 2.1)

Design order	FSD value	MCW value	Unconstrained weight	Bar 1	Bar 2	Bar 3	Bar 4	Bar 5	Bar 6
1st	**11480.7**	**5.26338**	**5.26338**	**C1**	**C1**	**C3**	**C3**	**C1**	**C1**
2nd	**11645.8**	**5.35907**	**5.35907**	**C1**	**C1**	**C3**	**C4**	**C1**	**C1**
3rd	**11645.8**	**5.35907**	**5.35907**	**C1**	**C1**	**C4**	**C3**	**C1**	**C1**
4th	11649.7	6.24373	5.16768	C1	C1	C3	C2	C1	C1
5th	11649.7	6.24373	5.16768	C1	C1	C2	C3	C1	C1
6th	11735.5	5.77910	5.26338	C1	C1	C4	C2	C1	C1
7th	11735.5	5.77910	5.26338	C1	C1	C2	C4	C1	C1
8th	**11861.9**	**5.45477**	**5.45477**	**C1**	**C1**	**C4**	**C4**	**C1**	**C1**
9th	11922.0	7.32865	5.07198	C1	C1	C2	C2	C1	C1
10th	**11944.9**	**5.49621**	**5.33993**	**C1**	**C1**	**C3**	**C3**	**C1**	**C2**
11st	**11944.9**	**5.49621**	**5.33993**	**C2**	**C1**	**C3**	**C3**	**C1**	**C1**
12nd	12026.5	6.31897	5.24423	C2	C1	C2	C3	C1	C1
13rd	12026.5	6.31897	5.24423	C1	C1	C3	C2	C1	C2
14th	**12036.9**	**5.45477**	**5.45477**	**C1**	**C1**	**C5**	**C3**	**C1**	**C1**
15th	**12036.9**	**5.45477**	**5.45477**	**C1**	**C1**	**C3**	**C5**	**C1**	**C1**
16th	**12038.5**	**5.43563**	**5.43563**	**C2**	**C1**	**C3**	**C4**	**C1**	**C1**
17th	**12038.5**	**5.43563**	**5.43563**	**C1**	**C1**	**C4**	**C3**	**C1**	**C2**

(continued)

Table 2.6 (continued)

Design order	FSD value	MCW value	Unconstrained weight	Bar 1	Bar 2	Bar 3	Bar 4	Bar 5	Bar 6
18th	12041.2	5.63246	5.33993	C2	C1	C2	C4	C1	C1
19th	12041.2	5.63246	5.33993	C1	C1	C4	C2	C1	C2
20th	12069.7	5.67125	5.35907	C1	C1	C5	C2	C1	C1
21st	12069.7	5.67125	5.35907	C1	C1	C2	C5	C1	C1
22nd	**12163.8**	**5.43563**	**5.43563**	**C2**	**C1**	**C4**	**C3**	**C1**	**C1**
23rd	**12163.8**	**5.43563**	**5.43563**	**C1**	**C1**	**C3**	**C4**	**C1**	**C2**
24th	12186.1	6.35272	5.24423	C2	C1	C3	C2	C1	C1
25th	12186.1	6.35272	5.24423	C1	C1	C2	C3	C1	C2
26th	12248.0	6.77926	5.16768	C1	C1	C1	C4	C1	C1
27th	12248.0	6.77926	5.16768	C1	C1	C4	C1	C1	C1
28th	12271.6	7.42623	5.07198	C1	C1	C1	C3	C1	C1
29th	12271.6	7.42623	5.07198	C1	C1	C3	C1	C1	C1
30th	12281.2	5.55047	5.55047	C1	C1	C5	C4	C1	C1
31st	12281.2	5.55047	5.55047	C1	C1	C4	C5	C1	C1
32nd	12309.8	5.53132	5.53132	C2	C1	C4	C4	C1	C1
33rd	12309.8	5.53132	5.53132	C1	C1	C4	C4	C1	C2
34th	**12312.0**	**5.43998**	**5.43563**	**C2**	**C1**	**C2**	**C5**	**C1**	**C1**
35th	**12312.0**	**5.43998**	**5.43563**	**C1**	**C1**	**C5**	**C2**	**C1**	**C2**
36th	12324.0	6.07500	5.33993	C2	C1	C4	C2	C1	C1
37th	12324.0	6.07500	5.33993	C1	C1	C2	C4	C1	C2
38th	12364.8	5.53132	5.53132	C1	C1	C5	C3	C1	C2
39th	12364.8	5.53132	5.53132	C2	C1	C3	C5	C1	C1
40th	12371.7	7.43926	5.14853	C2	C1	C2	C2	C1	C1
41st	12371.7	7.43926	5.14853	C1	C1	C2	C2	C1	C2
42nd	12432.3	6.46520	5.24423	C2	C1	C1	C4	C1	C1
43rd	12432.3	6.46520	5.24423	C1	C1	C4	C1	C1	C2
44th	**12449.8**	**5.41648**	**5.41648**	**C2**	**C1**	**C3**	**C3**	**C1**	**C2**
45th	12491.7	6.84455	5.26338	C1	C1	C5	C1	C1	C1
46th	12491.7	6.84455	5.26338	C1	C1	C1	C5	C1	C1
47th	12525.0	7.50132	5.14853	C2	C1	C1	C3	C1	C1
48th	12525.0	7.50132	5.14853	C1	C1	C3	C1	C1	C2
49th	12562.8	5.74984	5.45477	C1	C1	C6	C2	C1	C1
50th	12562.8	5.74984	5.45477	C1	C1	C2	C6	C1	C1

2.4.2 Optimization

In this section, both the MCW and the FSD problems are optimized.

An evolutionary algorithm with two population sizes (80 and 160 individuals), two mutation rates (1.5 and 3 %), uniform crossover and gray codification

(as in [5]) has been executed in 100 independent runs with 30,000 evaluations as stopping criterion. Results are shown in terms of average, best value and standard deviation of the fitness functions.

(a) Solving the problem of minimum constrained weight MCW optimum design: Figure 2.4 shows the average fitness value, Fig. 2.5 shows the best fitness value and Fig. 2.6 shows the standard deviation fitness value; each containing the population size of 80 in black lines and the population size of 160 in pink (gray) lines;

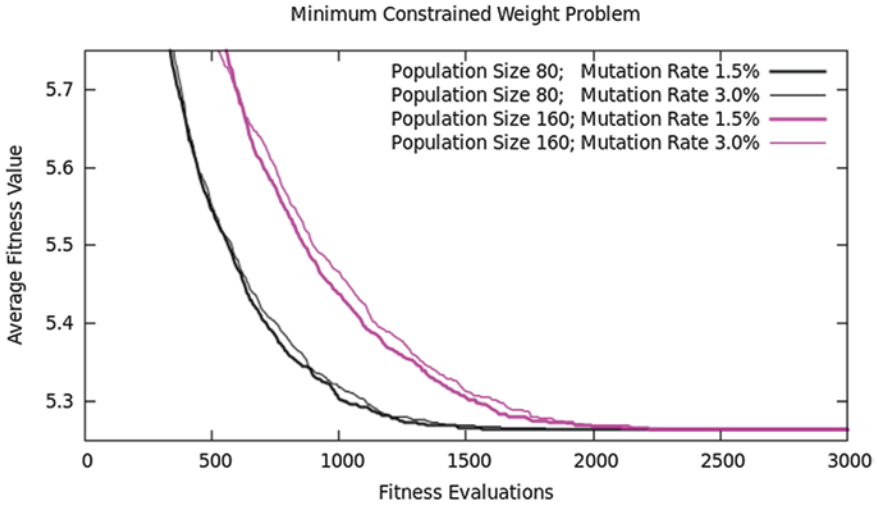

Fig. 2.4 Constrained minimum weight evolution over 100 independent runs in cR00 test case

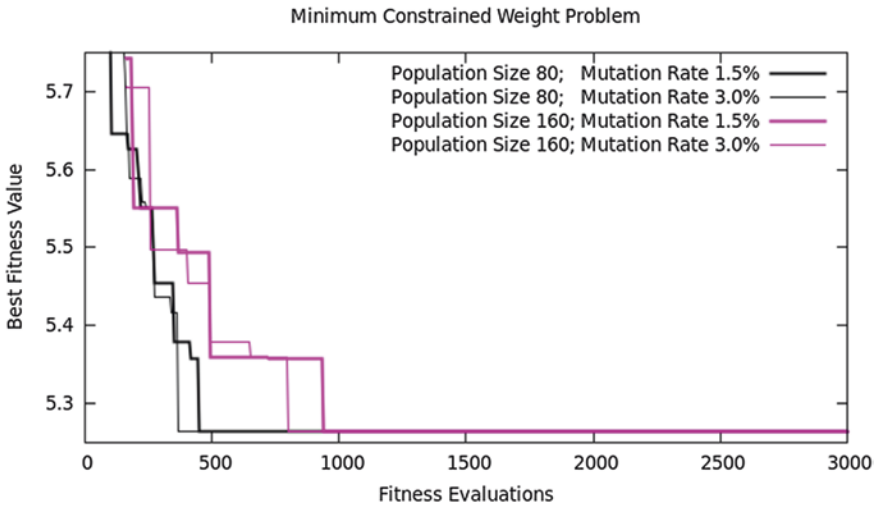

Fig. 2.5 Constrained minimum weight evolution over 100 independent runs in cR00 test case

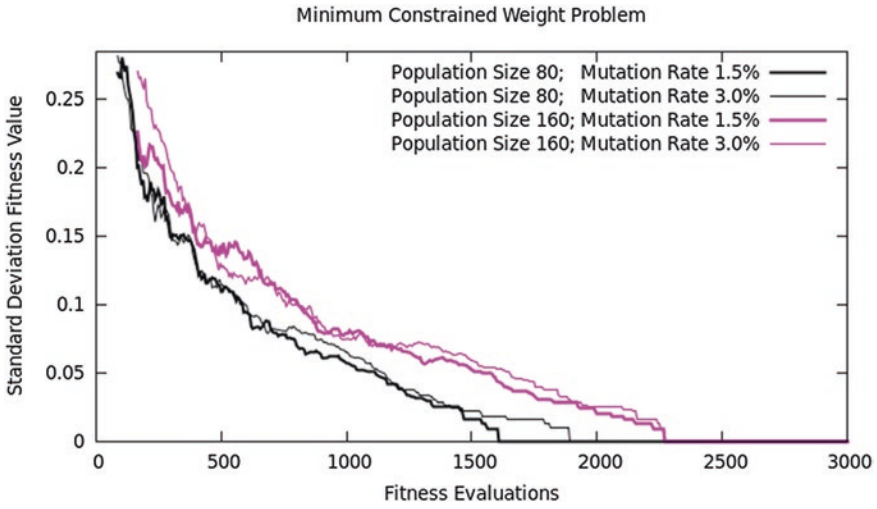

Fig. 2.6 Constrained minimum weight evolution over 100 independent runs in cR00 test case

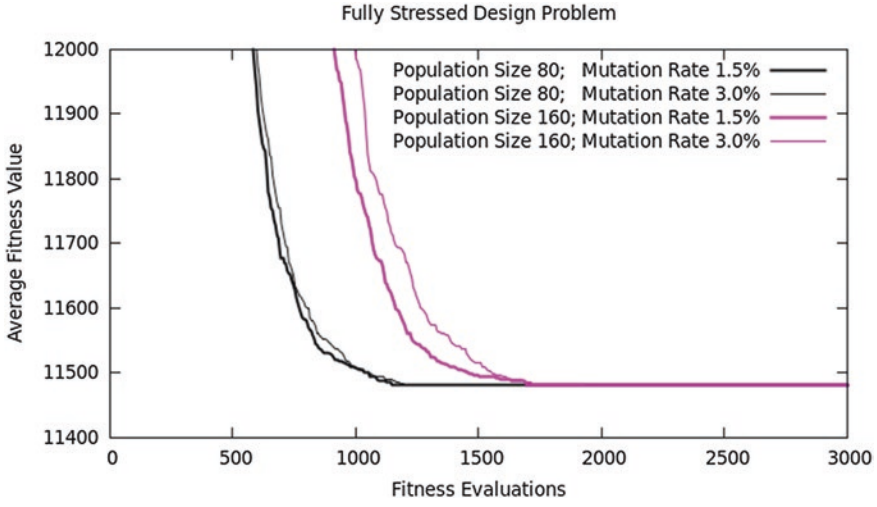

Fig. 2.7 FSD evolution over 100 independent runs in cR00 test case

the thicker line belongs to the 1.5 % mutation rate and the thinner line belongs to the 3.0 % mutation rate.

(b) Solving the problem of fully stressed design FSD optimum design:
Figure 2.7 shows the average fitness value, Fig. 2.8 shows the best fitness value and Fig. 2.9 shows the standard deviation fitness value; each containing the population size of 80 in black lines and the population size of 160 in pink (gray) lines; the thicker line belongs to the 1.5 % mutation rate and the thinner line belongs to the 3.0 % mutation rate.

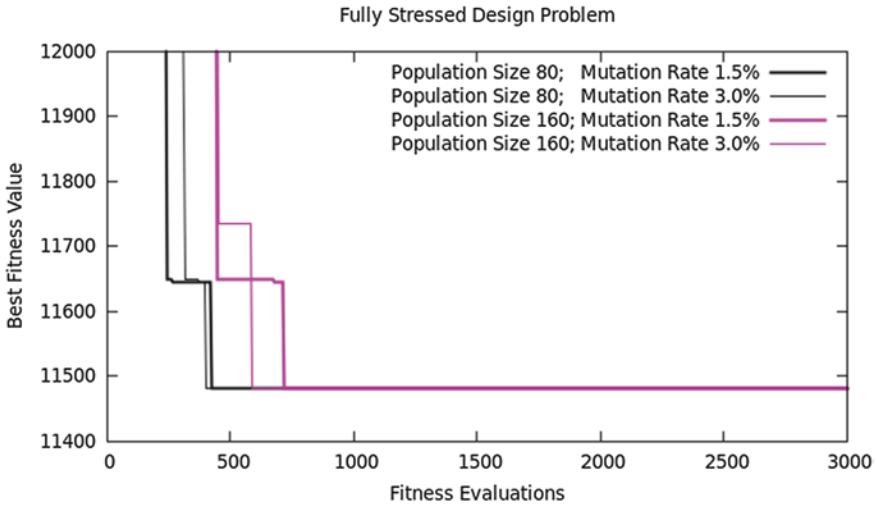

Fig. 2.8 FSD evolution over 100 independent runs in cR00 test case

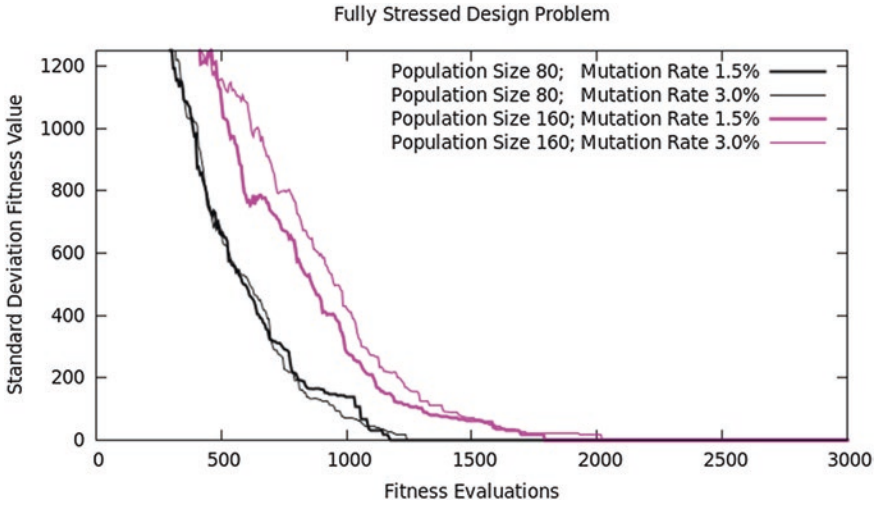

Fig. 2.9 FSD evolution over 100 independent runs in cR00 test case

(c) Comparing FSD and MCW problems with population size 80:
In this case, fitness values have been scaled between 0 and 1, in order to easily compare visually the behaviour of both problems (FSD and MCW). Figure 2.10 shows the scaled average fitness value, Fig. 2.11 shows the scaled best fitness value and Fig. 2.12 shows the scaled standard deviation fitness value; each containing the FSD optimization problem in black lines and the MCW optimization problem in pink (gray) lines; the thicker line belongs to the 1.5 % mutation rate and the thinner line belongs to the 3.0 % mutation rate.

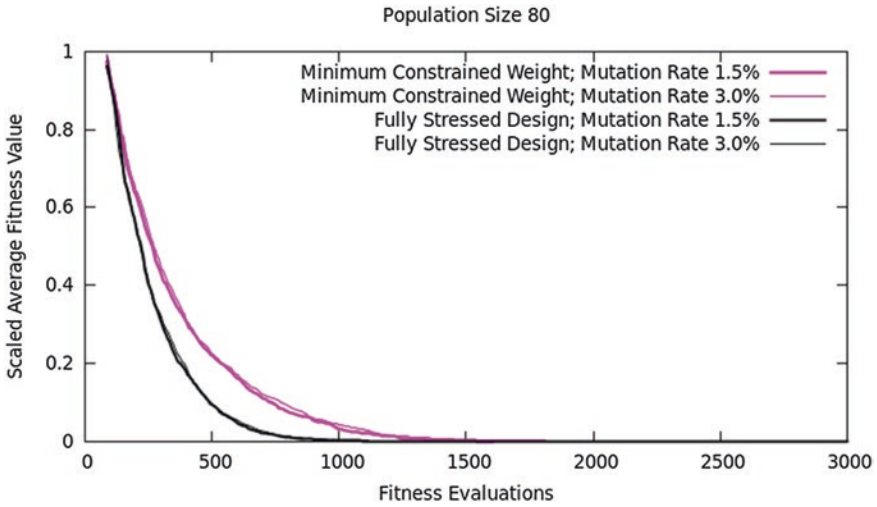

Fig. 2.10 Constrained minimum weight evolution over 100 independent runs in cR00 test case

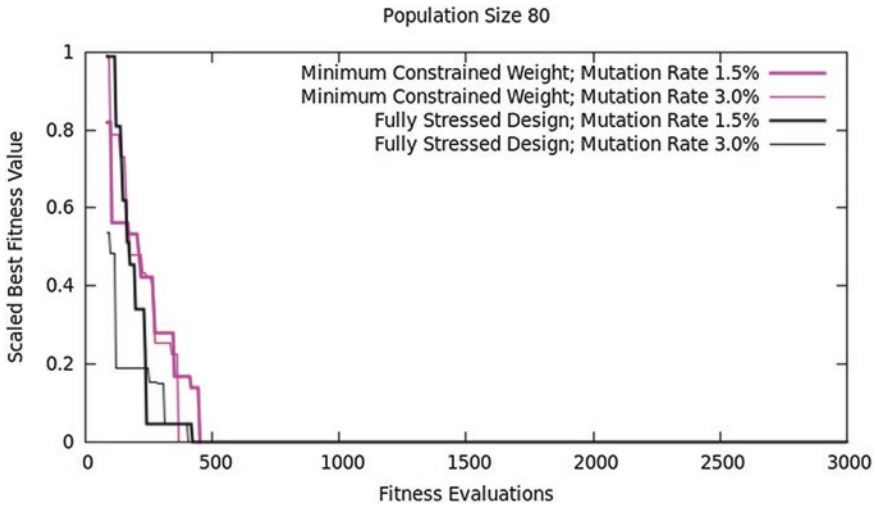

Fig. 2.11 Constrained minimum weight evolution over 100 independent runs in cR00 test case

(d) Comparing FSD and MCW problems with population size 160:
In this case, fitness values have been scaled between 0 and 1, in order to easily compare visually the behaviour of both problems (FSD and MCW). Figure 2.13 shows the scaled average fitness value, Fig. 2.14 shows the scaled best

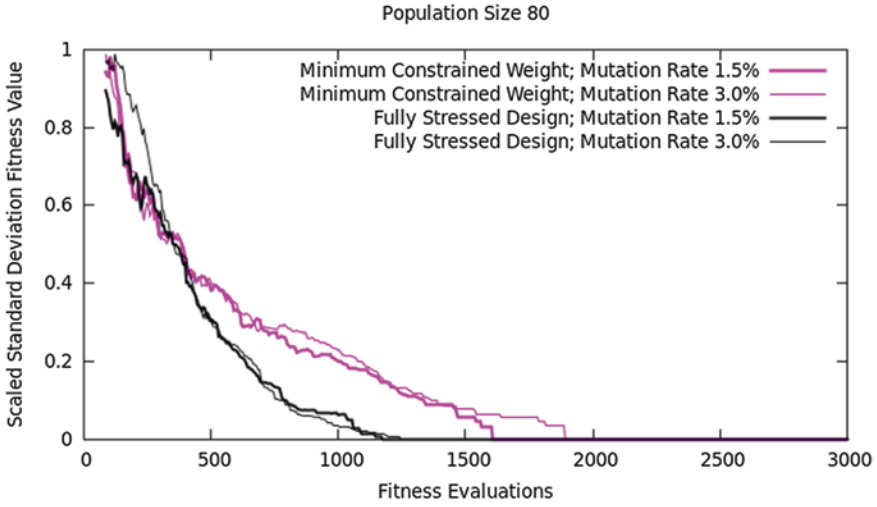

Fig. 2.12 Constrained minimum weight evolution over 100 independent runs in cR00 test case

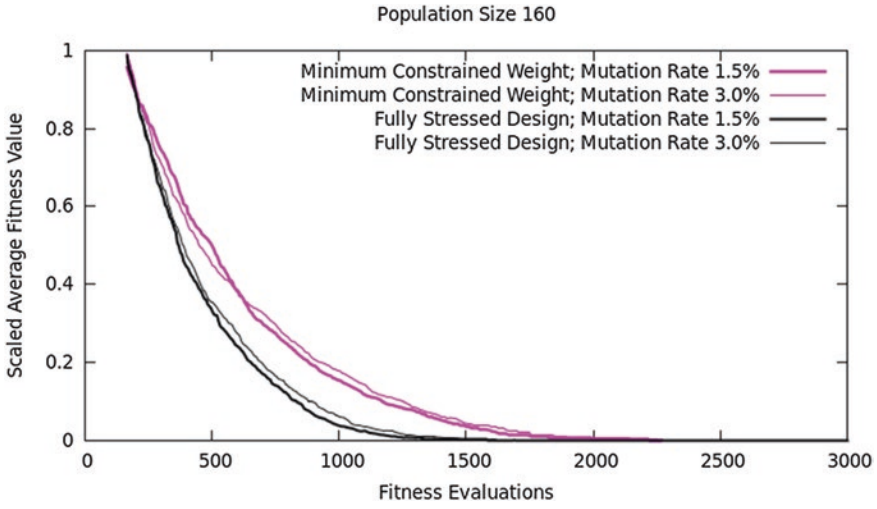

Fig. 2.13 FSD evolution over 100 independent runs in cR00 test case

fitness value and Fig. 2.15 shows the scaled standard deviation fitness value; each containing the FSD optimization problem in black lines and the MCW optimization problem in pink (gray) lines; the thicker line belongs to the 1.5 % mutation rate and the thinner line belongs to the 3.0 % mutation rate.

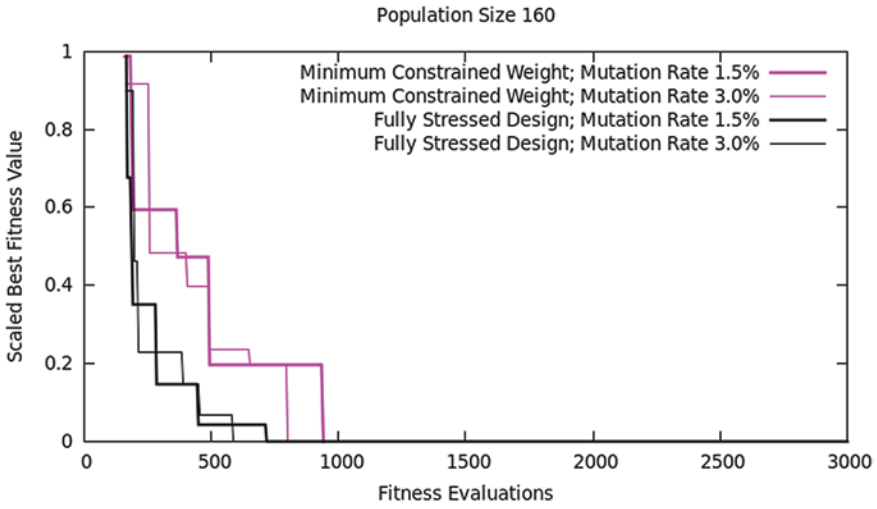

Fig. 2.14 FSD evolution over 100 independent runs in cR00 test case

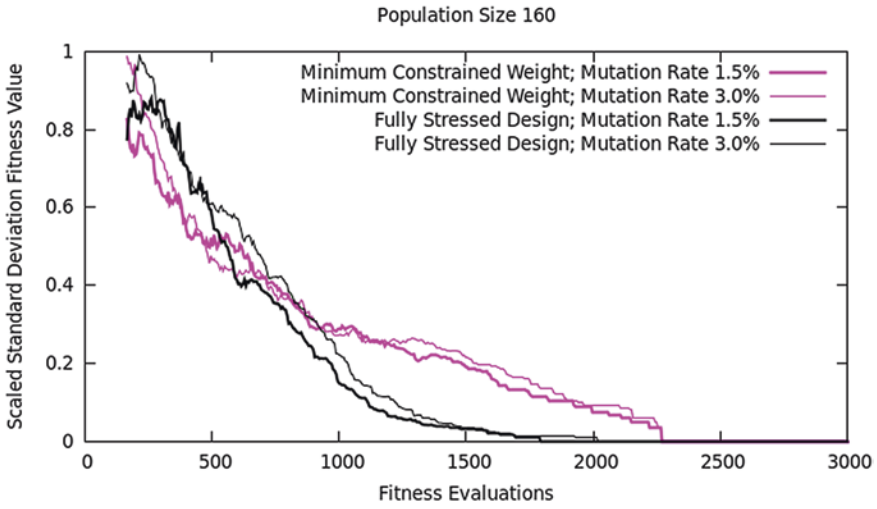

Fig. 2.15 FSD evolution over 100 independent runs in cR00 test case

2.4.3 Discussion

(a) Solving the problem of minimum constrained weight MCW optimum design (Figs. 2.4, 2.5 and 2.6):
The lower the population size (80 vs. 160), the faster the convergence of the algorithm without worsening the quality of the obtained solution in this test case as shown in Figs. 2.4, 2.5 and 2.6 (the chromosome length of this problem is easily

handled by the evolutionary algorithm). The behaviour of average and standard deviation values are slightly better in the lower mutation rate (1.5 % vs. 3 %). It is remarkable that the range of evaluations required to obtain the best design in all the 100 independent executions in this MCW problem vary from 1610 evaluations (population size 80, mutation rate 1.5 %) to 2272 evaluations (population size 160, mutation rates 1.5 and 3 %).

(b) Solving the problem of fully stressed design FSD (Figs. 2.7, 2.8 and 2.9):
Here also, the lower the population size (80 vs. 160), the faster the convergence of the algorithm without worsening the quality of the obtained solution in this test case as shown in Figs. 2.7, 2.8 and 2.9 (the chromosome length of this problem is easily handled by the evolutionary algorithm). As well, the behaviour of average and standard deviation values are slightly better in the lower mutation rate (1.5 % vs. 3 %). It is remarkable that the range of evaluations required to obtain the best design in all the 100 independent executions in this FSD problem vary from 1172 evaluations (population size 80, mutation rate 1.5 %) to 2020 evaluations (population size 160, mutation rate 3 %).

(c) Comparing FSD and MCW problems with population size 80 and 160 (Figs. 2.10, 2.11, 2.12, 2.13, 2.14, and 2.15):
Figures 2.10, 2.11, 2.12, 2.13, 2.14, and 2.15 show clearly through the scaled fitness, that when comparing the fitness behaviour, in all cases of average, best and standard deviation, the FSD problem has a faster convergence than the MCW problem in this test case (except in the best fitness of MCW problem with population size 80 and 3 % mutation rate), requiring lower number of fitness evaluations to achieve the same optimum design.

2.5 Conclusions

The relation of the problems MCW and FSD has been shown through a simple truss test case in discrete cross-section types sizing optimization. This relation has been evidenced by having a coincidental optimum design, a high number of coincidental best designs among the best ones and showing a high correlation coefficient when considering the whole search space of each problem.

The search process of the optimum design of both problems has been possible by using evolutionary algorithms, showing a high robust behaviour where in 100 out of 100 independent runs, this metaheuristic optimization was able to find the best solution in the range of 1000–2000 evaluations -versus a search space of tens of millions-. When comparing them, the fully stressed design (FSD) problem has shown an easier topology for the evolutionary algorithm optimization versus the MCW problem, that is, requiring less number of fitness function evaluations to achieve the same shared best design.

Application of this analysis to an increased number of test cases and generalization in other types of bar structures, -e.g. in the case of frame bar structures-, should be

developed in the future to provide more light about the comparison and relationship of the fully stressed design and the minimum constrained weight problems.

References

1. Coelho RF (2013) Co-evolutionary optimization for multi-objective design under uncertainty. J Mech Design 135–2:1–8
2. Deb K, Bandaru S, Greiner D, Gaspar-Cunha A, Tutum CC (2014) An integrated approach to automated innovization for discovering useful design principles: case studies from engineering. Appl Soft Comp 15:42–56
3. Greiner D, Hajela P (2012) Truss topology optimization for mass and reliability considerations—co-evolutionary multiobjective formulations. Struct Multidiscip O 45–4:589–613
4. Greiner D, Emperador JM, Winter G (2000) Multiobjective optimisation of bar structures by Pareto GA. In: European congress on computational methods in applied sciences and engineering -ECCOMAS, Barcelona, Spain
5. Greiner D, Emperador JM, Winter G (2005) Gray coding in evolutionary multicriteria optimization: application in frame structural optimum design. Evol Multi-Criterion Optim Lect Notes Comput Sci 3410:576–591
6. Greiner D, Emperador JM, Winter G, Galván B (2007) Improving computational mechanics optimum design using helper objectives: an application in frame bar structures. Evol Multi-Criterion Optim Lect Notes Comput Sci 4403:575–589
7. Greiner D, Periaux J, Emperador JM, Galvan B, Winter G (2015) A study of Nash evolutionary algorithms for reconstruction inverse problems in structural engineering. In: Greiner D, et al. (eds) Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences, Springer, pp 321−333
8. Greiner D, Emperador JM, Galvan B, Winter G, Periaux J (2014) Optimum structural design using bio-inspired search methods: a survey and applications. In: Becerra V, Vasile M (eds) Computational intelligence in the aerospace sciences, American Institute of Aeronautics and Astronautics -AIAA
9. Greiner D, Emperador JM, Galvan B, Winter G (2014) A Comparison of minimum constrained weight and Fully Stressed Design Problems in discrete cross-section type bar structures. In: Oñate E, Oliver X, Huerta A (eds) Proceedings of the 11th world congress on computational mechanics WCCM-XI, 5th european congress on computational mechanics ECCM-V, the 6th european congress on computational fluid dynamics ECFD-VI. CIMNE, pp 2064–2072
10. Greiner D, Galván B, Périaux J, Gauger N, Giannakoglou K, Winter G (2015) Advances in evolutionary and deterministic methods for design, optimization and control in engineering and sciences. Computational Methods in Applied Sciences. vol 36, Springer, Berlin
11. Gunnlaugsson G, Martin J (1973) Optimality conditions for fully stressed designs. SIAM J Appl Math 25–3:474–482
12. Lagaros N, Plevris V, Papadrakakis M (2005) Multi-objective design optimization using cascade evolutionary computations. Comput Method Appl M 194(30–33):3496–3515
13. Maxwell JC (1872) On reciprocal figures, frames and diagrams of forces. Trans Roy Soc Edinburgh 26(Plates 1–3):1–40
14. Mueller KM, Liu M, Burns SA (2002) Fully stressed design of frame structures and multiple load paths. J Struct Eng ASCE 128–6:806–814
15. Murotsu Y, Okada H, Taguchi K, Grimmelt M, Yonezawa M (1984) Automatic generation of stochastically dominant failure modes of frame structures. Struct Saf 2:17–25
16. Park S, Choi S, Sikorsky C, Stubbs N (2004) Efficient method for calculation of system reliability of a complex structure. Int J Solids Struct 41:5035–5050

Chapter 3
Multiobjective Approach for Time-Cost Optimization Using a Multi-mode Hybrid Genetic Algorithm

Jorge Magalhães-Mendes

Abstract This paper presents a hybrid genetic algorithm for the time-cost optimization (TCO) problem. The chromosome representation of the problem is based on random keys. The schedules are constructed using a priority rule in which the priorities are defined by the genetic algorithm. Schedules are constructed using a procedure that generates parameterized active schedules. In construction projects, time and cost are the most important factors to be considered. In this paper, a new hybrid genetic algorithm is developed for the optimization of the two objectives time and cost. The approach was developed in VBA and applied to test problems reported on the literature and compared with other approaches. The results indicate that this approach could assist decision-makers to obtain good solutions for project duration and total cost.

Keywords Construction management · Project management · Genetic algorithms · Time-cost optimization

3.1 Introduction and Background

Construction projects are found throughout business and areas such as manufacturing facilities, infrastructure development and improvement, and residential and commercial building.

A construction project is a group of discernible tasks or activities that are conducted in a coordinated effort to accomplish one or more objectives. Construction projects require varying levels of cost, time and other resources (i.e., labor, equipment, material, suppliers) [29].

J. Magalhães-Mendes (✉)
Departamento de Engenharia Civil, Instituto Superior de Engenharia do Porto, Instituto Politécnico do Porto, Porto, Portugal
e-mail: jjm@isep.ipp.pt

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*, Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2_3

In a construction project, there are two main factors, such as project duration and project cost. These are depended to each other. The activity duration is a function of resources (i.e. crew size, equipments and materials) availability. On the other hand, resources demand direct costs. Therefore, the relationship between project time and direct cost of each activity is a monotonously decreasing curve. It means if activity duration is compressed then that leads to an increase in resources and so that direct costs. But, project indirect costs increase with the project duration. In general, for a project, the total cost is the sum of direct and indirect costs and exists an optimum duration for the least cost, see Fig. 3.1. Hence, relationship between project time and cost is trade-off [9]. So, in a project it needs multi-objective approach to minimize both projects time and cost by varying options or modes of construction of the critical activities [28].

As the number of activities of a construction project grows TCO becomes an NP-hard problem. It is an important and challenging problem that has received increasing attention for several years. While the exact methods are available for providing optimal solution for small problems, its computation time is not feasible for large-scale problems. Hence, in practice heuristic and metaheuristic methods to generate near optimal solutions for large problems are of special interest [23].

Several approaches to solve the TCO problem have been proposed in the last years: mathematical, heuristic and search methods.

- Mathematical methods: several mathematical models such as linear programming [15, 16, 27], integer programming, or dynamic programming [3, 5, 6, 32] and LP/IP hybrid [2, 18], Meyer and Shaffer [24] and Patterson and Huber [30] use mixed integer programming. However, for large number of activity in network and complex problem, integer programming needs a lot of computation effort [7]. Since, these are suitable for small project.
- Heuristic methods: heuristic algorithms are not considered to be in the category of optimization methods. They are algorithms developed to find an acceptable near optimum solution. Heuristic methods are usually algorithms easy to

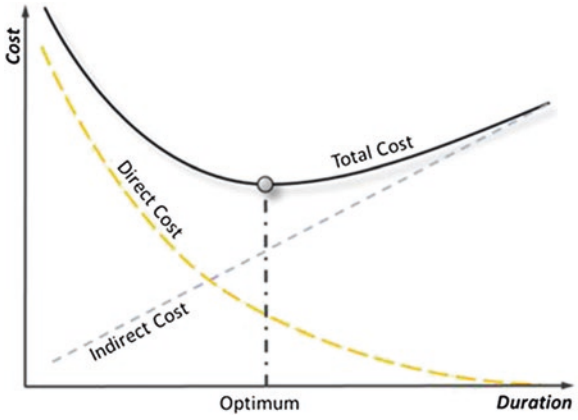

Fig. 3.1 Project time and cost curve (adapted from Golzarpoor [13])

understand which can be applied to larger problems and typically provide acceptable solutions [14]. However, they have lack mathematical consistency and accuracy and are specific to certain instances of the problem [8, 25, 31, 33] are some of the research studies that have utilized heuristic methods for solving TCO problems.
- Search methods: some researchers have tried to introduce evolutionary algorithms to find global optima such as genetic algorithm (GA) [7, 10, 28, 37, 38] the particle swarm optimization algorithm [35], ant colony optimization (ACO) [1, 26, 34] and harmony search (HS) [9].

In this paper it is proposed a new hybrid genetic algorithm based on the works [21, 23], with a novel objective function and a new chromosome structure to solve the time-cost optimization problem.

3.2 Problem Description

The TCO problem can be modeled as follows: a project consists of $n + 2$ activities where each activity has to be processed in order to complete the project. Let $J = \{0, 1, \ldots, n, n + 1\}$ denote the set of activities to be scheduled, $K = \{1, \ldots, k\}$ the set of resources, Cd_{jk} the direct cost of resources type k at activity j and Cid the indirect cost of project. Each resource type k has a limited capacity of R_k at any point in time.

The activities 0 and $n + 1$ are dummy, have no duration and represent the initial and final activities. The activities are interrelated by two kinds of constraints:

- The precedence constraints, which force each activity j to be scheduled after all predecessor activities, Pj, are completed.
- Performing the activities requires resources with limited capacities.

Each activity can be performed in one of several different modes. A mode represents a combination of different resources and/or levels of resources with an associated duration and a direct cost. Once an activity is started in one mode, it may not be changed. One activity j can be executed in m modes given by the set $M_j = \{1, \ldots, m_j\}$. The duration of activity j being performed in mode m_j is given by d_{jm}. The activity j executed in mode m_j uses r_{jmk} units of renewable resource k, where $r_{jmk} \leq R_k$ for each renewable resource k.

While being processed, activity j requires r_{jmk} units of resource type k ε K during every time instant of its non-preemptable duration d_{jm}. The parameters d_{jm}, r_{jmk}, Cd_{jk}, Cid and R_k are assumed to be non-negative and deterministic. Activities are subject to finish-start precedence relations with zero time lags, meaning that an activity can be started if and only if all predecessors have been completed. In addition, the activities in progress are not allowed to be interrupted. Therefore, the goal of solving the TCO is to find a mode combination for all activities, as well as the resultant schedule that leads to minimal project duration and total cost.

3.3 Multiobjective Optimization

With evolutionary techniques being used for single-objective optimization for over two decades, the incorporation of more than one objective in the fitness function has finally gained popularity in the research [4].

In principle, there is no clear definition of an "optimum" in multiobjective optimization (MOP) as in the case of single-objective issues; and there even does not necessarily have to be an absolutely superior solution corresponding to all objectives due to the incommensurability and conflict among objectives. Since the solutions cannot be simply compared with each other, the "best" solution generated from optimization would correspond to human decision-makers subjective selection from a potential solution pool, in terms of their particulars [38].

The classical methods reduce the MOP to a scalar optimization optimization by using multiobjective weighting (MOW) or a utility function (multiobjective utility analysis). Multiobjective weighting allows decisions makers to incorporate the priority of each objective into decision making. Mathematically, the solutions obtained by equally weighting all objectives may provide the least objective conflicts, but in most cases, each objective is first optimized separately and the overall objective value is evaluated depending on the weighting factors. The weakness of MOW is that the overall optimum is usually at the dominating objective only [7].

In a certain way we can say that the work of Zadeh [36] is the first to advocate the assignment of weights to each objective function and combined them into a single-object function. More recently, Gen and Cheng [10] adopted the adaptive weight approach (AWA) in construction TCO problem (also referred to as GC approach hereafter).

In the GC approach Gen and Cheng [10] proposed the following formulas:

$$Z^{+} = \left\{ Z_c^{\max}, Z_t^{\max} \right\} \tag{3.1}$$

$$Z^{-} = \left\{ Z_c^{\min}, Z_t^{\min} \right\} \tag{3.2}$$

where,

$Z_c^{\max}$ maximal value for total cost in the current population;
$Z_t^{\max}$ maximal value for time in the current population;
$Z_c^{\min}$ minimal value for total cost in the current population;
$Z_t^{\min}$ minimal value for time in the current population.

$$w_c = 1 \Big/ \left(Z_c^{\max} - Z_c^{\min}\right), w_t = 1 \Big/ \left(Z_t^{\max} - Z_t^{\min}\right) \tag{3.3}$$

$$f(x) = w_c\left(Z_c^{\max} - Z_c\right) + w_t\left(Z_t^{\max} - Z_t\right) \tag{3.4}$$

In 2004, Zheng et al. [38] proposed the modified weight approach (MAWA) to deal with the multi-objective problem. Under the MAWA, the adaptive weights are formulated through the following four conditions:

1. For $Z_t^{\max} \neq Z_t^{\min}$ and $Z_c^{\max} \neq Z_c^{\min}$

$$v_c = \frac{Z_c^{\min}}{Z_c^{\max} - Z_c^{\min}} \tag{3.5}$$

$$v_t = \frac{Z_t^{\min}}{Z_t^{\max} - Z_t^{\min}} \tag{3.6}$$

$$v = v_c + v_t \tag{3.7}$$

$$w_c = v_c / v \tag{3.8}$$

$$w_t = v_t / v \tag{3.9}$$

$$w_c + w_t = 1 \tag{3.10}$$

2. For $Z_t^{\max} = Z_t^{\min}$ and $Z_c^{\max} = Z_c^{\min}$

$$w_c = w_t = 0.5 \tag{3.11}$$

3. For $Z_t^{\max} = Z_t^{\min}$ and $Z_c^{\max} \neq Z_c^{\min}$

$$w_c = 0.1, w_t = 0.9 \tag{3.12}$$

4. For $Z_t^{\max} \neq Z_t^{\min}$ and $Z_c^{\max} = Z_c^{\min}$

$$w_c = 0.9, w_t = 0.1 \tag{3.13}$$

Zheng et al. [38] proposed a fitness formula in accordance with the proposed adaptive weight:

$$f(x) = w_t \frac{(Z_t^{\max} - Z_t) + \gamma}{(Z_t^{\max} - Z_t^{\min}) + \gamma} + w_c \frac{(Z_c^{\max} - Z_c) + \gamma}{(Z_c^{\max} - Z_c^{\min}) + \gamma} \tag{3.14}$$

where,

γ	is a small positive random number between 0 and 1;
$Z_c^{\max}$	maximal value for total cost in the current population;
$Z_t^{\max}$	maximal value for time in the current population;
$Z_c^{\min}$	minimal value for total cost in the initial population;

$Z_t^{\min}$ minimal value for time in the initial population;
Z_c represents the total cost of the xth solution in current population;
Z_t represents the time of the xth solution in current population.

This study proposes a new fitness formula with the following formulation:

$$f(x) = Gene_t \frac{(Z_t^{\max} - Z_t)}{(Z_t^{\max} - Z_t^{\min})} + Gene_c \frac{(Z_c^{\max} - Z_c)}{(Z_c^{\max} - Z_c^{\min})} \tag{3.15}$$

where,

$Z_c^{\max}$ maximal value for total cost in the current chromosome;
$Z_t^{\max}$ maximal value for time in the current chromosome;
$Z_c^{\min}$ minimal value for total cost in the initial population;
$Z_t^{\min}$ minimal value for time in the initial population;
Z_c represents the total cost of the xth solution in current chromosome;
Z_t represents the time of the xth solution in current chromosome.

3.4 The GA-Based Approach

The approach presented in this paper is based on a genetic algorithm to perform its optimization process. Figure 3.2 shows the architecture of approach.

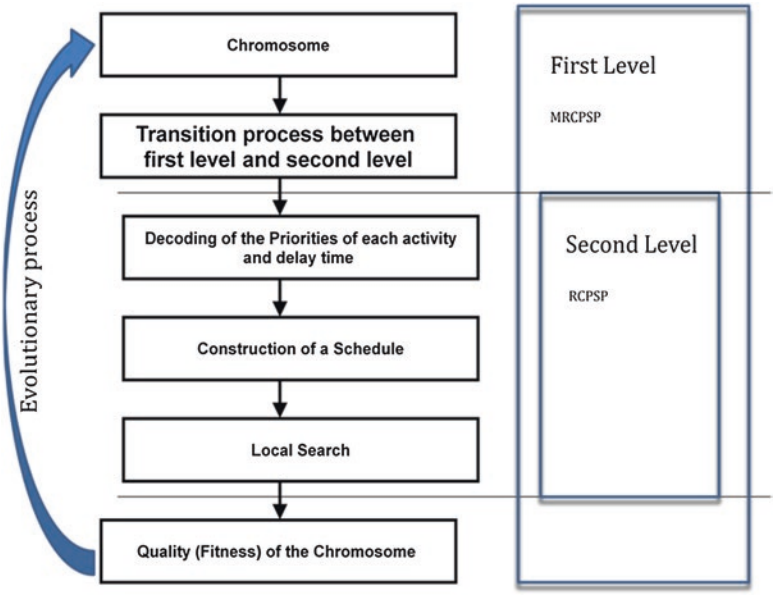

Fig. 3.2 Architecture of the approach

The approach combines a genetic algorithm, a schedule generation scheme and a local search procedure. The genetic algorithm is responsible for evolving the chromosomes which represent the priorities of the activities.

For each chromosome the following four phases are applied:

(1) *Transition parameters*—this phase is responsible for the process transition between first level and second level;
(2) *Schedule parameters*—this phase is responsible for transforming the chromosome supplied by the genetic algorithm into the priorities of the activities and delay time;
(3) *Schedule generation*—this phase makes use of the priorities and the delay time and constructs schedules;
(4) *Schedule improvement*—this phase makes use of a local search procedure to improve the solution obtained in the schedule generation phase.

After a schedule is obtained, the quality is processed feedback to the genetic algorithm. Figure 3.2 illustrates the sequence of phases applied to each chromosome. Details about each of these phases will be presented in the next sections.

3.4.1 GA Transition Process

The evolutionary algorithms are an interdisciplinary research area comprising several paradigms inspired by Darwinian principle of evolution.

The current stage of research considers, among others, the following paradigms: genetic algorithms, genetic programming, evolutionary strategies, neuroevolution and differential evolution.

The Genetic Algorithms (GAs) have been applied successfully in several areas, such as bioinformatics, computational science, engineering, economics, chemistry, manufacturing, mathematics and physics.

The GAs are search algorithms which are based on the mechanics of natural selection and genetics to search through decision space for optimal solutions. One fundamental advantage of GAs versus traditional methods is described by Goldberg [12]: in many optimization methods, we move gingerly from a single solution in the decision space to the next using some transition rule to determine the next solution.

First of all, an initial population of potential solutions (individual) is generated randomly. A selection procedure based on a fitness function enables to choose the individual candidate for reproduction. The reproduction consists in recombining two individuals by the crossover operator, possibly followed by a mutation of the offspring. Therefore, from the initial population a new generation is obtained. From this new generation, a second new generation is produced by the same process and so on. The stopping criterion is normally based on the number of generations.

The GA based-approach uses a random key alphabet U (0, 1) and an evolutionary strategy identical to the one proposed by Goldberg [12].

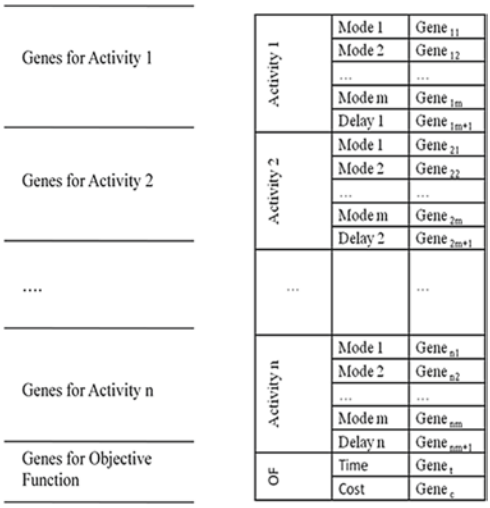

Fig. 3.3 Chromosome structure

Each chromosome represents a solution to the problem and it is encoded as a vector of random keys (random numbers). Each solution encoded as *initial chromosome* (first level) is made of $2 + mn + n$ genes where n is the number of activities and m is the number of execution modes, see Fig. 3.3.

The called first level as the capacity to solving the multi-mode resource constrained project scheduling problem (MRCPSP) [21, 23].

In this case of study we do not consider the requirements to the type and number of resources needed for construction mode for each activity as well as the maximum number of available resources.

The transition process between first level and second level consists in choosing the option or construction mode m_j for each activity j. Using this process we obtain the solution chromosome (second level) composed by $2*(n + 1)$ genes. In general, for a project, the total cost is the sum of direct and indirect costs and exists an optimum duration for the least cost, see Fig. 3.4.

The called second level as the capacity to solving the resource constrained project scheduling problem (RCPSP) [21, 23]. In this case of study we do not consider the requirements to the type and number of resources needed for each activity as well as the maximum number of available resources.

After, we evaluate the quality (fitness) of the solution chromosome.

3.4.2 GA Decoding

A real-coded GA is adopted in this article. Compared with the binary-code GA, the real-coded GA has several distinct advantages, which can be summarized as follows [19]:

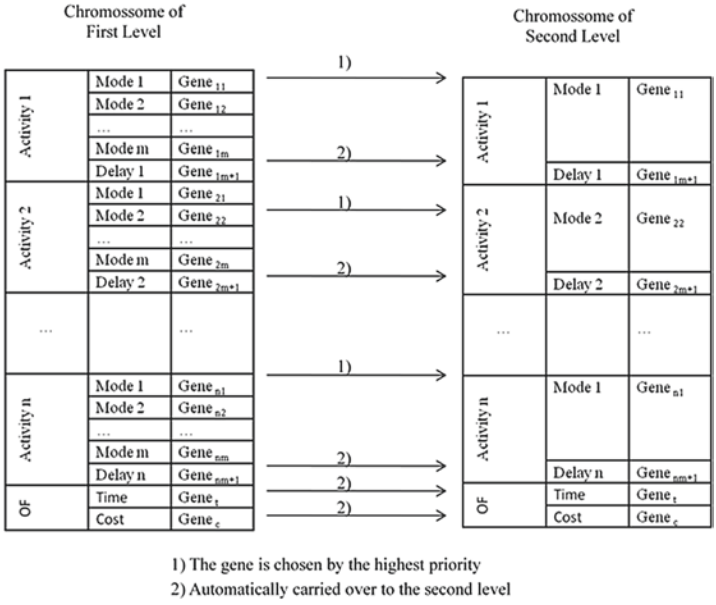

Fig. 3.4 Transition process between first and second level

- It is more convenient for the real-coded GA to denote large scale numbers and search in large scope, and thus the computation complexity is amended and the computation efficiency is improved;
- The solution precision of the real-coded GA is much higher than that of the binary-coded GA;
- As the design variables are coded by floating numbers in classical optimization algorithms, the real-coded GA is more convenient for combination with classical optimization algorithms.

The priority decoding expression uses the following expression:

$$PRIORITY_j = \frac{LLP_j}{LCP} \times \left[\frac{1 + gene_{jm}}{2}\right] \quad j = 1, \ldots, n \tag{3.16}$$

where,

LLP_j is the longest length path from the beginning of the activity j to the end of the project;
LCP is the length along the critical path of the project [20];
mj is the gene of the selected mode for activity j.

The gene $jm + 1$ is used to determine the delay time when scheduling the activities. The delay time used by each activity is given by the following expression:

$$Delay\,time = gene_{jm+1} \times 1.5 \times MaxDur \tag{3.17}$$

where *MaxDur* is the maximum duration of all activities. The factor 1.5 is obtained after some experimental tuning.

A maximum delay time equal to zero is equivalent to restricting the solution space to non-delay schedules and a maximum delay time equal to infinity is equivalent to allowing active schedules. To reduce the solution space is used the value given by formula (3.17), see Gonçalves et al. [11].

3.4.3 Construction of a Schedule

Schedule generation schemes (SGS) are the core of most heuristic solution procedures for project scheduling. SGS start from scratch and build a feasible schedule by stepwise extension of a partial schedule. A partial schedule is a schedule where only a subset of the $n + 2$ activities have been scheduled.

There are two different classics methods SGS available. They can be distinguished into activity and time incrementation. The so called serial SGS performs activity-incrementation and the so called parallel SGS performs time-incrementation.

A third method for schedule generating can be applied: the parameterized active schedules. This type of schedule consists of schedules in which no resource is kept idle for more than a predefined period if it could start processing some activity. If the predefined period is set to zero, then we obtain a non-delay schedule. This type of SGS is used on this work.

Figure 3.5 presents the relationship diagram of various schedules with regard to optimal schedules.

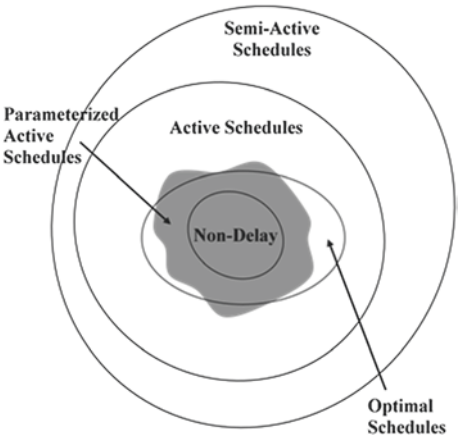

Fig. 3.5 Types of schedules (adapted from Magalhães-Mendes [22])

3.4.4 Local Search

Local search algorithms move from solution to solution in the space of candidate solutions (the search space) until a solution optimal or a stopping criterion is found. In this paper it is applied backward and forward improvement based on Klein [17].

Initially it is constructed a schedule by planning in a forward direction starting from the project's beginning. After it is applied backward and forward improvement trying to get a better solution. The backward planning consists in reversing the project network and applying the scheduling generator scheme. A detailed example is described by Magalhães-Mendes [20].

3.4.5 Evolutionary Strategy

There are many variations of genetic algorithms obtained by altering the reproduction, crossover, and mutation operators. Reproduction is a process in which individual (chromosome) is copied according to their fitness values (makespan). Reproduction is accomplished by first copying some of the best individuals from one generation to the next, in what is called an elitist strategy.

In this paper the fitness proportionate selection, also known as roulette-wheel selection, is the genetic operator for selecting potentially useful solutions for reproduction. The characteristic of the roulette wheel selection is stochastic sampling.

The fitness value is used to associate a probability of selection with each individual chromosome. If f_i is the fitness of individual i in the population, its probability of being selected is,

$$p_i = \frac{f_i}{\sum_{i=1}^{N} f_i}, \quad i = 1, \ldots, n \tag{3.18}$$

A roulette wheel model is established to represent the survival probabilities for all the individuals in the population. Then the roulette wheel is rotated for several times [12].

After selecting, crossover may proceed in two steps. First, members of the newly selected (reproduced) chromosomes in the mating pool are mated at random. Second, each pair of chromosomes undergoes crossover as follows: an integer position k along the chromosome is selected uniformly at random between 1 and the chromosome length l. Two new chromosomes are created swapping all the genes between $k + 1$ and l, see Magalhães-Mendes [21].

The mutation operator preserves diversification in the search. This operator is applied to each offspring in the population with a predetermined probability. We assume that the probability of the mutation in this paper is 5 %.

3.5 Computational Experiments

To illustrate the effectiveness of the proposed RKV-TCO (Random Key Variant for Time-Cost Optimization) approach, a project of seven activities proposed initially by Liu et al. [18] and a project of 18 activities originally introduced by Feng et al. [7] are used.

This computational experience has been performed on a computer with an Intel Core 2 Duo CPU T7250 @2.33 GHz and 1.95 GB of RAM. The algorithm proposed in this work has been coded in VBA under Microsoft Windows NT.

3.5.1 First Case

A project of seven activities proposed initially by Liu et al. [18] and fitted by Zheng et al. [38] is presented in Table 3.1 with available activity options and corresponding durations and costs. Indirect cost rate was $1500/day.

The robustness of the new proposed model RKV-TCO in the deterministic situation was compared with three other previous models:

(1) Gen and Cheng [10] using GC approach;
(2) Zheng et al. [38] using MAWA with a GA-based approach;
(3) Parveen and Saha [28] using MATLAB 7.7.0.471 (R2008b).

The results of RKV-TCO approach are presented in Table 3.2. The Table 3.2 shows the values of time and cost for the first six generations with Gen and Cheng [10] and Zheng et al. [38] approaches. The algorithm RKV-TCO obtains in the sixth generation a better solution than the works mentioned above. Furthermore, the solutions obtained in generations 6, 7 and 10 are solutions of Pareto front, which proves the effectiveness of this algorithm. The Pareto front solutions are reported by the work [28]. The RKV-TCO ends with an optimal solution (project time = 60 days and cost = $233,500) in Table 3.2.

We can say that the RKV-TCO algorithm has better results than Gen and Cheng [10] and Zheng et al. [11] and compares very well with the work of Parveen and Saha [28] who used MATLAB.

Additionally we can also state that the RKV-TCO approach produces high-quality solutions quickly once needed only 2 s to complete 50 generations.

3.5.2 Second Case

The second case of study is a project of eighteen activities originally introduced by Feng et al. [7]. The activity relationships for the model project consisting of 18 activities and the three modes of construction for each activity and their associated time and cost are presented in Table 3.3. Indirect cost rate was $1000/day.

Table 3.1 Time and cost for each option/mode of activity

Activity description	Activity number	Precedent activity	Option/ Mode	Duration (days)	Direct cost ($)
Site preparation	1	–	1	14	23000
			2	20	18,000
			3	24	12,000
Forms and rebar	2	1	1	15	3,000
			2	18	2,400
			3	20	1,800
			4	23	1,500
			5	25	1,000
Excavation	3	1	1	15	4,500
			2	22	4,000
			3	33	3,200
Precast concrete girder	4	1	1	12	45,000
			2	16	35,000
			3	20	30,000
Pour foundation and piers	5	2, 3	1	22	20,000
			2	24	17,500
			3	28	15,000
			4	30	10,000
Deliver PC girders	6	4	1	14	40,000
			2	18	32,000
			3	24	18,000
Erect girders	7	5, 6	1	9	30,000
			2	15	24,000
			3	18	22,000

The Table 3.4 shows the results for several mathematical and evolutionary-based methods. The algorithm RKV-TCO obtains better solution than the other GA-based approaches. Furthermore, the algorithm RKV-TCO reaches the *optimal* solution quickly, i.e., in 5 s.

The results of the RKV-TCO illustrates that evolutionary methods based on genetic algorithms can obtain the better solutions and in very reasonable computational time. The time necessary by RKV-TCO to obtain the optimal solution is highly promising and shows that a good implementation can be critical to the success of the genetic algorithms. Also important to emphasize is that genetic algorithms can solve large problems as opposed to exact methods.

3.5.3 GA Configuration

Though there is no straightforward way to configure the parameters of a genetic algorithm, we obtained good results with values: population size of 5 × number

Table 3.2 Summary of the results

Approaches	Generation number	Criteria		Calculation time
		Time	Cost ($)	
Gen and Cheng [10]	0	83	243,500	Not reported
	1	80	242,400	
	2	80	261,900	
	3	79	256,400	
	4	79	256,400	
	5	79	256,400	
Zheng et al. [38]	0	73	251,500	Not reported
	1	73	251,500	
	2	73	251,500	
	3	66	236,500	
	4	66	236,500	
	5	66	236,500	
Parveen and Saha [28] (using MATLAB)	Not reported	60	233,500	Not reported
This paper	0	73	242,300	2 (two) s for 50 generations
	1	73	228,300	
	2	67	225,300	
	3	67	230,300	
	5	65	227,900	
	6	63	225,500	
	7	62	233,000	
	10	60	233,500	

of activities in the problem; mutation probability of 0.05; top (best) 1 % from the previous population chromosomes are copied to the next generation; stopping criterion of 50 generations.

3.6 Conclusions and Further Research

A new GA based-approach to solving the time-cost optimization problem has been proposed. The project activities have various construction modes, which reflect different ways of performing the activity, each mode having a different impact on the duration and cost of the project. This approach combines a genetic algorithm, a schedule generation scheme and a local search with a new fitness formula. The chromosome representation of the problem is based on random keys. The schedules are constructed using a priority rule in which the priorities are defined by the genetic algorithm. Schedules are constructed using a procedure that generates parameterized active schedules. The present approach provides an attractive alternative for the solution of the construction multi-objective optimization problems.

Table 3.3 Time and cost for each option/mode of activity (adapted from Golzarpoor [13])

Activity number	Precedent activity	Option/Mode 1		Option/Mode 2		Option/Mode 3	
		Duration (days)	Direct cost ($)	Duration (days)	Direct cost ($)	Duration (days)	Direct cost ($)
1	–	24	1,200	21	1,500	14	2,400
2	–	25	1,000	23	1,500	15	3,000
3	–	33	3,200	33	3,200	15	4,500
4	–	20	30,000	20	30,000	12	45,000
5	1	30	10,000	30	10,000	22	20,000
6	1	24	18,000	24	18,000	14	40,000
7	5	18	22,000	18	22,000	9	30,000
8	6	24	120	21	208	14	220
9	6	25	100	23	150	15	300
10	2, 6	33	320	33	320	15	450
11	7, 8	20	300	20	300	12	450
12	5, 9, 10	30	1,000	30	1,000	22	2,000
13	3	24	1,800	24	1,800	14	4,000
14	4, 10	18	2,200	18	2,200	9	3,000
15	12	16	3,500	16	3,500	12	4,500
16	13, 14	30	1,000	28	1,500	20	3,000
17	11, 14, 15	24	1,800	24	1,800	14	4,000
18	16, 17	18	2,200	18	2,200	9	3,000

Table 3.4 Summary of the results for mathematical and evolutionary-based methods

Approaches	Deviation (%)[b]	Criteria		Calculation time
		Time	Cost ($)	
Optimal solution	0	110	216,270	–
Excel solver[a]	18	110	254,620	2 min
Risk solver platform standard SLGRG nonlinear[a]	0	110	216,270	1.5 min
Risk solver platform standard large-scale GRG solver[a]	0	110	216,270	1.5 min
TCT optimization using evolver (includes an evolutionary engine)[a]	10	110	238,070	30 min
Risk solver platform standard evolutionary solver[a]	27	110	275,320	18 min
Optimization results using CPLEX CP optimizer[a]	0	110	216,270	9 min
IBM ILOG optimization studio[a]	0	110	216,270	9 min
This paper (RKV-TCO)	0	110	216,270	5 (five) s for 50 generations

[a]Reported by Golzarpoor [13]
[b]Percentage of deviation of the result from optimal solution

Further research can be extended to more construction project problems to reinforce the results obtained namely expanding the optimization model to consider resource allocation and resource leveling constraints and test the use of contrasted multiobjective evolutionary algorithms of second generation (e.g.: NSGA2) in these problems with the proposed approach.

Acknowledgments This work has been partially supported by the CIDEM (Centre for Research & Development in Mechanical Engineering). CIDEM is an investigation unit of FCT—Portuguese Foundation for the Science and Technology located at the Instituto Superior de Engenharia do Instituto Politécnico do Porto.

References

1. Afshar A, Ziaraty A, Kaveh A, Sharifi F (2009) Nondominated archiving multicolony ant algorithm in time-cost trade-off optimization. J Constr Eng Manage 135(7):668–674
2. Burns SA, Liu L, Feng CW (1996) The LP/IP hybrid method for construction time-cost trade-off analysis. Constr Manage Econ 14(3):265–276
3. Butcher WS (1967) Dynamic programming for project cost-time curve. J Constr Div ASCE 93(C01):59–73
4. Coello CC (2000) An updated survey of GA-based multiobjective optimization techniques. ACM Comput Surv 32(2):109–143
5. De P, Dunne EJ, Ghosh JB, Wells CE (1995) The discrete time-cost trade-off problem revisited. Eur J Oper Res 81(2):225–238
6. Elmaghraby SE (1993) Resource allocation via dynamic programming in activity networks. Eur J Oper Res 64:199–215
7. Feng CW, Liu L, Burns SA (1997) Using genetic algorithms to solve construction time-cost trade-off problems. J Comput Civ Eng ASCE 11(3):184–189
8. Fondahl JW (1961) A non-computer approach to the critical path method for the construction industry. Technical report no. 9, The Construction Institute, Department of Civil Engineering, Stamford University
9. Geem ZW (2010) Multiobjective optimization of time-cost trade-off using harmony search. J Constr Eng Manage ASCE 136(6):711–716
10. Gen M, Cheng R (2000) Genetic algorithms and engineering optimization. Wiley-Interscience, New York
11. Gonçalves JF, Magalhães-Mendes JJ, Resende MCG (2005) A hybrid genetic algorithm for the job shop scheduling problem. Eur J Oper Res 167:77–95
12. Goldberg DE (1989) Genetic algorithms in search, optimization and machine learning. Addison-Wesley, Reading
13. Golzarpoor B (2012) Time-cost optimization of large-scale construction projects using constraint programming. Master thesis, University of Waterloo, Ontario, Canada, 2012
14. Hegazy T (1999) Optimization of construction time-cost trade-off analysis using genetic algorithms. Can J Civ Eng 26(6):685–697
15. Hendrickson C, Au T (1989) Project management for construction: fundamental concepts for owners, engineers, architects, and builders. Prentice-Hall International Series in Civil Engineering and Engineering Mechanics
16. Kelley JE (1961) Critical-path planning and scheduling: mathematical basis. Oper Res 9(3):296–320
17. Klein R (2000) Bidirectional planning: improving priority rule-based heuristics for scheduling resource constrained projects. Eur J Oper Res 127:619–638

18. Liu L, Burns S, Feng C (1995) Construction time-cost trade-off analysis using LP/IP hybrid method. J Constr Eng Manage 121(4):446–454
19. Luo YZ, Tang GJ, Wang ZG, Li HY (2006) Optimization of perturbed and constrained fuel-optimal impulsive rendezvous using a hybrid approach. Eng Optim 38(8):959–973
20. Magalhães-Mendes J (2008) Project scheduling under multiple resources constraints using a genetic algorithm. WSEAS Trans Bus Econ World Sci Eng Acad Soc USA 11:487–496
21. Magalhães-Mendes J (2011) A two-level genetic algorithm for the multi-mode resource-constrained project scheduling problem. Int J Syst Appl Eng Dev 5(3):271–278
22. Magalhães-Mendes J (2011) Active, parameterized active, and non-delay schedules for project scheduling with multi-modes. In: Proceedings of the 16th WSEAS international conference on applied mathematics, Montreaux, Switzerland, 29–31 Dec 2011, pp 134–139
23. Magalhães-Mendes J (2012) Hybrid genetic algorithm for the multi-mode resource-constrained project scheduling. In: Proceedings of the 6th European congress on computational methods in applied sciences and engineering (ECCOMAS 2012) e-Book full papers. Vienna University of Technology, Austria, pp 8871–8882
24. Mayer WL, Shaffer LR (1963) Extension of the critical path method through the application of integer programming. Civil Engineering Construction Research Series 2, University of Illinois, Urbana, III
25. Moselhi O (1993) Schedule compression using the direct stiffness method. Can J Civ Eng 20(1):65–72
26. Ng ST, Zhang Y (2008) Optimizing construction time and cost using ant colony optimization approach. J Constr Eng Manage ASCE 134(9):721–728
27. Pagnoni A (1990) Project engineering: computer-oriented planning and operational decision making. Springer, New York
28. Parveen S, Saha SK (2012) GA based multi-objective time-cost optimization in a project with resources consideration. Int J Modern Eng Res (IJMER) 2(6):4352–4359
29. Patrick C (2004) Construction project planning and scheduling. Pearson Prentice Hall, Ohio
30. Patterson JH, Huber D (1974) A horizon-varying, zero-one approach to project scheduling. Manage Sci 20(6):990–998
31. Prager W (1963) A structured method of computing project cost polygons. Manage Sci 9(3):394–404
32. Robinson DR (1975) A dynamic programming solution to cost-time tradeoff for CPM. Manage Sci 22(2):158–166
33. Siemens N (1971) A simple CPM time-cost trade-off algorithm. Manage Sci 17(6):B-354–B-363
34. Xiong Y, Kuang Y (2008) Applying an ant colony optimization algorithm-based multiobjective approach for time–cost trade-off. J Constr Eng Manage ASCE 134(2):153–156
35. Yang IT (2007) Using elitist particle swarm optimization to facilitate bicriterion time-cost trade-off analysis. J Constr Eng Manage ASCE 133(7):498–505
36. Zadeh LA (1965) Fuzzy sets. Inf Control 8:338–353
37. Zheng DXM, Ng ST (2005) Stochastic time–cost optimization model incorporating fuzzy sets theory and nonreplaceable front. J Constr Eng Manage ASCE 131(2):176–186
38. Zheng DXM, Ng ST, Kumaraswamy MM (2004) Applying a genetic algorithm-based multiobjective approach for time-cost optimization. J Constr Eng Manage 130(2):168–176

Chapter 4
On Efficiency Increase of Evolutionary Algorithms for Large Non-linear Constrained Optimization Problems with Applications to Mechanics

Janusz Orkisz and Maciej Glowacki

Abstract Advances in development of highly efficient dedicated Evolutionary Algorithms (EA) for a wide class of large non-linear constrained optimization problems are considered in this paper. The first objective of this general research is development and application of the improved EA to residual stress analysis in railroad rails and vehicle wheels. However, the standard EA are not sufficiently efficient for solving such large optimization problems. Therefore, our current research is mostly focused on development of various new very efficient acceleration techniques proposed, including smoothing and balancing, adaptive step-by-step mesh refinement, as well as a'posteriori error analysis and related techniques. This paper presents an efficiency analysis of chosen speed-up techniques using several simple but demanding benchmark problems, including residual stress analysis in elastic-plastic bodies under cyclic loadings. Preliminary results obtained for numerical tests are encouraging and show a clear possibility of practical application of the improved EA to large optimization problems.

Keywords Evolutionary algorithms · Large non-linear constrained optimization · Computation efficiency increase

J. Orkisz · M. Glowacki (✉)
Institute for Computational Civil Engineering, Cracow University of Technology,
24 Warszawska Street, 31-155 Kraków, Poland
e-mail: mglowac@gmail.com

J. Orkisz
e-mail: plorkisz@cyf-kr.edu.pl

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*,
Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2_4

4.1 Introduction

A lot of scientific and engineering problems, including many important problems of mechanics, may be formulated in terms of constrained optimization. Complexity of such problems results mostly from their non-linearity, as well as from a large number of decision variables and constraints. Thus, this paper considers development of an efficient optimization approach based on the Evolutionary Algorithms (EA) for a wide class of large non-linear constrained optimization problems.

In contrast to most deterministic methods, the EA may be successfully applied with similar efficiency to both the convex and non-convex problems [4, 10]. However, general efficiency of the standard EA is rather low. Therefore, significant acceleration of the convergence process is needed. Moreover, the improved EA should provide possibility of solving such optimization problems, when the standard EA fail. Improvement of the standard EA may be obtained in several ways. We have already proposed several new acceleration techniques [5, 15, 16, 18]. These techniques have been preliminarily tested using several demanding benchmark problems. Numerical results of these tests indicate significant acceleration of the large optimization processes involved.

The engineering objective of our research includes residual stress analysis in railroad rails and vehicle wheels [7, 12, 13], as well as a wide class of problems resulting from the Physically Based Approximation (PBA) of experimental and/or numerical data [8]. Tensile residual stresses are of great importance in reliable prediction of rail and wheel life service resulting from its fatigue failure. Both the theoretical and experimental investigations of residual stress may be expressed in terms of constrained optimization problems. Theoretical model of residual stress analysis in bodies under cyclic loadings is based on the shakedown theory and may be found in [12–14]. Several discrete methods (Finite Element Method, Boundary Element Method, Meshless Finite Difference Method) and the deterministic solution approach were already used to solve such problems [13, 14]; neural networks were also investigated. The experimental model is based on the PBA approach [8, 13]. In general, the PBA may be applied for smoothing of any experimentally measured data. It allows for simultaneous use of the whole experimental, theoretical, and heuristic knowledge of analyzed problems in a way dependent on the reliability of such information [8]. The PBA may be also applied for smoothing of discrete data obtained from any rough numerical solution of any boundary value problem. So far mostly the deterministic methods have been used for solving the PBA problems [8, 13]. However, preliminary attempts of application of the EA to such problems have been also recently made in [17].

Due to the size and complexity of the considered optimization problems, our research is focused, first of all, on the efficiency increase of the algorithms applied. We are presenting here the state of the art of our research, including overview of the proposed acceleration techniques, advances in their development, and chosen numerical results carried out for various benchmarks problems. The present work is a continuation of the previous papers [5, 15–18].

4.2 General Problem Formulation

Considered is a wide class of large non-linear constrained optimization problems. Usually such problems are formulated as optimization of functionals, where a function $u(\mathbf{x})$, $\mathbf{x} \in R^N$ is sought, usually in the discrete form of the vector $\mathbf{u} = \{u_i\}$ consisting of nodal values u_i, $i = 1, 2, \ldots, n$. These nodal values are defined on a mesh formed by arbitrarily distributed nodes. Here, N is the dimension of the physical space (1D, 2D or 3D), and n is a number of decision variables. In general, considered optimization problems may be posed as follows:

find a function $u = u(\mathbf{x})$, that yields the stationary point of a functional $\Phi(u)$, satisfying the equality

$$\mathbf{A}(u) = 0 \tag{4.1}$$

and inequality constraints

$$\mathbf{B}(u) \leq 0 \tag{4.2}$$

In particular case of the PBA approach [8], the functional

$$\Phi = \lambda\, \Phi^E + (1 - \lambda)\, \Phi^T, \quad \lambda \in [0, 1] \tag{4.3}$$

consists of the experimental $\Phi^E(\sigma)$ and theoretical $\Phi^T(\sigma)$ parts, scaled to be dimensionless quantities. Here, σ is the required solution, and λ is a scalar weighting factor. In the PBA, the equality constraints are usually of theoretical nature, while the inequality ones are mostly of experimental nature.

The experimental part of the functional is defined as the weighted averaged error resulting from discrepancies between the measured data and its approximation [8]:

$$\Phi^E(\sigma) = \frac{1}{m} \sum_{i=1}^{m} F\left(\frac{f(\sigma(r_i)) - f_i^{\exp}}{e_i} \right) \tag{4.4}$$

where σ represents the required unknown field, f is a measured function of σ, $f_i^{\exp}$ is its experimental value at the point r_i, e_i is an admissible experimental error, m is a number of measurements, $F(x) = p(\bar{x}) - p(x - \bar{x})$ is a data scattering function defined by the probability density function $p(x - \bar{x})$, and $\bar{x}$ is the expected value.

The enhanced field $\sigma(r)$ cannot differ too much from experimental data. Thus, the inequality constraints are defined as local requirements:

$$\left| f(\sigma(r_i)) - f_i^{\exp} \right| \leq e_i, \quad i = 1, 2, 3, ..., m \tag{4.5}$$

It is useful to impose also an averaged global constraint:

$$\sqrt{\Phi^E} \leq e_E \tag{4.6}$$

Admissible experimental errors e_E and e_i, $i = 1, 2, 3, \ldots, m$ should be evaluated taking into account the true statistics of measurements.

The theoretical part of the functional (4.3) is based on a known theory, and/or on heuristic principles [8]. In mechanics it may be represented by an energy

functional that has to be minimized, e.g., the total complementary energy of statically admissible stresses. On the other hand, as a heuristic principle, e.g., requirement of smoothness may be also introduced. In such case, the minimal average curvature κ in the whole domain Ω can be used, hence

$$\Phi^T = \frac{1}{\Omega} \int_{\Omega} \kappa^2 d\Omega \tag{4.7}$$

where

$$\kappa^2 = \frac{1}{2\pi} \int_0^{2\pi} \left(\frac{\partial^2 f}{\partial \nu^2} \right)^2 d\varphi \tag{4.8}$$

In the 2D Cartesian coordinate system the above definition may be replaced by the following one:

$$\kappa^2(f) = \tfrac{1}{4}(f_{xx} + f_{yy})^2 + \tfrac{1}{8}(f_{xx} - f_{yy})^2 + \tfrac{1}{2} f_{xy}^2 \tag{4.9}$$

and transformed to the polar coordinate system if necessary.

One of the main difficulties in the general formulation is the problem of how to establish the weighting factor λ, i.e., how to determine a reasonable balance between experiment and theory involved. Specific formulations addressing this problem may be found in [8].

4.3 Evolutionary Algorithms and Acceleration Techniques

Nowadays, the EA form a wide group of biologically inspired methods based on theory of evolution and genetics. This group includes such methods like genetic algorithms, genetic programming, evolutionary strategies, evolutionary programming, and others [4, 10]. In this paper, the EA are precisely understood as genetic algorithms with decimal (floating-point) chromosomes. The standard algorithm consists of three operators: selection, crossover and mutation [4]. Significant acceleration of the EA-based solution approach may be achieved in various ways, including appropriate hardware, software, and algorithm improvements.

Hardware acceleration techniques include distribution and parallelization of calculations on various parallel architectures, e.g., general-purpose Graphics Processing Units (GPUs), Field-Programmable Gate Array (FPGA) devices, or standard computer clusters. Efficient software implementations dedicated for particular hardware architectures are crucial as well. Many various parallel EA have been already developed and tested [9, 11]. Algorithmic acceleration of an optimization process may be obtained by, e.g., development of hybrid algorithms

[2, 6] combining the EA with deterministic methods (such as feasible direction method), and introduction of new, problem-oriented operators.

Our approach includes three ways for the EA speed-up. At first, a choice of the most efficient combination of particular selection, crossover, and mutation operators was sought, out of a variety of available ones. Three types of classification rules were proposed and applied for this purpose. Evaluation of the best values of EA parameters (like population size, probability of mutation and crossover) was done as well. Later on, we have proposed and preliminarily investigated several new acceleration techniques based on simple concepts. These techniques include smoothing and balancing [15, 16], a'posteriori error analysis and related techniques (like solution averaging and cloning, creating population of representatives) [18], as well as adaptive step-by-step mesh refinement [5, 16], and possible combinations of the above. Proposed techniques are well supported by non-standard use of parallel and distributed calculations. Some of them are problem- (or class of problems) oriented, other are of more general nature. Some of these techniques are addressed to optimization of functionals, where a large set of nodal values of a function is searched. Appropriate constraint handling [3] is also very important, especially in the case of optimization problems involving large number of inequality constraints. Therefore, we have paid particular attention to investigation of various penalty functions for constraint handling and their impact on the convergence rate of the optimization process. Finally, we consider application and further development of chosen well-known acceleration techniques, such as standard distribution and parallelization of computations, hybrid approach, and use of other evolutionary operators (e.g. gradient mutation).

4.3.1 Smoothing and Balancing

In the case of optimization processes involving large number of decision variables, raw results obtained from the EA approach usually present a collection of locally scattered data. If information about solution smoothness (at least in subdomains) is available, it may be used for acceleration of the solution process. This may be done in various ways [5, 15]. First of all, an extra procedure based on the Moving Weighted Least Squares (MWLS) technique [19, 20], or any other equivalent approximation method, is applied in order to smooth the raw results obtained from the standard EA procedure.

In the MWLS technique, the weighted error functional

$$B = \sum_{i=1}^{n} (u_i - \bar{u}_i)^2 w_i^2 \tag{4.10}$$

is minimized at each point $\bar{x}$ with the respect to the set of local derivatives of function u. Here u_i is a nodal function value supplied by the EA, while $\bar{u}_i$ presents

its approximation by means of expansion into the p-th order truncated Taylor series, and w_i is a weighting factor. In the case of 1D, a local p-th order approximation is obtained as follows:

$$\bar{u}_i = \bar{u}(x_i) = \bar{u} + h_i \bar{u}' + \frac{1}{2} h_i^2 \bar{u}'' + \cdots + \frac{1}{p!} h_i^p \bar{u}^{(p)} + R \tag{4.11}$$

where $h_i = x_i - \bar{x}$, and R is a residuum of the Taylor series. Weighting function may be introduced as in [8]:

$$w_i^2 = \left(h_i^2 + \frac{g^4}{h_i^2 + g^2} \right)^{-p-1} \tag{4.12}$$

where g is a smoothing parameter, allowing us to control the intensity of smoothing. Minimization conditions:

$$\frac{\partial B}{\partial \bar{u}} = 0, \quad \frac{\partial B}{\partial \bar{u}'} = 0, \quad \frac{\partial B}{\partial \bar{u}''} = 0, \quad \ldots, \quad \frac{\partial B}{\partial \bar{u}^{(p)}} = 0 \tag{4.13}$$

provide a set of linear equations to be solved for the unknown function $\bar{u}$ and its derivatives up to the order p at each point $\bar{x}$.

The above formulation may be easily transformed into a 2D one. It may be found in [19, 20].

Appropriate choice of a value of the smoothing parameter $g \geq 0$ is of significant importance. For $g = 0$ the weighting function is singular and provides interpolation. Otherwise, we deal with the best approximation problem. The higher the value of the parameter is, the smoother is the approximation obtained.

In problems of mechanics each smoothing may result in the global equilibrium loss of a considered body. The equilibrium may be restored by the standard EA approach in a series of iterations. However, it may be also faster restored by means of an artificial balancing of body forces performed directly after the smoothing [16].

Information about smoothness may be also used in a selection of chromosomes process [5, 15]. A new criterion based on a mean solution curvature may be introduced into any selection operator. Mean local solution curvature κ may be calculated, e.g., using the definition based on the directional derivative, the same as in the case of the PBA formulation (4.8 and 4.9). The mean curvature of the solution in the whole domain may be calculated using the formula (4.7).

4.3.2 A'posteriori Error Analysis and Related Techniques

Due to stochastic nature of evolutionary computations, solutions obtained from independent populations may differ from each other. The weighted average of the best solutions taken from such populations is expected to be more precise than majority of these solutions. Such averaged and additionally smoothed afterwards solution may be used as a reference one for a'posteriori error estimation [1, 18].

Later on, the knowledge about the magnitude and the distribution of solution errors is used in order to intensify calculations in zones of large errors. We have proposed improved mutation and crossover operators taking into account information about local solution errors [18]. Information about estimated global error may be used by the modified selection operator.

A'posteriori error analysis may be well supported by parallel and distributed calculations in addition to other standard advantages provided by clusters. Moreover, representation of the best chromosomes, collected at the same time from all populations involved, may be also very useful, and significantly improve the solution process. All independent populations, as well as a population of representatives, are calculated simultaneously in a parallel way. Calculations carried out in each population may be partitioned among processing units as well.

More detailed information and wider numerical analysis of mentioned techniques using chosen benchmark problems may be found in [18].

4.3.3 Adaptive Step by Step Mesh Refinement

Solution time needed for optimization of functional is in many problems strictly dependent on the number of decision variables used, i.e. on the mesh density in the domain. The denser is a mesh in the domain the more time-consuming the solution process is. Therefore, the analysis can start from a coarse mesh and a fast, though not precise enough solution may be obtained at first. Starting from such solution, the mesh may be refined by inserting new nodes. Initial function values at these new nodes are found by means of an approximation built upon the nodal values of the coarse mesh. A general approach for most optimization problems may be obtained by using the MWLS approximation [19, 20] approach. However, any other approximation or interpolation method might be applied as well. Such approach may be repeated several times, until a sufficiently dense mesh is obtained.

Furthermore, the step-by-step mesh refinement may be also combined with the a'posteriori error analysis. Such strategy, using all techniques mentioned above, may be found in [5, 16].

4.4 Selected Benchmark Problems

The EA efficiency was examined using several benchmark problems, including residual stress analysis in chosen elastic perfectly-plastic bodies under various cyclic loadings. In particular, we have analyzed the residual stress in a cyclically bent bar, and in a thick-walled cylinder subject to cyclic loadings, like internal pressure, torsion and tension, including combined loadings [5, 15, 16, 18]. These problems may be analyzed as either 1D (taking into account existing symmetries)

or as 2D ones as well. Another advantage of the considered benchmark problems is possibility of testing almost any number of decision variables involved.

We have also investigated several benchmark problems using simulated pseudo-experimental data and the PBA approach, including smoothing of beam deflections, and reconstruction of residual stresses in a thick-walled elastic-perfectly plastic cylinder subject to cyclic internal pressure [17]. For smoothing of beam deflections we also used real experimental data obtained by vision measurement system.

Three chosen benchmark problems are described in a more detailed way below.

4.4.1 Residual Stress Analysis in Bending Bar

Considered is the residual stress analysis in an elastic-perfectly plastic bar of the rectangular cross-section subject to cyclic bending by the moment exceeding its elastic capacity. In the simplest 1D case the solution of the following optimization problem was searched:

Find self-equilibrated normal stress $\sigma = \sigma(z)$ minimizing complementary energy of the bar

$$\min_{\sigma} \int_0^H \sigma^2 dz \tag{4.14}$$

and satisfying the global self-equilibrium bending moment equation

$$M = \int_0^H \sigma \, z dz = 0 \tag{4.15}$$

as well as the inequality conditions resulting from the yield criterion

$$\left| \sigma + \sigma^e \right| \leq \sigma_Y \tag{4.16}$$

where σ_Y is the yield stress (plastic limit), and σ^e is the purely elastic solution of the problem considered. After discretization, where the sought normal stress $\sigma = \sigma(z)$ is replaced by the piecewise linear function spanned over the nodal values σ_i, the following formulation is obtained:

Find stresses $\sigma_1, \sigma_2, \ldots, \sigma_n$ satisfying

$$\min_{\sigma_1, \sigma_2, \ldots, \sigma_{n-1}} \left(\sum_{i=1}^{n-1} \sigma_i^2 + \frac{1}{2}\sigma_n^2 \right), \quad \sigma_n = -\frac{2}{z_n} \sum_{i=1}^{n-1} \sigma_i z_i \tag{4.17}$$

and inequality constraints

$$-\sigma_Y \leq \sigma_i + \sigma_i^e \leq \sigma_Y, \quad i = 1, 2, 3, \ldots, n \tag{4.18}$$

Numerical integration is used providing the exact results for piece-wise linear functions.

The target 3D non-linear constrained optimization problem resulting from residual stress analysis in railroad rails, and vehicle wheels is of similar, though much more complex nature. The exact formulation of this problem is given in [12, 13].

4.4.2 *Residual Stress Analysis in Pressurized Thick Walled Cylinder*

Considered is an elastic-perfectly plastic thick-walled cylinder under cyclic internal pressure. The following optimization problem given in the polar coordinates for residual stress is analyzed:

find the minimum of the total complementary energy

$$\min_{\sigma_r^r, \sigma_t^r, \sigma_z^r} \frac{1}{2E} 2\pi L \int_a^b \left((\sigma_r^r - \sigma_t^r)^2 + (\sigma_t^r - \sigma_z^r)^2 + (\sigma_z^r - \sigma_r^r)^2 \right) r \, dr \tag{4.19}$$

subject to the equilibrium equation

$$\frac{\partial \sigma_r^r}{\partial r} + \frac{\sigma_r^r - \sigma_t^r}{r} = 0 \tag{4.20}$$

the yield condition

$$\varphi(\sigma_r^r, \sigma_t^r, \sigma_z^r, \sigma^e) \leq \sigma_Y \tag{4.21}$$

the incompressibility equation

$$\sigma_z^r = \nu \, (\sigma_r^r + \sigma_t^r) \tag{4.22}$$

and boundary conditions

$$\sigma_{r|a}^r = 0, \quad \sigma_{r|b}^r = 0 \tag{4.23}$$

where σ_r^r, σ_t^r, σ_z^r are respectively the radial, circumferential and longitudinal residual stresses, $\sigma^e = \{\sigma_r^e, \sigma_t^e, \sigma_z^e\}$ is the purely elastic solution of the same problem, σ_Y is the yield stress, a, b are respectively the internal and external cylinder radii, L is its length, and E is the Young modulus.

4.4.3 Reconstruction of Residual Stresses Using the PBA Approach

Given are strains $\varepsilon_i^{\exp}$, $i = 1, 2, 3, \ldots, n$, experimentally measured in the 2D cross-section of the thick-walled cylinder under cyclic internal pressure. Find the residual stresses in its 2D cross-section. The following formulation in the polar coordinate system is used:

find the stationary point of the functional

$$\Phi(\sigma) = \lambda\, \overline{\Phi}^E(\sigma) + (1-\lambda)\, \overline{\Phi}^T(\sigma)\ , \quad \lambda \in [0,1] \tag{4.24}$$

where

$$\Phi^E(\sigma) = \left(\frac{1}{n} \sum_{i=1}^{n} (\varepsilon_i^{\exp} - \varepsilon_i^{app}(\sigma))^2 \right)^{\frac{1}{2}} \tag{4.25}$$

$$\Phi^T(\sigma) = \frac{1}{\Omega} \int_{\Omega} \kappa^2(\sigma) d\Omega \tag{4.26}$$

satisfying equality constraints (4.20, 4.22 and 4.23), and inequality constraints for admissible local and global errors

$$\left| \varepsilon_i^{\exp} - \varepsilon_i^{app}(\sigma) \right| \le e_i\ , \quad i = 1, 2, 3, \ldots, n \tag{4.27}$$

$$\sqrt{\Phi^E(\sigma)} \le e_E \tag{4.28}$$

The mean solution curvature is calculated using the formula (4.9).

4.5 Numerical Results

The main objective of numerous executed tests was to evaluate correctness, efficiency, and ability of the proposed acceleration techniques to deal with large, and very large optimization problems. At first a choice of the most efficient combination of the standard EA operators was sought. Searching the best combination of operators, as well as adjusting their parameters, the acceleration up to several times may be reached. From numerous variants of operators we preliminarily chose several popular ones: rank and tournament selection, arithmetic and heuristic crossover, uniform, non-uniform and border mutation. Using the best

combination found, namely rank selection, heuristic crossover, and non-uniform mutation, particular already mentioned acceleration techniques were analyzed. Some other results of our efficiency analysis were also described in [16, 18], and very briefly in [5, 15].

Acceleration of calculations was measured using four speed-up factors proposed, and defined in [18]. These factors take into account convergence of mean solution error as a function of time, or number of iterations. Convergence of fitness function is also measured.

All presented results were averaged over 10 independent solution processes.

4.5.1 Smoothing and Balancing

In the considered tests, the MWLS technique was used for additional smoothing of raw EA results. When using this technique it is necessary to establish values for two extra parameters: order of local approximation p, and smoothing parameter g. Various values of these parameters may have significant influence on the convergence of the solution process. Number of standard iterations between subsequent smoothing and balancing operations has to be considered as well. In Figs. 4.1 and 4.2 one may see results obtained in the bending bar analysis (benchmark 4.4.1).

In the case of this benchmark test, the best results were obtained for the linear local approximation $p = 1$ (see Fig. 4.1). There is also no significant difference between results obtained for $p = 2$ and $p = 3$. However, the best results obtained in case of $p = 1$ may result from the specific features of the sought solution of the problem, which is piece-wise linear. For more complex solutions higher local approximation orders will be needed. In general, the order of local approximation should depend on the order of differential operators used.

In Fig. 4.1 you may also see additional time needed for each smoothing and balancing operation. These operations were repeated after each 300 iterations. All optimization processes shown in Fig. 4.1 were carried out for 3000 iterations, so you may find the whole additional time needed for all extra smoothing operations. This extra time is not significant when compared to obtained gains. The results were obtained for smoothing parameter $g = 5$. Other executed tests, not presented here, showed that choice of the value of g parameter was not significant, excluding small ones. For $g \in [3, 20]$ obtained results were very similar.

Application of our smoothing technique based on the MWLS, and balancing procedure based on the linear correction terms allowed to achieve up to about 4 times efficiency increase (see Fig. 4.2). Smoothing technique was also tested using benchmark 4.4.2 and gave encouraging results as well.

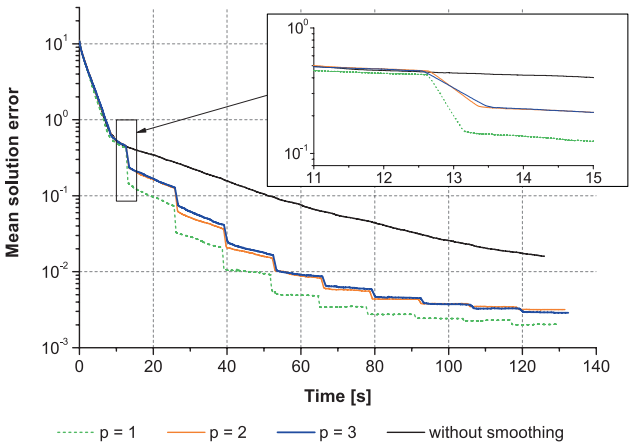

Fig. 4.1 Results of smoothing and balancing for various orders of local approximation in the MWLS technique

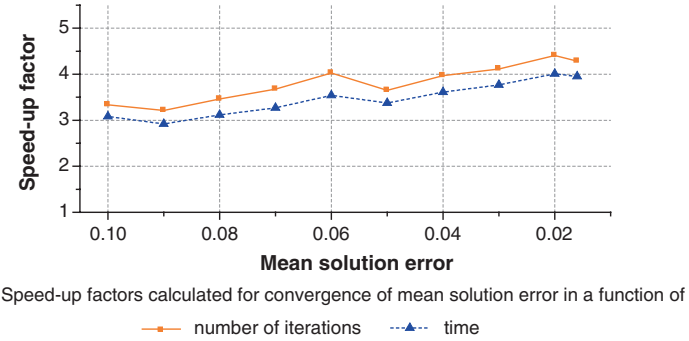

Fig. 4.2 Speed-up factors for smoothing and balancing

4.5.2 Constraint Handling—Penalty Functions

A type of penalty function, as well as its parameters may have a significant impact on the convergence rate of the optimization process. In our research, the following function was used for constraint handling:

$$F = f + \sum_{i=1}^{n} \alpha \, d_i^{\beta} \tag{4.29}$$

where F is the new (expanded) objective function, f is the standard fitness function, d_i is the distance of i-th decision variable to constraint boundaries, n is the number of decision variables, α and β are parameters.

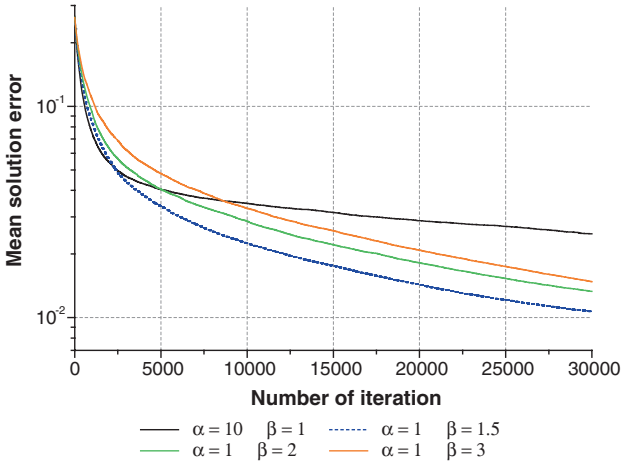

Fig. 4.3 Comparison of convergence of mean solution error for various penalty functions

In Fig. 4.3 one may see results obtained for various values of parameters α and β in cyclically pressurized cylinder analysis (benchmark 4.4.2). In this example, the best results were obtained for $\alpha = 1$ and $\beta = 1.5$. When comparing the best and the worst cases shown in Fig. 4.3, the speed-up about 3 times was reached.

4.5.3 Step by Step Mesh Refinement

Results shown in Fig. 4.4 were obtained in reconstruction of residual stresses in the thick walled cylinder under cyclic internal pressure using pseudo-measurements of strains and the PBA approach (benchmark 4.4.3). Numerical data used in this experiment were randomly generated using the true (analytical) solution as a base curve. A strain gauge technique was simulated. Assumed were delta type rosettes, giving three components of strains. All calculations were carried out in the 2D domain. The random data generator used Gaussian distribution. More detailed description of methodology of such tests, as well as solutions obtained, may be found in [17]. In this paper only a brief analysis of calculation efficiency is presented.

Comparison of the convergence of mean solution error for standard and improved algorithms is shown in Fig. 4.4. In this case, the improved EA used a series of denser and denser meshes, combined with smoothing technique. The process started with 16 nodes, and was continued until the number of 1248 nodes was reached. The mesh was refined 4 times. Each nodal value corresponded to one decision variable (gene in a chromosome). In comparison to the standard EA, the acceleration factor of the optimization process up to about 140 times was reached.

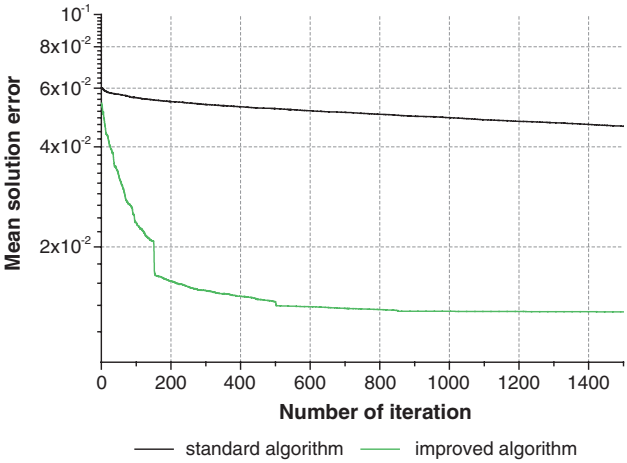

Fig. 4.4 Comparison of convergence of mean solution error for standard and improved algorithm

4.6 Conclusions

The general objective of this research is development of highly efficient, dedicated EA for solving large non-linear constrained optimization problems. Preliminary results of many executed tests clearly show a possibility of significant efficiency increase when using all proposed acceleration techniques. The speed-up about 140 times was reached. It is also worth noticing, that the improved EA allowed obtaining solutions in cases when the standard EA failed to solve problems due to too large number of decision variables. Results obtained indicate also a clear possibility of practical application of the improved EA to the PBA of experimental and/or numerical data for large optimization problems. Application of the accelerated EA to the PBA is still at the initial stage of research development, however preliminary results are very encouraging.

Future research includes continuation of various efforts oriented towards efficiency increase of the EA-based optimization approach, analysis of further, demanding benchmark problems, and application of such developed method to residual stress analysis in railroad rails, and vehicle wheels [8, 12, 13]. The PBA approach for smoothing of experimental data is also expected.

References

1. Ainsworth M, Oden JT (1997) A-posteriori error estimation in finite element analysis. Comput Methods Appl Mech Eng 142:1–88
2. Burczynski T, Orantek P (1999) Evolutionary and hybrid algorithms. In: Gajewski RR (ed) Neural Networks, Genetic Algorithms, Fuzzy Sets [in Polish], BEL, Rzeszow, pp 99–117

3. Coello Coello CA (2002) Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. Comput Methods Appl Mech Eng 191(11–12):1245–1287
4. Engelbrecht AP (2007) Computational intelligence: an introduction. Wiley
5. Glowacki M, Orkisz J (2013) Advances in development of dedicated evolutionary algorithms for large non-linear constrained optimization problems. IPPT Rep Fundam Technol Res 47(4):25–29
6. Grosan C, Abraham A, Ishibuchi H (eds) (2007) Hybrid evolutionary algorithms. Studies in Computational Intelligence, vol 75. Springer, Berlin Heidelberg
7. Hill R (2004) The mathematical theory of plasticity. Oxford University Press, London
8. Karmowski W, Orkisz J (1993) Physically based method of enhancement of experimental data - concepts, formulation and application to identification of residual stresses. In: Tanaka M, Bui HD (eds) Proceedings of IUTAM symposium on inverse problems in engineering mechanics, Tokyo, 1992, Inverse problems in engineering mechanics, Springer-Verlag, pp 61–70
9. Kus W, Burczynski T (2008) Parallel bioinspired algorithms in optimization of structures. Lect Notes Comput Sci 4967(2008):1285–1292
10. Michalewicz Z (1996) Genetic algorithms + data structures = evolution programs. Springer, Heidelberg
11. Nedjah N, Alba E, Mourelle L (eds) (2006) Parallel evolutionary computations. Studies in Computational Intelligence, vol 22. Springer, Berlin Heidelberg
12. Orkisz J (1992) Prediction of actual residual stresses by constrained minimization of energy. In: Orringer O, Orkisz J, Swiderski Z (eds) Residual stress in rails, vol. 2. Kluwer Academic Publisher, New York, pp 101–124
13. Orkisz J et al (2004) Development of advanced methods for theoretical prediction of shake-down stress states and physically based enhancement of experimental data. US DOT Report, DTFR53-03-G-00012, Cracow
14. Orkisz J, Cecot W (1997) Prediction of actual residual stresses resulting from cyclic loading in kinematic hardening material. In: Onate E, Owen DRJ (eds) Proceedings of the 5th international conference on computational plasticity, CIMNE, Barcelona, pp 1039–1042
15. Orkisz J, Glowacki M (2014) On dedicated evolutionary algorithms for large non-linear constrained optimization problems. In: Igel Ch (ed) GECCO'14 Companion Publication of the 2014 Genetic and Evolutionary Computation Conference, Vancouver. ACM, New York, pp 121–122
16. Orkisz J, Glowacki M (2014) On dedicated evolutionary algorithms for large non-linear constrained optimization problems in application to residual stresses analysis. In: Onate E, Oliver X, Huerta A (eds) Proceedings of the jointly organized 11th world congress on computational mechanics 5th european conference on computational mechanics and 6th european conference on computational fluid dynamics, CIMNE, Barcelona, pp 2073–2084
17. Orkisz J, Glowacki M (2014) On improved evolutionary algorithms application to the physically based approximation of experimental data. Comput Assist Methods Eng Sci 21(1):27–38
18. Orkisz J, Glowacki M (2014) On acceleration of evolutionary algorithms taking advantage of a posteriori error analysis. Comput Inform 33(1):154–174
19. Orkisz J, Liszka T (1980) The finite difference method at arbitrary irregular grids and its applications in applied mechanics. Comput Struct 11:83–95
20. Salkauskas K, Lancaster P (1990) Curve and surface fitting. Academic Press, San Diego

Chapter 5
Lightweight Optimization for Additive Manufacturing Parts Based on Genetic Algorithms, Metamodels and Finite Element Analysis

Rubén Paz, Mario Monzón, Begoña González, Gabriel Winter and Fernando Ortega

Abstract Additive manufacturing (AM) has become in a competitive method for short series production and high flexibility applications even for functional parts. Few constraints in the manufacturing process involve a great design freedom, allowing minimization of weight by using internal cellular and lattice structures, while minimal mechanical requirements are kept. Weight minimization implies a lower use of material and hence a reduction in manufacturing time, leading to a cost reduction. However, design optimization requires a greater effort in the design process, which also results in more costs. In order to reduce the design process, an optimization method based on genetic algorithms (GAs) and computer aided design/finite element method (CAD/FEM) simulations is proposed to optimize the cellular structure design and minimize the weight for AM parts. New optimization strategies based on GAs combined with surrogate models are evaluated and compared to reduce as much as possible the number of FEM simulations.

R. Paz (✉) · M. Monzón · F. Ortega
Departamento de Ingeniería Mecánica, Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas G.C, Spain
e-mail: rpaz@lfi.gi.ulpgc.es

M. Monzón
e-mail: mmonzon@dim.ulpgc.es

F. Ortega
e-mail: fortega@dim.ulpgc.es

B. González · G. Winter
Institute of Intelligent Systems and Numerical Applications in Engineering—SIANI, Evolutionary Computation and Applications (CEANI), Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas G.C, Spain
e-mail: bgonzalez@iusiani.ulpgc.es

G. Winter
e-mail: gabw@step.es

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*, Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2_5

Keywords Lightweight optimization · Additive manufacturing · Genetic algorithms · Metamodels

5.1 Introduction and Objectives

The continuous evolution of AM technologies in terms of materials, reliability and reproducibility of the processes, as well as cost reduction compared with conventional manufacturing techniques have led to an increasing use of these technologies in the industry. The low manufacturing constraints associated to these processes involve an enormous design freedom ideal to manufacture complex parts without any cost increase.

Moreover, the possibilities around the CAD/FEM software are well known. FEM numerical simulations allow the determination of the mechanical behavior of any part, being a fundamental tool in the design process. Combining the potential of AM technologies and CAD/FEM tools is possible to reduce the part weight by introducing cellular structures repeated inside the part (without changing the previous external design) [1]. Cellular structures can be generated and parameterized in a CAD model without an excessive effort, especially if the cells are defined with a repeated pattern. FEM simulations of any new design with cellular structures allow knowing its mechanical properties, information that is essential for the design process. Weight minimization can be achieved by optimizing the cell pattern dimensions with an optimization method. In fact, results of finite element analysis (FEA) can be employed to the evaluation of the fitness function in an optimal searching with GAs [3]. Finally, the best design can be manufactured by AM technologies despite the complex internal cellular structure.

Weight minimization not only means a greater efficiency in multiple applications, but also significant reduction of manufacturing costs, either material savings or manufacturing time. However, the extra time required for the design optimization also entails a cost increase, which means that the optimization time must be minimized in order to obtain a more competitive product.

For these reasons, weight minimization will be made through repeated cell geometries (from a pattern) inside the part, which implies less variability of individuals and a smaller number of design variables (less than 7). Although this simplification reduces the searching space and probably the quality of the optimal individual, greatly facilitates the CAD modeling and optimization tasks, significantly reducing the design costs.

Moreover, the evaluation of the fitness function for each individual generated during the GA evolution requires FEM simulations. This would involve an excessive computational time [3]. Therefore, the use of surrogate models to estimate the FEM results without doing the simulations is proposed, reducing the number of the computationally expensive analyses as much as possible [5]. The aim of this

approach is to establish a simple methodology that can be used by any AM user through commercial CAD/FEM software. These technologies are becoming more and more affordable due to cost decreasing associated to the patent expiration. Thus, SMEs or even particular users will be able to buy AM machines and manufacture their own parts, taking advantage of the optimization strategies developed in this proposal through commercial CAD/FEM software easily accessible.

5.2 Main Program Structure

To create a surrogate model it is necessary a previous information of the system behavior. This information is achieved through an initial design of experiments (DOE), where a set of designs are simulated by FEM.

Once the surrogate model is defined, a GA is applied to search the optimal design by evaluating the fitness function of each individual through the surrogate model estimations. Although different versions of the program were tested, the general approach is to refine the metamodel by simulating new designs strategically located in interesting regions, including the results into the database to upload the metamodel. Once the surrogate model guarantees a certain level of accuracy in the estimations, the optimal design is searched again using GAs and metamodel to evaluate the fitness function, reducing as much as possible the number of FEM simulations.

The design variables are related to the dimensions of the pattern cell geometry repeated inside the part, existing in this case a monotonic relation with the system responses. An increase in a variable associated with the hollow cell size always involves a mass reduction and a worse mechanical behavior, adversely affecting the problem optimization constraints (displacements or stresses). This particular relation implies that the optimal design will be always in the border between the feasible and unfeasible regions, so that the optimum will have at least one constraint very close to its limit value.

Given this fact, the addition of new points in the DOE phase or metamodel refinement stage is carried out trying to increase the sampling density in areas close to this border (feasible/unfeasible), which means simulating designs in interesting zones. This strategy of DOE and surrogate model refinement also implies a better fitting in the feasible/unfeasible border than in other regions of the searching space. Refinement strategies usually add new sample points where the lowest accuracy of the metamodel are estimated. However, in this proposal the accuracy of the metamodel is only relevant in the feasible/unfeasible zones. In other regions the surrogate model has a lower precision, but enough to estimate the results without affecting the convergence of the GAs. This refinement method requires a lower sampling, allocating the new points in areas near to the optimum and consequently maximizing the sampling effort.

5.3 Comparison Between Different Metamodels

First a comparison between different surrogated models was carried out to determine the most appropriated metamodels in terms of estimation error for this application. The evaluated metamodels were:

- Inverse distance interpolation: Four different configurations of this interpolation method were evaluated. The first one was implemented using an exponent of 2 in the inverse distance calculation and involving all the available data (IDI2). This same method (with exponent 2 in the inverse distance calculation) was applied again but involving only the 6 nearest data to the point to be estimated (IDI2 6p). After that, the inverse distance exponent was increased to 3, considering all available data in the estimations (IDI3). Finally, a fourth configuration was carried out by using again exponent 3 in the inverse distance calculation but taking into account only the 6 nearest data to the point to be estimated (IDI3 6p). Some authors have observed better results for low sampling problems with this method than with other more complex ones [7].
- Spline interpolation (SI). The main advantage of this method is that it allows the interpolation of values that are below the minimum or above the maximum of the available data, while other methods cannot [2].
- Least square fitting: Two different configurations of the least square fitting were evaluated. The first configuration was developed by fitting the coefficients of a two-order polynomial to the available data (LSF2). The second configuration was carried out in a similar way but using a three-order polynomial to be fitted to the available data (LSF3). The use of polynomial equations in fitting problems with unknown response is a common practice, although the most usual practice is to employ one- or two-order equations [6].
- Linear interpolation based on Delaunay triangulation (LIDT). This method partitions the space into discrete simplex (n-dimensional) following the Delaunay triangulation (dual to the Voronoi diagram or Thiessen polygons). Given a set of points (P) in the n-dimensional space, the Delaunay triangulation is a triangulation such that no point in P is inside the circumhypersphere of any simplex. This method maximizes the minimum angle of all the simplexes. Once the space is discretized according to Delaunay triangulation, the method identifies the simplex of the point to be evaluated and finally a linear interpolation of the vertex values is applied (by a weighted sum of the vertex values, being the weights the barycentric coordinates). The main disadvantage of this interpolation method (also known as Triangulated Irregular Network, TIN) is that the domain is limited to the convex envelope of the data and the resulting surface is not smooth [2].
- Nearest neighbor interpolation (NNI). The NNI selects the value of the nearest point. This is equivalent to the Voronoi diagram, which means that the interpolation values will be the same in each one of the tessellation cells. This method is less accurate but quite simple.

A problem of four design variables was employed in order to compare the accuracy or estimation error among the different surrogated models mentioned above.

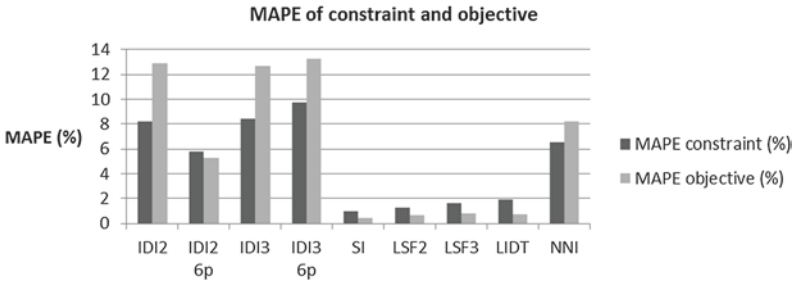

Fig. 5.1 MAPE for evaluated metamodels

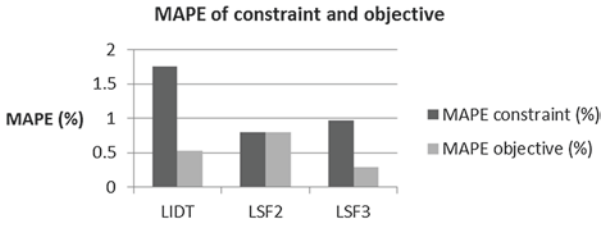

Fig. 5.2 MAPE for least square fitting and linear interpolation metamodels

Eighty one individuals related to a 3-level full factorial DOE were evaluated by FEM simulations. The different types of metamodels were constructed with the data obtained in the previous simulations. After that, 10 random points of the domain were evaluated by FEM and were also estimated by the surrogate models in order to evaluate the estimation error (difference between the estimated value and the FEM result in absolute value). Figure 5.1 shows the mean absolute percentage error (MAPE) of these 10 points for the different metamodels and for the 2 responses of the problem.

Best results were obtained for SI, LSF2/LSF3 and LIDT. Although SI provides more accurate results for both responses, the data distribution required to construct the spline must be in a grid, which complicates the refinement tasks and implies an enormous sampling intensity even using T-splines with the "quadtree" method [4].

Another similar study was made comparing only the LSF2, LSF3 and LIDT. In this case 33 different designs were evaluated by FEM. The allocation of these points was defined according to the first optimization strategy developed in Sects. 5.5 and 5.5.1. The first 17 points correspond to a 2-level full factorial DOE and central point, and the 16 remaining points correspond to an iteration of the border (feasible/unfeasible) approximation. The metamodels were elaborated from the results of these 33 sampling points. Then 16 new points associated with a new iteration of the border approximation were evaluated either by FEM or by the predictions of the surrogate models. Figure 5.2 shows the MAPE obtained for LIDT, LSF2 and LSF3, for both constraint and objective responses.

Although the results of LSF2 and LSF3 are even better than those of LIDT, this last metamodel was chosen for this proposal because it is an interpolation method, which means exact predictions on the data points and ensures greater accuracy than the least squares fitting when the sampling is intensified in an area. The least square fitting does not estimate exact results in the data points and its potential is limited by the shape of the equation to be fitted, which could imply a high distortion and error in some areas when the sampling density is increased in a specific zone. Furthermore, the refinement strategy discussed above will only work correctly with a surrogate model which can improve its accuracy as new points are added.

5.4 Genetic Algorithm

A GA with different configurations was implemented to solve a known problem with four variables (real numbers), one constraint and one objective to be minimized. This reference problem was employed to validate the different programs developed in this paper without doing the FEM simulations. The fitness value of the theoretical optimal design is F = 1600.809. The number of individuals of the population was fixed in 100, with a tournament selection of 2 individuals, arithmetic crossover and application of elitism. The parameters of the different GA configurations were as follows:

- Penalty amplification factor (AF): individuals who did not satisfy any constraint of the optimization problem were penalized with a certain penalty which was amplified by a factor defined as a fixed value or as a value that grew exponentially with the number of generations. This latter option seeks to assign a greater freedom in the first iterations of the GA and become more restrictive as the GA evolves.
- Type of penalty: individuals who did not satisfy any constraint of the optimization problem were penalized with an error value which was obtained from the squared error (SE) or absolute error (AE), always amplified by the penalty amplification factor mentioned above. The total penalty for each individual was determined as the sum of the penalties associated with each of the non-satisfied restrictions.
- Total number of generations evaluated: 50 or 100.
- Cross probability: 50 or 80 %.
- Mutation probability: mutation probability was defined as a fixed value or a variable value that increases linearly with the number of generations. This latter option seeks to provide a greater localized variability as the GA evolves, in order to avoid convergence problems (convergence to local optima instead of global optima). The more evolved the population is, the higher the mutation probability becomes, which reduces the problems of stagnation in a local optimum.

Table 5.1 Different configurations tested

Geometry parameters	Config. 1	Config. 2	Config. 3	Config. 4	Config. 5	Config. 6
Amplification factor	10^{99}	10^{gen}	10^{gen}	10^{gen}	10^{99}	10^{99}
Penalty	SE·AF	SE·AF	SE·AF	AE·AF	SE·AF	SE·AF
Number of generations	50	50	100	50	50	50
Cross probability (%)	80	80	80	80	50	50
Mutation probability (%)	10	10	10	0–60	60	80
Mutation amplitude (%)	10	10	10	0–50	50	10
Fitness average (F)	1615.3	1615.0	1606.3	1601.7	1601.9	1601.8

- Mutation amplitude: individuals were randomly mutated with a maximum amplitude value defined by a fixed value or variable with the number of generations. This latter option seeks to provide a more intensive mutation as the GA evolves in order to improve the convergence to the optimal.

Table 5.1 shows a summary of the six configurations tested and an average value of the optimal fitness function for 10 different runs. The best results were obtained for configurations 4, 5 and 6, with fitness values very close to the theoretical optimal value. This means that the GA can converge to the optimal with different configurations, which demonstrates its robustness and flexibility. Configurations 5 and 6 were chosen to be implemented in the different developed programs, while configuration 4 was rejected due to be a more complex option.

5.5 Optimization Programs Developed

Different optimization strategies were developed and tested with the previous known problem (without FEM simulations). The last 2 versions were also tested with a case study with FEM analysis.

5.5.1 Version 1

The first version consist of a 2-level full factorial DOE and central point, followed by a phase of addition of points near to the feasible/unfeasible border and finally a GA.

LIDT can only be applied inside the convex hull of the data. For this reason, it is important that the initial DOE allows the creation of a convex hull that covers the entire domain, in order to apply LIDT throughout the searching space. A 2-level full factorial DOE was chosen as the best option to achieve the desired convex hull with the minimal number of points (which is equivalent to evaluating

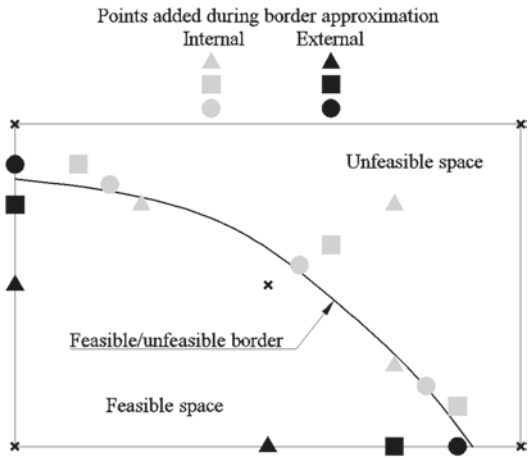

Fig. 5.3 Points added during the initial DOE and border approximation in a 2D problem

all vertices of the domain). Apart from these points, the central point of the domain was also simulated in this initial stage of DOE (black crosses in Fig. 5.3).

The phase of approximation to the feasible/unfeasible border was divided into 2 parts: (a) an "internal" approximation by adding new points in the middle between the central point and the corners when one of them is in the feasible space and the other does not (or vice versa); (b) an approximation "along the edges of the domain", adding new points in the middle between adjacent corners when one of them is in a feasible area and the other does not (square, triangle and circle points in Fig. 5.3). This new points will be closer to the feasibly/unfeasible border and consequently closer to the optimal design. This phase is repeated in a loop until the mean absolute deviation of the points added in the last iteration is less than the maximum deviation assigned by the user to each response. Triangles, squares and circles of Fig. 5.3 represent the points added in 3 iterations respectively, both internal (in grey) and external (in black) approximation.

Finally, a GA based on configuration 5 is applied. In this case, the fitness value is evaluated by LIDT through the available data of previous simulations. In addition, the best individual of each generation is simulated and then the information obtained is added to the available data in order to refine the metamodel, thus getting a more accurate metamodel as the GA evolves. Once the GA ends, the best simulated design is chosen.

This version was executed 10 times with the reference problem (5 tests with 100 generations and 5 with 500). The average value of the optimal fitness functions was 1659.267 and 1660.593 respectively, so no improvements were observed by increasing the number of generations. The total average value of fitness function was 1659.93 (with an average of 80 points evaluated), which differs significantly from the fitness value of the theoretical optimum ($F = 1600.809$). After several tests, it was observed that the GA does not converge to the theoretical optimum because of the lack of accuracy of the metamodel. Hence, the simulation of

the best individual of each generation does not significantly improve the fit of the metamodel. So, new points must be added before applying the GA if the results are intended to be improved.

5.5.2 Version 2

In order to improve the results, new middle points were added between the point with minimum mass found in the last iteration of the border approximation along the edges and the remaining points associated with adjacent corners of the feasible/unfeasible border. This approach was implemented in version 2. Figure 5.4 shows this new strategy in a 3D problem. The square black point outlined in grey represents the point with lower mass among the square black points added in the last iteration of the border approximation along the edges. This point is combined with the remaining adjacent square black points to obtain the two middle points represented as black circles in Fig. 5.4.

Additionally, the phase of border approximation (internally and along the edges) was carried out by linear interpolation to improve the convergence to the border, identifying the two closest points to the border (each in the opposite feasible/unfeasible space) and allocating the new point on the border line estimated by linear interpolation of the 2 selected data for each constraint of the problem. Finally, the proposed point that is closer to the feasible zone (corresponding to the most restrictive constraint) is simulated. For example, in the case of having 2 constraints involved in a border approximation along one edge (see Fig. 5.5), each involved constraint will lead to a proposed point (square and triangle points). These points are obtained by linear interpolation of the constraint values associated with the 2 closest data on this edge (one in the feasible zone of the specific constraint and another one in the unfeasible zone). Therefore, only one of the 2 proposed points must be chosen. The closest to the feasible vertex of the edge will

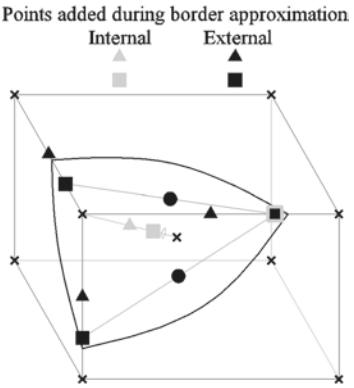

Fig. 5.4 New middle points (*black circles*) added in a 3D problem

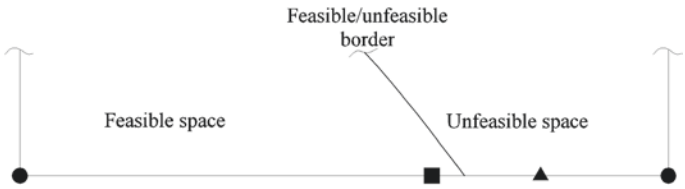

Fig. 5.5 Border approximation phase through different constraints and selection of the proposed point which is closer to the feasible zone

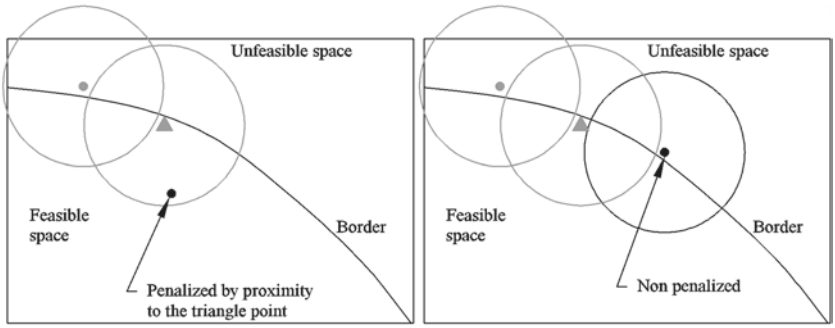

Fig. 5.6 Proximity penalty strategy during the border approximation phase by GAs

be selected (in this case the square point). This step is repeated while the MAPE of the critical constraint value of the different points added in the last iteration (compared to the limit value) is greater than 1 %.

Afterwards a border approximation phase by GAs (configuration 5 with 100 generations) was implemented, using LIDT to evaluate the fitness function. The best individual achieved by the GA is analyzed by FEM and added to the database to upload the metamodel. The GA is executed again, but penalizing the individual that is near to the points added previously in this phase. This strategy converges to a different optimum in each successive GA execution, which involves adding different new points along the feasible/unfeasible border, exploring interesting zones. For example, the circle black point (Fig. 5.6, left image) would be penalized by proximity to the triangle grey point (added in the previous execution of the GA). Hence, the GA evolves towards a point out of the proximity penalty radius of the points added in this stage of the program. Once the circle black point (Fig. 5.6, right image) is evaluated, the GA is executed again but also penalizing the proximity to this new point. This step is repeated until at least "n" points (n = number of design variables) have been added in this phase. After that, the MAPE of the last added point (response estimations compared to simulations) is evaluated. If the MAPE is bigger than 1 %, the metamodel is uploaded with this last point and this GA is executed again applying proximity penalty. And so on until the MAPE value is less than 1 %.

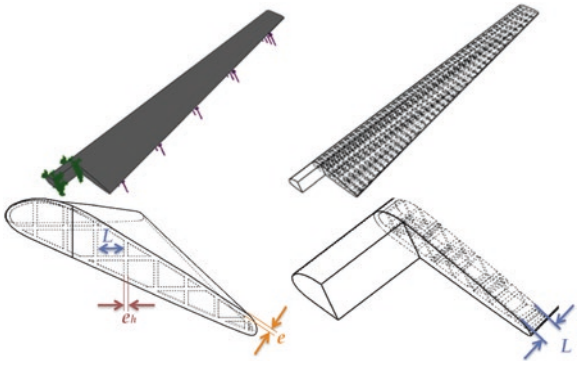

Fig. 5.7 Case study geometry and design variables

Subsequently a final GA (configuration 6 with 200 generations) is run, using LIDT to calculate the fitness function value. The best individual is simulated by FEM. If it is in the feasible zone, is the optimum, otherwise the results are added to the database and the metamodel is uploaded to execute again this final GA. And so on until reaching a feasible optimum.

In 10 different runs of this program version with the reference problem, the average value of the optimal fitness function was F = 1603.715, very close to the fitness value of the theoretical optimum (1600.809), with an average of 62 evaluated designs.

A case study with FEM simulations (see Fig. 5.7) in which it is pretended to minimize the weight of a blade for wind power micro-turbine lightened by cellular structures (3 design variables) keeping the maximum deflection under 15 mm (constraint) was also solved. The 3 design variables (see Fig. 5.7) were the length of the sides of the cubic hollows ("L", varying between 20 and 60 mm), the external thickness ("e", varying between 3 and 8 mm) and the thickness between the cubic hollows ("e_h" varying between 3 and 8 mm).

The optimization problem can be represented as follows:

$$\begin{aligned} &\textit{Minimize} && \textit{mass}\,(L, e, e_h) \\ &\textit{Subject to} && \max.\textit{deflection} \leq 15 \\ & && 20 \leq L \leq 60 \\ & && 3 \leq e \leq 8 \\ & && 3 \leq e_h \leq 8 \\ &\textit{Individual} && (L, e, e_h) \end{aligned} \tag{1}$$

Figure 5.8 shows the responses for each one of the 40 designs evaluated during the program evolution (maximum deflection and weight). The weight values were divided by the weight of the optimal design obtained (1632.55 g), while the deflection values were divided by the maximum permitted deflection (15 mm), representing then the relative values of both responses in the same graphic. It can be observed how the relative deflection tends to 1, which means that the program

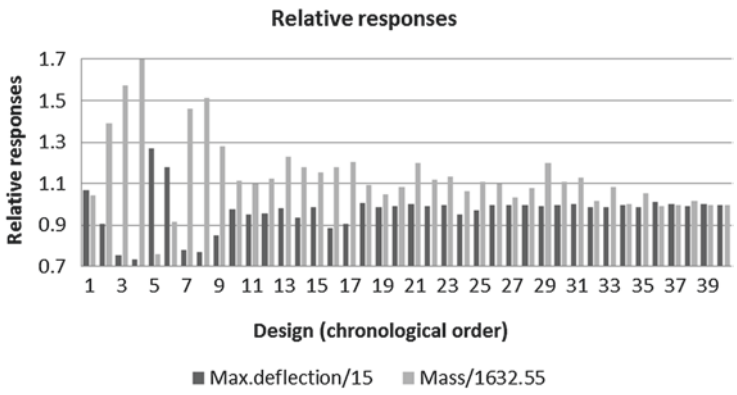

Fig. 5.8 Relative responses of the designs evaluated during the optimization process

evolves to designs with a maximum deflection close to 15 mm in order to minimize the weight as much as possible. Therefore, as it was expected, best design is near to the feasible/unfeasible border.

The optimal design obtained after 40 FEM simulations has a mass of 1632.55 g and 14.992 mm of maximum deflection, being its design variables "L = 39.875 mm", "e = 4.091 mm" and "e_h = 3 mm". This same problem was also solved by an optimization method based on Box-Behnken DOE and optimal estimation by response surface method (BBRS), an optimization strategy available in the commercial software of design and FEM simulations, SolidWorks. Response Surface Methods (RSMs) are considered a very effective approach for optimization problems with a small number of design variables, which is ideal for this application. The BBRS method achieves an optimal of 1690.07 g ("L = 47.531 mm", "e = 4.496 mm" and "e_h = 3.193 mm") with only 14 simulations. The proposed methodology reaches an optimum 3.52 % better but requires quite more simulations. For this reason, a new version was developed in order to reduce the number of FEM simulations required during the evolution of the optimization algorithm.

5.5.3 Version 3

The last two phases of the program (based on GAs) were tested by excluding different data set in order to evaluate the convergence of the program to the optimum without these points. These tests were carried out solving the reference problem without FEM analysis in order to accelerate the process. The conclusions obtained in this analysis are listed below:

- The points added during the internal feasible/unfeasible border approximation were deleted and the algorithm evolved to practically the same optimal solution, which means that the points added during the internal border approximation have no effect on the quality of the optimum. For this reason, this step was excluded.
- The phase of border approximation along the edges was carried out only edge by edge, achieving a deviation from the real feasible/unfeasible border less than 1 % at each affected edge. This new strategy showed a significant improvement in the solution because it helps to correctly select the best corner of the feasible/unfeasible border for the next phase of the code. Furthermore, this idea has the advantage of varying the number of iterations depending on the design variable that is being changed during the border approximation. This means that the border approximation through one of the design variables may need 5 iterations to achieve a deviation less than 1 % at the most restrictive constraint, while the border approximation along another edge may need just 2 iterations. Hence, the control statement in the new program (version 3) is just the difference (absolute value) between the critical value of the most restrictive constraint and the value obtained for this same constraint in the simulation of the last point added during the border approximation, thus controlling the deviation in the associated edge. However, in the previous version, the border approximation was carried out by adding a new point in each affected edges, repeating this process if the MAPE of the points simulated in this iteration was greater than 1 %. So, the number of points added during the border approximation was the same in all the edges affected and thus some points were probably closer to the border than others, increasing the risk of error in the next phase of the program, where it must be selected the best corner (best point added in the last iteration) of the border between feasible and unfeasible regions.
- Despite the proposal has been developed for a small number of design variables (less than 7), the addition of new middle points between the best border corner and the remaining adjacent corners involves incorporating a lot of points, growing exponentially as the number of design variables increases. However, after some tests it was observed that the method also converged to the theoretical optimum just combining the best border corner with the "n-1" best remaining adjacent corners. For this reason, version 3 was implemented with this new strategy, which means to combine only the best border corner with the "n-1" best remaining adjacent corners instead of considering all possible combinations, hence reducing the sampling intensity.

The new version 3 was programmed and then it was executed 10 times for solving the reference problem (without FEM analysis). The average value of the optimal fitness function was $F = 1606.050$, with an average of 44 sampling points. The new version converges to a solution 0.15 % worse than the previous version, but requires only 44 instead of 62 simulations, reducing in approximately 29 % the CPU time if a linear relation between CPU time and number of designs evaluated by FEM is assumed.

Table 5.2 Some of the designs evaluated during the evolution of version 3 (with FEA)

Point	L (mm)	e (mm)	e_h (mm)	Maximum deflection (mm)	Mass (g)
1	20	3	3	16.061	1705.09
8	60	8	8	11.543	2470.68
9	40	5.5	5.5	12.741	2087.14
11	20	3	4.7	14.905	1948.70
13	20	3.562	3	14.975	1814.40
15	35.157	3	8	15.015	1847.10
18	60	4.577	3	14.851	1643.70
22	60	4.115	8	15.045	1741.10
23	60	4.346	5.5	14.987	1694.90
24	40	4.07	3	15.04	1626.80
25	60	4.521	3	15.07	1630.30
27	31.33	3.858	3	14.943	1668.40
28	56.705	4.479	3	15.043	1631.60
29	60	4.54	3	14.975	1634.85

Applying this new version in the previous case study (with FEM analysis), an optimal design of 1634.85 g and only 29 sampling points was found ("L = 60.000 mm", "e = 4.540 mm" and "e_h = 3.000 mm"). This optimal design increased the mass by 0.14 % compared to the optimum obtained with version 2, but the number of evaluated designs was reduced from 40 to 29 (approximately 27.5 % of CPU time reduction). Compared with the result of BBRS method, this program improves the optimal 3.38 % but requires more sampling points (29 vs. 14). However, it ensures the convergence to the theoretical optimum due to the refinement loops, while the BBRS method does not guarantee the convergence to a feasible design and its refinement is quite limited by the equation shape to be fitted. In addition, it should be noted that the sampling point number 18, which is added during the border approximation along the edges, improves the optimum obtained by BBRS method with only 4 more simulations (see Table 5.2).

Table 5.2 shows most of the designs evaluated during the optimization process. Points from 1 to 9 correspond to the initial DOE (2-level full factorial DOE and central point). Points 10–11, 12–13, 14–15, 16–18 and 19–22 are added during the border approximation (5 different edges of the domain). Points 23 and 24 are associated to the middle points added between the best corner of the border (point 18) and the 2 best remaining adjacent corners (points 22 and 13). Points 25–27 are added during the phase of exploration along the feasible/unfeasible border by GAs with proximity penalty. Finally, points 28 and 29 are added in 2 different executions of the final GA.

5.6 Conclusions

A new lightweight optimization method for cellular structures in AM has been presented, based on a 2-level full factorial DOE and central point, border approximation along the edges, addition of new middle points between the best border corner and the best "n-1" adjacent remaining corners, addition of new points along the feasible/unfeasible border using GAs with proximity penalty and LIDT metamodel, and a final optimal searching through a GA also combined with LIDT metamodel.

The border approximation phase along the edges allows achieving good designs with a low sampling effort in the case of a small number of design variables. Moreover, in many cases, the optimum is on the boundary of the domain. For this reason, the border approximation phase along the edges is a good and simple strategy for problems with a small number of design variables.

The proximity penalty in the GA allows the addition of new points along the feasible/unfeasible border (interesting zones). These points improve the fitting of the surrogate model in areas where the optimal will be found. Hence, in the next executions of the GAs, the algorithm leads to solutions closer to the theoretical optimum thanks to the refinement achieved during this stage of the program.

Finally, it should be also noted that the linear interpolation metamodel drastically reduced the FEM simulations, obtaining a methodology that guarantees convergence to the optimal design with a low sampling density.

Although this proposal achieves good results in lightweight optimization of cellular structures for Additive Manufacturing parts, further research must be conducted in the future to further reduce the number of FEA and consequently allow design cost savings. In addition, new strategies must be developed in order to apply this concept in problems with a larger number of design variables.

References

1. Chu C, Graf G, Rosen DW (2008) Design for additive manufacturing of cellular structures. Comput Aided Des Appl 5(5): 686–696. doi:10.3722/cadaps.2008.686-696
2. Dressler M (2009) Art of surface interpolation. In: PhD thesis: Technical University of Liberec, Faculty of Mechatronics and Interdisciplinary Engineering Studies, Kunštát, 2009. http://m.dressler.sweb.cz/AOSIM.pdf
3. Gätzi R, Uebersax M, König O (2000) Structural optimization tool using genetic algorithms and ansys. In: Proceedings 18 CAD-FEM User's Meeting, Internationale FEM-Technologietage, Graf-Zeppelin-Haus, Friedrichshafen, 2000. http://www.powerpac.ethz.ch/pub-pat/optimierung_mittels_genetischen_algorythmen.pdf
4. González-López JI (2012) Modelización de sólidos para el análisis isogeométrico. Trabajo Fin de Máster. Máster Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería, Universidad de Las Palmas de Gran Canaria, 2012. http://acceda.ulpgc.es/handle/10553/7600

5. Lee JH, Hwang SC, Park JH, Lee KH (2010) Structural design examples using metamodel-based approximation model. In: Proceedings of the 9th WSEAS International Conference on Applied Computer and Applied Computational Science, World Scientific and Engineering Academy and Society (WSEAS), pp 153–156. http://www.wseas.us/e-library/conferences/2010/Hangzhou/ACACOS/ACACOS-26.pdf
6. Melcón CDF, Piñeiro-Barcia M (2004) Superficies de Respuesta Métodos y Diseños, Métodos y diseños. In: Diseño de Experimentos. Departamento de Estadística e Investigación Operativa y Didáctica de la Matemática, 2004. http://www.fbcb.unl.edu.ar/laboratorios/ladaq/curso_TopQuim_2013/Bibliografia%20RSM/superficie%20de%20respuesta%201.pdf
7. Villatoro M, Henríquez C, Sancho F (2008) Comparación de los interpoladores IDW y Kriging en la variación espacial de pH, Ca, CICE y P del suelo. Agronomía Costarricense 32(1): 95–105. http://estudiosterritoriales.org/resumen.oa?id=43632109

Chapter 6
Calibration of Dynamic Models of Railway Bridges Based on Genetic Algorithms

Diogo Ribeiro, Rui Calçada and Raimundo Delgado

Abstract This chapter presents the main experimental calibration methodologies of finite element numerical models, with particular focus on methodologies based on modal parameters. In this context, the computational implementation of an iterative method based on a genetic algorithm is described. The iterative method involves the resolution of an optimization problem, which involves the minimization of an objective function by varying a set of preselected model parameters. The objective function includes residuals associated to natural frequencies and mode shapes. The proposed methodology is applied to the calibration of the dynamic models of two railway bridges, São Lourenço bridge and Alverca viaduct, both located in the northern line of the Portuguese railways in recently upgraded track sections. The calibration results demonstrate a very good agreement between numerical and experimental modal responses and a significant improvement of the numerical models before calibration. Also the stability of a significant number of parameters, considering different initial populations, proved the robustness of the genetic algorithm in the scope of the optimization of the numerical models. The updated numerical models were validated based on dynamic tests under railway traffic. The results showed an excellent agreement between numerical and experimental responses in terms of displacements and accelerations of the bridges' decks.

Keywords Railway bridges · Numerical models · Modal parameters · Experimental calibration · Genetic algorithm · Validation

D. Ribeiro (✉)
School of Engineering, Polytechnic of Porto, Porto, Portugal
e-mail: drr@isep.ipp.pt

R. Calçada · R. Delgado
Faculty of Engineering, University of Porto, Porto, Portugal
e-mail: ruiabc@fe.up.pt

R. Delgado
e-mail: rdelgado@fe.up.pt

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*, Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2_6

6.1 Introduction

Railway bridges are structures subjected to high intensity moving loads, where the dynamic effects can reach significant values. At present, these effects are being given greater importance due to the increase of the circulation speed, not only in conventional lines but also in new lines, such as the case of high speed railway lines.

In structures with complex behaviour the evaluation of these effects is performed by means of dynamic analyses using finite element (FE) models. The process of developing a FE model of a structure involves assumptions and simplifications that may cause errors. These errors are usually related to the inaccuracy in the FE model discretization, uncertainties in geometry and boundary conditions and variation in the material properties.

Therefore, the accuracy of the FE model strongly depends on the experimental validation of the numerical results that is usually performed by means of static or quasi-static measurements based on load tests [11], dynamic measurements based on ambient vibration or forced vibration tests [8, 13], or a combination of static and dynamic measurements [25]. In recent years, in situ dynamic testing has been used and reported by several authors [12, 28] in the scope of the identification of the modal parameters of structures, namely the natural frequencies and mode shapes. Experimental modal data is also perturbed by measurement errors typically related with the environmental variability (such as temperature and wind), the variability in operational conditions during the measurements (e.g. traffic) and errors with measured signals and post-processing techniques [6, 31]. Despite the presence of the referred errors it is generally assumed that the experimental data is a better representation of the structural behaviour than the initial estimations from the FE model [9].

Finite element model updating, also known as calibration of a finite element model, is a procedure to determine uncertain parameters in the initial model based on experimental results to achieve a more suitable updated model of the structure [10]. Updated models can be used for the prediction of dynamic responses under new load scenarios, for damage identification, to design health monitoring systems, as well as for improved remaining lifetime predictions [2, 12]. There are basically two distinct finite element model updating methodologies in structural dynamics: the direct [29] and the iterative methods [27, 33].

The direct methods directly update the elements of the stiffness and mass matrices in a one-step procedure. In this method the experimental modal properties can be exactly represented by the updated system matrices. Unfortunately, the updated system matrices have little physical meaning, and cannot be related to physical properties of the finite element model [10]. This approach can also lead to non-sparse and non-positive definite system matrices, where the connectivity relations between the different structural elements, in which the structure was discretized, can be disregarded.

The iterative methods are typically related to a penalty function, which is improved by a step-by-step approach. The penalty function denotes the objective function based on the discrepancy between numerically obtained and experimentally derived features, such as natural frequencies or modal deflections. This approach is more flexible in its application as the uncertain physical properties of the FE model, typically material properties or geometrical dimensions can be updated.

Brehm [2] distinguished the solving algorithms for model updating in sensitivity-based methods and optimization-based methods. The sensitivity-based methods, applied in the works developed by Friswell and Mottershead [10], Teughels [26], Jaishi and Ren [13], Huang et al. [12], among others, depends on the update of the sensitivity matrices to proceed to the next iteration. The sensitivity matrices, also referred as Jacobian matrices, contain the first derivatives of each residue of the objective function with respect to the parameters of the numerical model. Brehm [2] reports that the success of this approach requires that the initial numerical parameters values are close to the optimal solution, and also its variation during the optimization process occurs in reduced intervals. Furthermore the selection of a large number of numerical parameters, and the coexistence of parameters with high and low sensitivity in relation to the responses, can cause numerical errors due to poor conditioning of the sensitivity matrix [4, 26]. Teughels [26] also refers that in this approach the number of responses must be equal or greater than the number of numerical parameters in order to avoid numerical errors. Some of the refereed limitations may be minimized by performing modifications to the gradient-based algorithm [26, 27] or by applying regularization techniques [10].

The optimization-based methods are generally more flexible since it does not require the calculation of sensitivities and are particularly adequate to situations where there are uncertainties in the objective function, or objective functions with multiple local minima. Applications of such methods in the scope of model updating of railway bridges were referred by Chellini and Salvatore [7], Liu et al. [14] and Cantieni et al. [5]. Concerning the optimization algorithm, several methods are available to solve the optimization problem. These include gradient-based methods (quasi-Newton, sequential quadratic programming, augmented Lagrangian, etc.) [26], response surface methods [21] and nature inspired algorithms (e.g., genetic algorithm, evolutionary strategies, particle swarm optimization) [30]. The genetic algorithm, used in the present work, is not a regularly reported and referenced methodology in the scope of model updating, particularly in the field of model updating of bridges based on experimental vibration data. In this specific research topic, the research of Cantieni et al. [5] and Zabel and Brehm [31] should be emphasized. Genetic algorithms have recognized advantages such as the non-dependence of the initial starting point, capability to manage a large number of parameters and constraints, possibility to handle with discrete and binary variables, ability to find the global minimum in functions with several local minima and the possibility to accept failed designs. On the other hand, a low convergence rate in comparison to gradient-based methods is generally agreed to be its main disadvantage.

More recently Brehm [2] states that the success of the numerical parameters estimate largely depends on the reliability of the experimental data and numerical

responses. This author considers that the uncertainties associated with the experimental and numerical responses estimates, can be included in the optimization problem based on statistical parameters related with coefficients of variation and associated confidence intervals. The proposed methodology, called stochastic model updating, are also referenced in the works developed by Mares et al. [16], Mottershead et al. [19] and Zabel and Brehm [32].

This chapter describes an iterative methodology for the calibration of numerical models based on genetic algorithms. This methodology is applied to the calibration of the dynamic models of two railway bridges located on the northern line of the Portuguese railways, which establishes the connection between the cities of Lisbon and Porto, in recently upgraded track sections. The calibration results of the numerical models of both bridges show a significant improvement in relation to the initial models demonstrating the efficiency and robustness of the implemented technique and particularly the genetic algorithms. The comparison of the experimental and numerical dynamic responses after calibration, in terms of displacements and accelerations on the bridges' decks for the passage of Alfa Pendular tilting train showed an excellent agreement, as well as an important improvement in relation to the initial numerical models.

6.2 Computational Implementation of an Iterative Method

The computational implementation of an iterative method based on a genetic algorithm involved the use of three software packages: Ansys [1], Matlab [17] and OptiSlang [20]. Figure 6.1 shows a flowchart that illustrates the computational implementation of the method, indicating the softwares involved in the different phases.

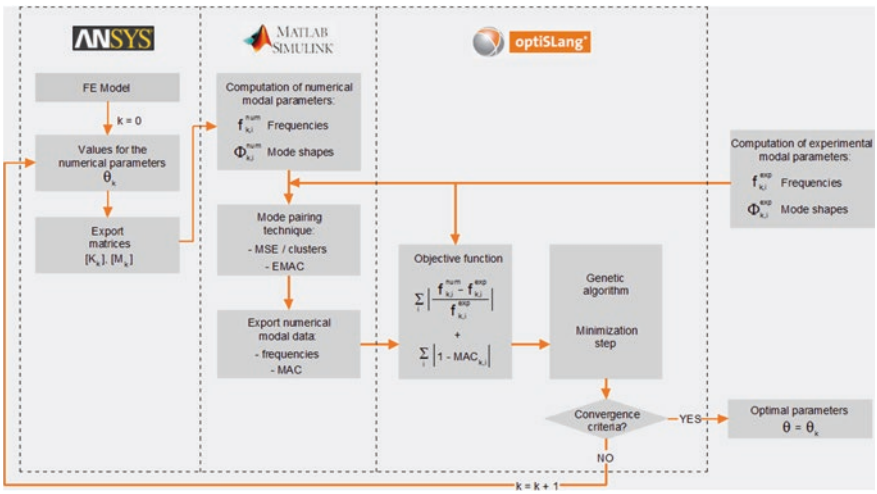

Fig. 6.1 Iterative calibration methodology based on a genetic algorithm

In ANSYS environment the FE numerical model is developed based on a set of initial parameter values θ_1, θ_2, …, θ_k, where k is the number of individuals in each generation, and the mass and stiffness matrices are extracted. The pre-selection of the calibration parameters is performed based on global sensitivity analysis [22]. The sets of parameter values of generation 1 are randomly generated in OptiSlang software by applying the Latin Hypercube method. The export of mass and stiffness matrices is performed through text files in Harwell-Boeing format, suitable for storing sparse matrices.

In Matlab software, the eigenvalues and eigenvectors problem is solved, and based on the experimental modal information, the mode pairing between numerical and experimental modes using a modal strain energy criterion based on EMAC parameter is performed [3]. The values of the natural frequencies and the corresponding MAC (Modal Assurance Criterion) values are exported in text format.

Finally, the OptiSlang software, based on an objective function and on the application of an optimization technique supported by a genetic algorithm, estimates a new set of parameters focused on the minimization of the objective function residuals. The objective function includes two terms, one relative to the residuals of the frequencies of vibration and other related to the residuals of modal configurations. This procedure is repeated iteratively until the maximum number of generations is reached.

6.3 São Lourenço Bridge

6.3.1 Description

São Lourenço railway bridge is located at km +158.662 of the northern line of the Portuguese railways. The bridge is a bowstring arch consisting of two half-decks with 42 m span, each one carrying a single track. Each deck consists of a 0.40 m thick prestressed concrete slab suspended by two longitudinal arches. The suspension is performed by means of metallic hangers and diagonals. The arches are linked in the upper part by transversal girders that assure the bracing of the arches.

The deck is supported at each abutment by two pot bearings. The distance between the supports is 38.4 m, and the extremities of the deck slab work as cantilevers with 1.8 m span. Each half-deck cross section, with a total width of 7.35 m, consists of a concrete slab laterally supported by two main girders, forming a U-section, and a side footway. In Fig. 6.2 a lateral view of São Lourenço bridge and a cross section of the deck are presented.

6.3.2 Numerical Model

The dynamic analysis of São Lourenço railway bridge was performed using a three-dimensional model, including the track, developed in Ansys software. A global view of the numerical model of the bridge is presented in Fig. 6.3.

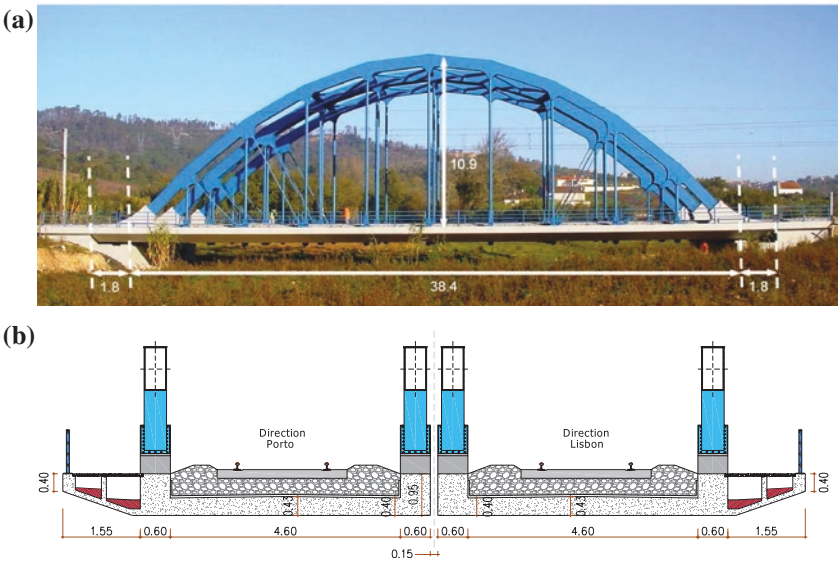

Fig. 6.2 São Lourenço bridge: **a** lateral view; **b** cross-section of the deck

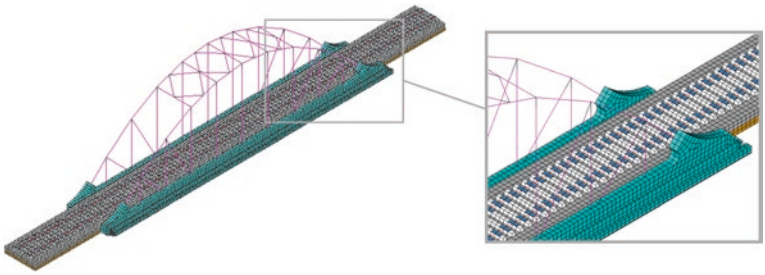

Fig. 6.3 Numerical model of São Lourenço bridge including the track

The deck slab was modelled with solid elements. The arches, hangers, diagonals and bracings were modelled with beam elements. The track was modelled in an extension corresponding to the bridge length and in a distance of about 10 m from each abutment, in order to simulate the support of the track on the adjacent embankments. The rails were modelled by beam elements levelled with the center of gravity axis, and the sleepers and the ballast layer were modelled using solid finite elements. The connections related to the support bearings were located at their centers of rotation. To correctly reproduce the deformability length of the hangers and diagonals, rigid elements were introduced in the extremities of the beam elements. The structure was divided into 26,754 nodes and 80,029 degrees-of-freedom.

Table 6.1 describes the main geometric and mechanical parameters of the numerical model of the bridge, including its designation, the adopted value and the respective unit. Additionally, the statistical properties of some of the parameters

Table 6.1 Characterization of the main parameters of the numerical model of São Lourenço bridge

Parameter	Designation	Statistical properties		Limits (lower/upper)	Adopted value	Unit
		Distribution type	Mean value/standard deviation			
E_c	Modulus deformability concrete	Normal	38.7/3.87	31.0/46.4	38.7	GPa
ρ_c	Density concrete	Normal	2446.5/97.9	2286/2607	2446.5	kg/m^3
E_{bal}	Modulus deformability ballast	–	–/–	–/–	130	MPa
ρ_{bal}	Density ballast	Uniform	1885/147.2	1630/2140	1733	kg/m^3
E_s	Modulus deformability steel	Normal	202/8.1	188.7/215.3	202	GPa
ρ_s	Density steel	–	–/–	–/–	7850	kg/m^3
K_v	Vertical stiffness of supports	Log-normal	7419/6929	–/–	3847	MN/m

that will be used later in the model calibration phase are listed. The lower and upper limits of the normal statistical distributions were obtained by subtracting or adding to the average value, a value equal to two times the standard deviation.

6.3.3 Calibration

The calibration of the bridge numerical model involved the use of 5 design variables and 24 modal responses (12 frequencies and 12 MAC values) related to global vibration modes of the deck and arches. The experimental modal parameters were obtained from an ambient vibration test described in Ribeiro et al. [23].

The genetic algorithm was based on an initial population consisting of 30 individuals and 150 generations, for a total of 4500 individuals. The initial population was randomly generated by Latin Hypercube method. In this algorithm the number of elites was equal to 1 and the number of substitute individuals was also defined equal to 1. The crossing rate was considered equal to 50 % and the mutation rate was set equal to 10 % with a standard deviation, variable along the optimization, between 0.10 and 0.01.

The optimal values of the parameters were obtained based on the results of 4 independent optimization runs (GA1 to GA4) with different initial populations. In Fig. 6.4 are represented the ratios of the values of the main numerical parameter relative to the limits indicated in Table 6.1 for optimization runs GA1 to GA4. A ratio of 0 % means that the parameter coincides with the lower limit. A ratio of

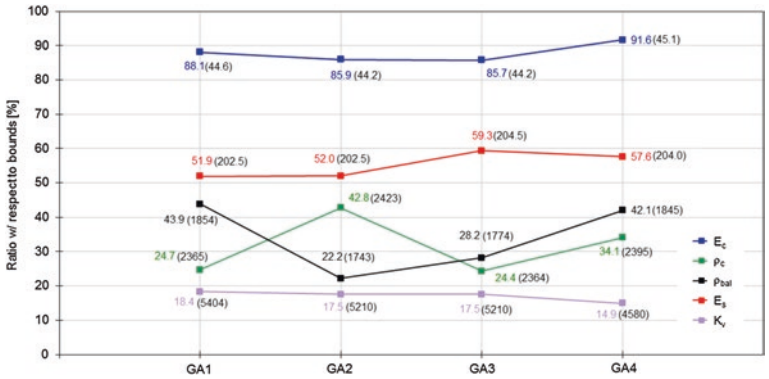

Fig. 6.4 Values of the numerical parameters for the optimization runs GA1 to GA4

100 % means that it coincides with the upper limit. The values of the numerical parameters are indicated in brackets.

The optimum value of the module of deformability of concrete situated in the range between 44 and 45 GPa, the value of the module of deformability of steel was set between 202.5 and 204.5 GPa and the value of the vertical stiffness of the supports was in the range of 4600 and 5600 MN/m. These parameters, which influence more the numeric responses, show variations always below 10 %. For the densities of the concrete and ballast, the estimates show slightly higher variations, close to 20 %. This should be related to the fact that these parameters contribute similarly to the mass of the deck and therefore it exist different combinations of these parameters that lead to the same solution in terms of the optimization problem.

Figure 6.5 presents a comparison between the experimental and numerical after calibration modal configurations of the bridge. To simplify the graphical

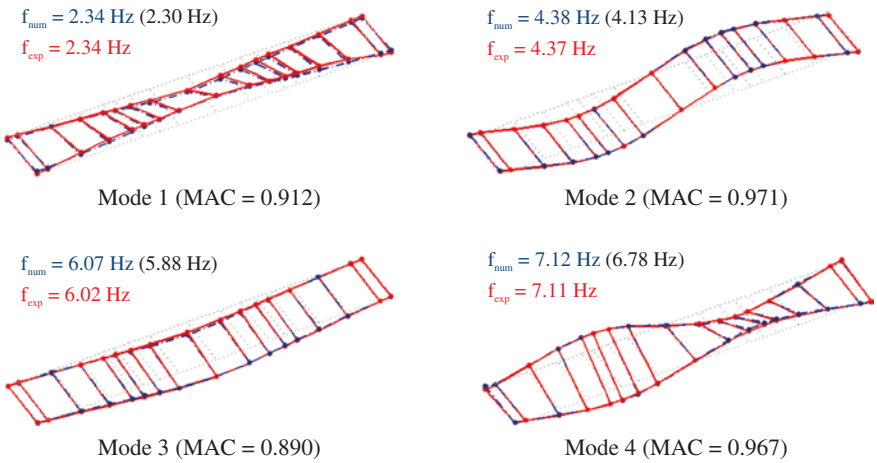

Fig. 6.5 Comparison between the experimental and numerical modal parameters

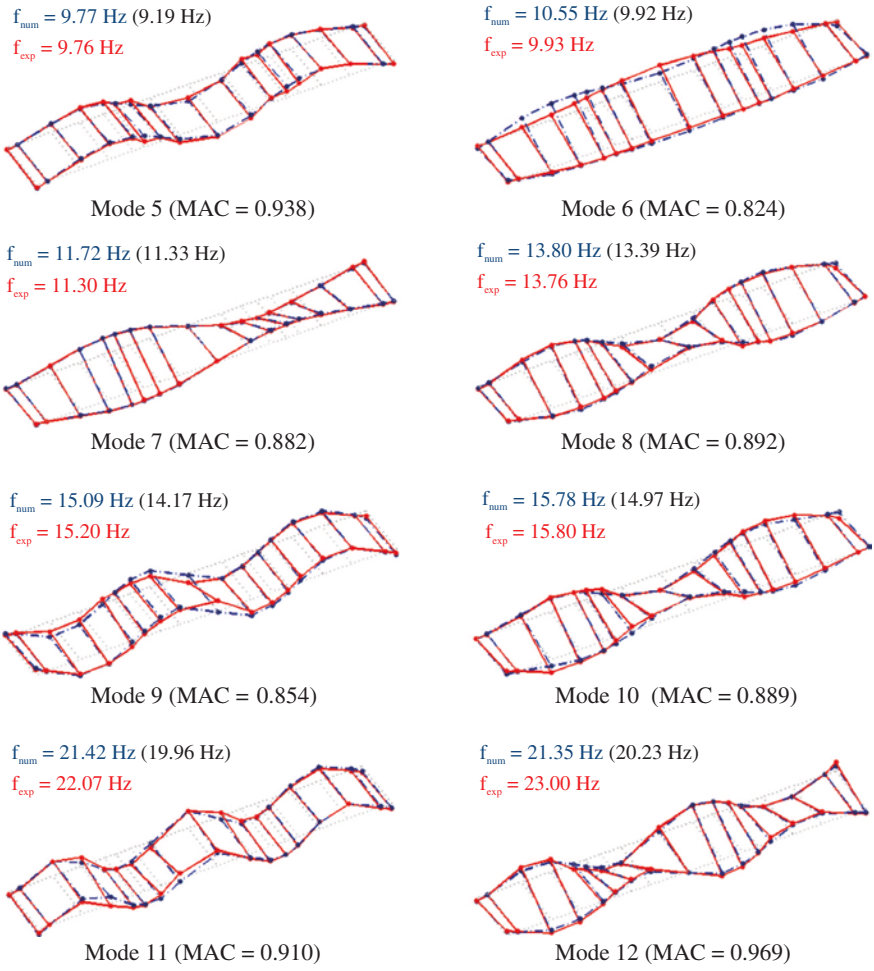

Fig. 6.5 (continued)

representation only points belonging to the deck are presented. The modal configurations are related to transversal bending modes of the arches (1, 4, and 8) and bending modes (2, 3, 5, 9 and 11) and torsion modes of the deck (6, 7, 10 and 12). The results after calibration refer to the optimization run GA1, which is associated with the lowest residual of the objective function, equal to 1.327. In the same Figure the values of the numerical frequencies after calibration (f_{num}), experimental frequencies (f_{exp}) and MAC values, are also shown. In brackets the values of the numerical frequencies before calibration, which resulted from the modal problem resolution based on the adopted values of parameters listed in Table 6.1, are also indicated.

The calibration results showed a very good agreement between numerical and experimental modal responses and a significant improvement in relation to the numerical model before calibration. The average error of the frequencies decreased from 4.7 %, before calibration, to 1.9 % after calibration. The average MAC value increased from 0.880, before calibration, to 0.908 after calibration.

6.3.4 Validation

The validation of the numerical model was performed based on a dynamic test under railway traffic which allowed evaluating the dynamic response in terms of displacements and accelerations at several locations of the bridge deck [23].

Figure 6.6 presents some details of the instrumentation used, consisting of LVDTs for measuring the displacement in one of the supports (Fig. 6.6a) and in a section between 1/3 and 1/4 span of the deck (Fig. 6.6b), and a piezoelectric accelerometer (Fig. 6.6c) positioned in the same section of the deck.

The numerical responses were obtained based on a dynamic analysis considering train-bridge interaction, including measured track irregularities, performed by TBI software [22]. In problems with train-bridge interaction, TBI software uses the modal superposition method for solving the dynamic problem of the bridge, and the Newmark method, for solving the dynamic problem of the train. The numerical model of Alfa Pendular train was calibrated based on experimental modal parameters, as described in detail in Ribeiro et al. [24]. The contribution of 85 vibration modes for the response of the bridge, with frequencies between 2.34 and 30 Hz, was considered. The time step of the analysis was equal to 0.001 s. The adopted values of the damping coefficients were equal to the average values of the coefficients obtained from an ambient vibration test [23].

Figure 6.7 compares the dynamic responses of the bridge obtained by experimental and numerical calibration, before and after updating, for the passage of

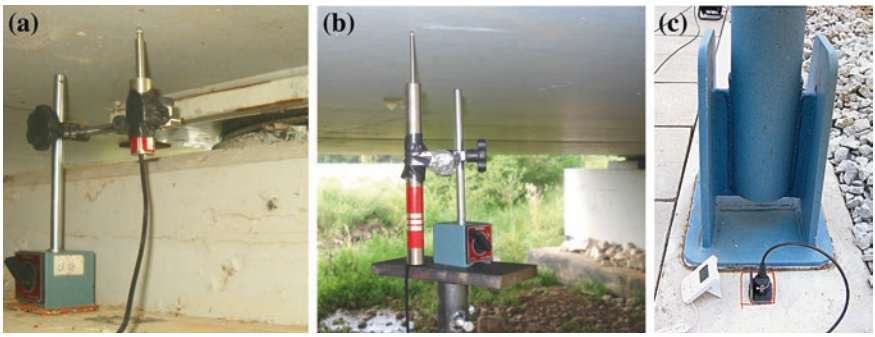

Fig. 6.6 Dynamic test under railway traffic: **a** LVDT nearby the support; **b** LVDT and **c** accelerometer, both on the main girder of the deck

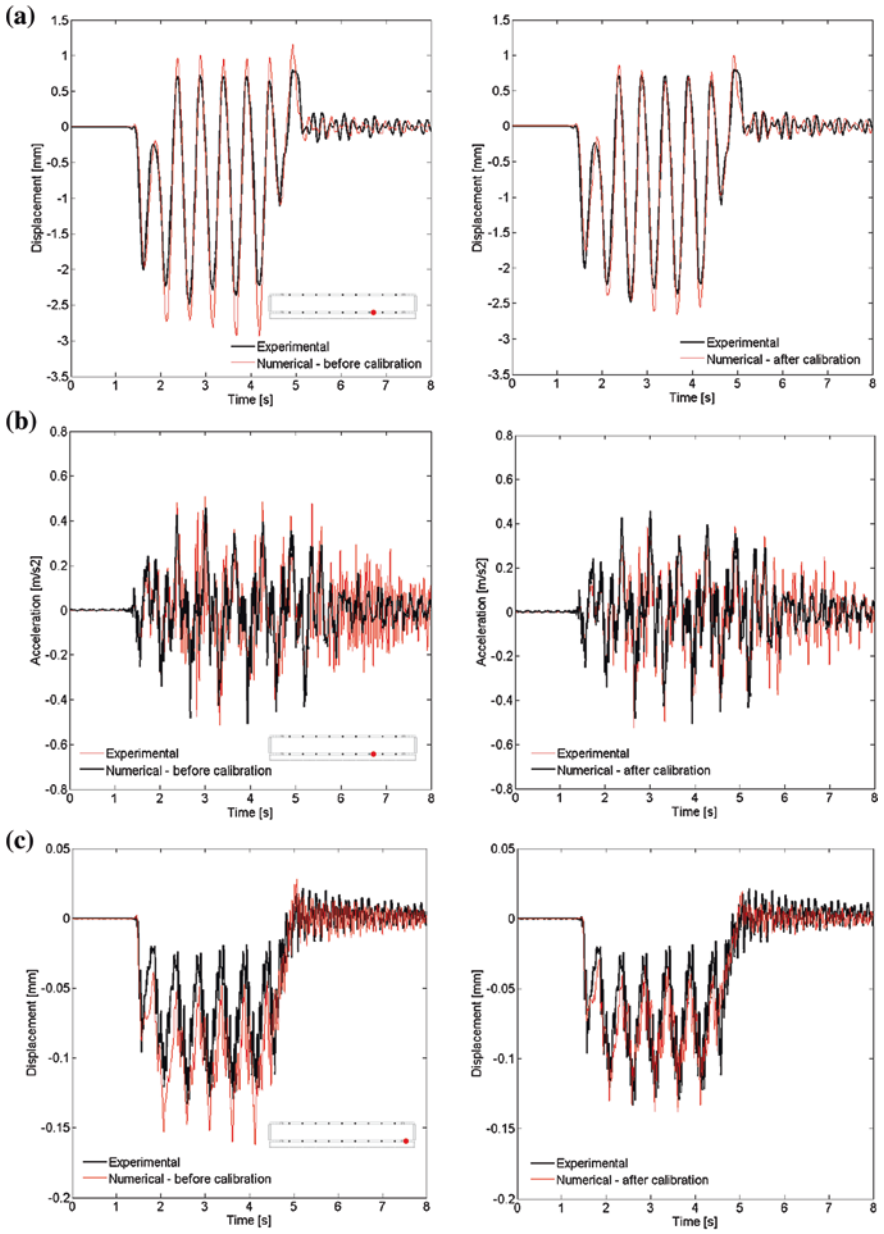

Fig. 6.7 Comparison of the experimental and numerical, before and after updating, dynamic responses of the bridge for the passage of Alfa Pendular train at a speed of 180 km/h: **a** displacements and **b** accelerations in a section between 1/3 and 1/4 span of the deck; **c** displacements at the support

Alfa Pendular train at a speed of 180 km/h. The experimental acceleration records were filtered based on a low-pass digital filter with a cut-off frequency equal to 30 Hz.

The numerical results after updating revealed a better approximation to the experimental results, in comparison with the results before updating. In this context it should be highlighted the significant improvement of the correlation between the records of displacements at deck and supports, once the model calibration process led to an overall increase of the stiffness of the structure and also the vertical stiffness of the supports. In terms of accelerations, the inclusion of the track irregularities was crucial to obtain a better agreement with the experimental results, especially for higher frequencies [22, 23].

6.4 Alverca Viaduct

6.4.1 Description

Alverca railway viaduct is a flyover structure located at km +18.676 of the northern line of the Portuguese railways. Its construction allowed to separate the rail traffic flowing in the downstream and upstream directions and also to maintain the maximum speed of trains at 200 km/h. Figure 6.8 presents a side view of the current zone of the viaduct (Fig. 6.8a) and a cross-section of the deck (Fig. 6.8b).

The viaduct has a total length of 1091 m divided into 47 simply supported spans with the following spans: 9 × 16.5 m + 9 × 17.5 m + 29 × 21.0 m. Each span supports one single railway track and is composed of a prefabricated and prestressed U shape beam on which pre-slabs serving as formwork to the concrete upper slab cast in situ were placed, forming a single-cell box-girder deck. The ballast retaining walls are monolithically connected to the upper slab of the deck.

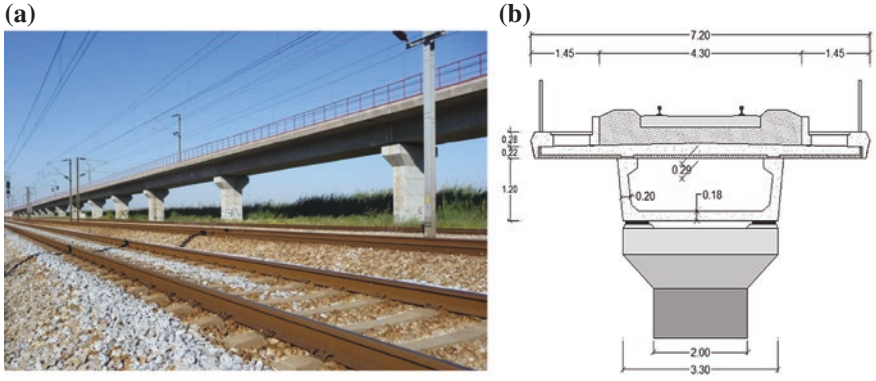

Fig. 6.8 Alverca viaduct: **a** perspective view; **b** cross-section of the deck

The deck is directly supported in the piers and in the abutments by elastomeric reinforced bearings. In each span the supports are fixed in one extremity and longitudinally guided in the other extremity. The track consists of UIC60 continuously welded rails, elastomeric rubber pads, prestressed concrete monoblock sleepers and a 30 cm ballast layer under sleepers.

6.4.2 Numerical Model

The dynamic analysis of the Alverca railway viaduct was carried out using a three-dimensional numerical model, including the track, developed in Ansys software. The analysis focused on the three spans adjacent to the north abutment: one 16.5 m long span (Span 1) and two 21 m long spans (Spans 2 and 3). Additionally, an extra extension of the track, with a length of 6 m, apart from the abutment, was modelled in order to simulate the effect of the track over the adjacent embankment. Figure 6.9 shows an overview of the numerical model with a detail of the track components.

The prefabricated beam, the upper slab and the ballast retaining walls were modelled by shell finite elements. The sleepers, the rail pads and the ballast layer were modelled by volume finite elements. The compatibility of displacements and rotations between the nodes of the precast beam and the nodes of the upper slab as well as the compatibility of displacements between the nodes of the upper slab of the deck and the lower nodes of the ballast layer were accomplished by rigid finite elements. Each support was regarded as a single point and modelled by a spring element. The rails were modelled as beam elements, positioned at their center of gravity. The non-structural elements such as safeguards and edge beams were considered as additional masses and applied to the nodes of the finite element mesh according to the real location of those elements. The numerical model of the viaduct includes 19,018 nodes and 20,906 elements. The level of refinement of the finite element mesh was optimized in order to reduce the time of modal analyses, which will have to be performed during the automatic calibration process of the numerical model.

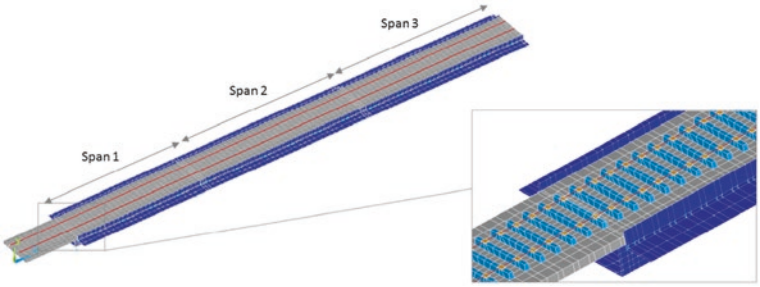

Fig. 6.9 Numerical model of Alverca viaduct including the track

Table 6.2 Characterization of the main parameters of the numerical model of Alverca viaduct

Parameter	Designation	Statistical properties		Limits (lower/upper)	Adopted value	Unit
		Distribution type	Mean value/ standard deviation			
E_{c1}	Modulus of elasticity of concrete of the upper slab (Span 1/Span 2/Span 3)	Normal	35.4/4.3	28.4/42.4	35.4	GPa
E_{c2}						
E_{c3}						
E_c	Modulus of elasticity of concrete of the prefabricated beam	Normal	40.9/4.9	32.9/49.0	40.9	GPa
ρ_c	Density of the concrete	Normal	2446.5/122.3	2245.9/ 2647.1	2469.8	kg/m^3
K_v	Vertical stiffness of supports	Uniform	5400/2020.7	1900/8900	5200	MN/m
K_{h1}	Longitudinal stiffness of the supports (Span 1/Span 2/Span 3)	Uniform	3.35/0.89	1.8/4.9	3.6	MN/m
K_{h2}						
K_{h3}						
e_{bal}	Thickness of the ballast layer	Normal	0.25/0.013	0.23/0.27	0.25	m
E_{bal}	Modulus of elasticity of the ballast	Uniform	140/34.6	80/200	145	MPa
ρ_{bal}	Density of the ballast	Uniform	1875/129.9	1650/2100	2039	kg/m^3

Table 6.2 presents the main geometrical and mechanical parameters taken under consideration in the numerical model of the viaduct, including its designation, the statistical properties and the adopted value. The lower and upper limits of each parameter are also defined, and will be taken into account during the calibration process of the numerical model.

6.4.3 Calibration

The calibration of the model involved the use of 11 numerical parameters and 12 modal responses (6 frequencies and 6 MAC values) regarding the global vibration

modes of the structure and the local vibration modes of the upper slab of the deck of span 2. The experimental modal parameters were obtained from an ambient vibration test described in Malveiro et al. [15].

The genetic algorithm was based on an initial population consisting of 30 individuals and 200 generations, for a total of 6000 individuals. In this algorithm the number of elites, as well as the number of substitute individuals, was equal to 1. The crossing rate was equal to 50 % and the mutation rate was equal to 15 % with a standard deviation varying throughout the optimization process and ranging between 0.10 and 0.01.

Figure 6.10 shows the ratios of the values of the main numerical parameter relative to the limits indicated in Table 6.2, obtained from three independent optimization runs (GA1 to GA3) based on different initial populations. The values of the numerical parameters are indicated in brackets.

The parameters that demonstrate to have more influence over modal responses, e.g., the modulus of elasticity of concrete of the precast beam and upper slab of the decks, the horizontal stiffness of supports and the density of concrete, provide estimates with lower variability. Furthermore, the parameters that less influence the modal responses, such as density and modulus of elasticity of ballast tend to present greater variation for the different optimization runs [15].

Figure 6.11 shows a comparison of the experimental and numerical after calibration modal configurations of the viaduct. The configurations presented are related to global modes of vibration of the structure (1G to 3G) and local modes of vibration of the upper slab of the deck (1L to 3L). The graphical representation of the global mods include the 3 spans of the viaduct, while for local modes only span 2 is presented. The results after calibration concern the case of optimization GA2, which was the case with lower residual of the objective function, equal to 0.556. In the same Figure the values of the numerical frequencies after calibration (f_{num}), experimental frequencies (f_{exp}) and MAC values, are also shown. In brackets the values of the numerical frequencies before calibration, which resulted from

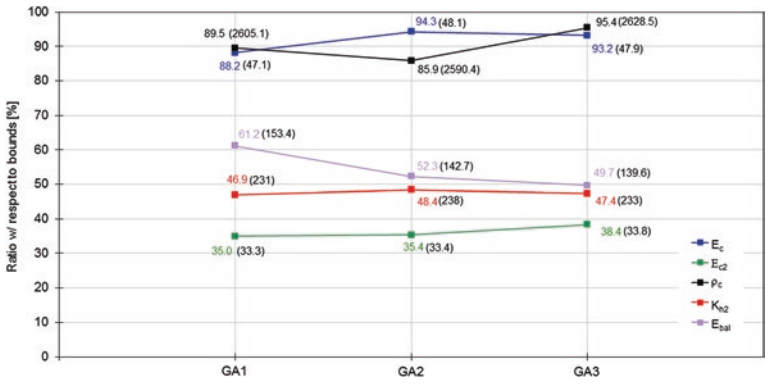

Fig. 6.10 Values of the numerical parameters for the optimization runs GA1 to GA3

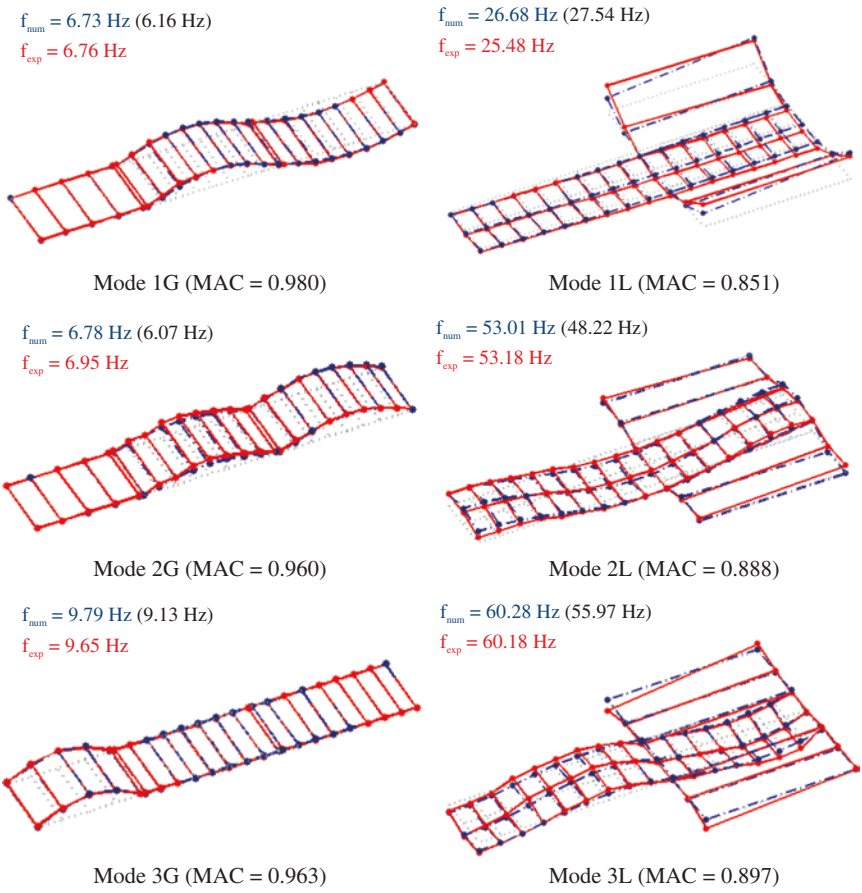

Fig. 6.11 Comparison between the experimental and numerical modal parameters

the modal problem resolution based on the adopted values of the parameters listed in Table 6.2, are also indicated.

There is a good match between the numerical and experimental modal configurations, particularly in global vibration modes. The average error of global modes frequencies decreased from 9.0 % before calibration to 1.5 % after calibration. The average value of the MAC parameter remains the same, before and after calibration, and equal to 0.968. For local modes, the average error of the frequencies went from 8.1 %, before calibration, to 1.7 % after calibration, while the average value of the MAC parameter increased from 0.784, before calibration, to 0.879 after calibration.

6.4.4 Validation

The validation of the numerical model was based on the results of a dynamic test under traffic actions. This test allowed the evaluation of the dynamic response in terms of displacements and accelerations at the mid-span section of the deck slab of span 2, for the passage of Alfa Pendular train at 185 km/h [15, 18].

The vertical displacement was measured on the lower slab of the deck by a LVDT positioned by means of a metallic tripod fixed on the ground (Fig. 6.12a). The vertical acceleration was measured on the upper slab (Fig. 6.12b) and lower slab (Fig. 6.12c) of the deck using two piezoelectric accelerometers. The accelerometer located on the upper slab involved the installation of a metallic protection tube in the interior of the ballast layer. The accelerometer installed on the lower slab was fixed by means of metallic plates bonded to the surface of the concrete.

The numerical responses were obtained based on a dynamic analysis considering train viaduct interaction, including measured track irregularities, performed by TBI software. The numerical model of the viaduct used for the dynamic analysis involved a reduction of the longitudinal stiffness of the guided supports in order to correctly reproduce the mobility of the supports under traffic actions [15]. The numerical model of Alfa Pendular train used in the analyzes is described by Meixedo et al. [18]. The contribution of 33 vibration modes for the response of the viaduct, with frequencies between 6.73 and 30 Hz, was considered. The time step of the analysis was equal to 0.001 s. The adopted values of the damping coefficients were equal to the average values obtained from an ambient vibration test [15].

Figure 6.13 presents a comparison between the numerical and experimental time records of the vertical displacement of the lower slab and vertical accelerations of

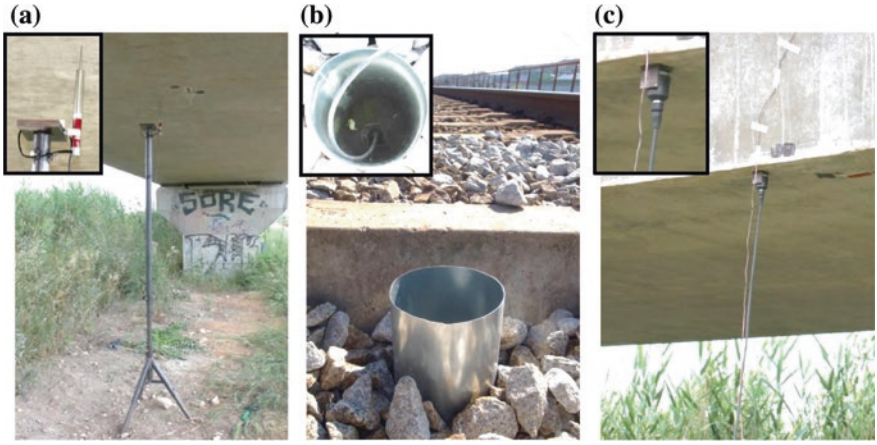

Fig. 6.12 Dynamic test under railway traffic: **a** LVDT on the deck; **b** accelerometer on the upper slab of the deck; **c** accelerometer on the lower slab of the deck

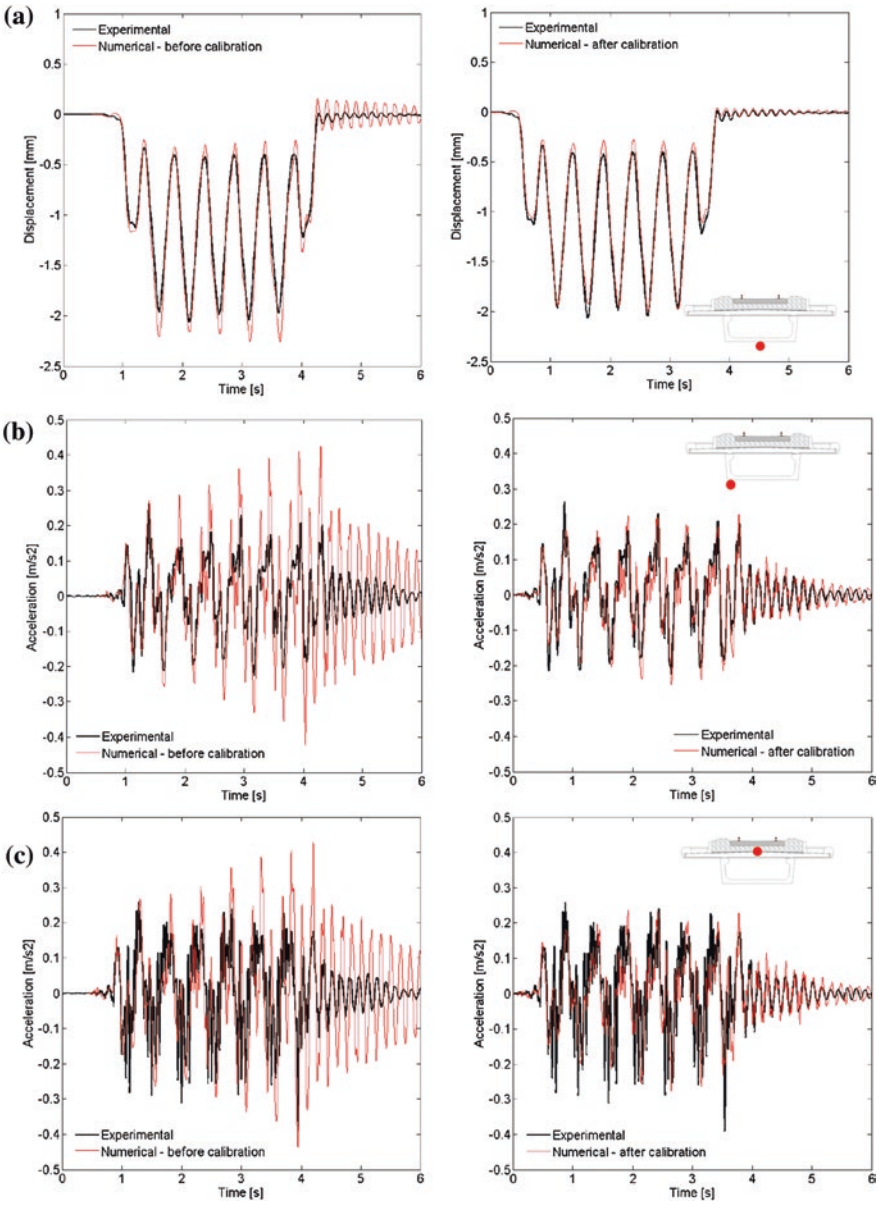

Fig. 6.13 Comparison of the experimental and numerical, before and after updating, dynamic responses of the viaduct for the passage of Alfa Pendular train at a speed of 185 km/h, at the midspan section of the deck: **a** displacements; **b** accelerations in the lower slab; **c** accelerations in the upper slab

the upper and lower slabs, in the mid-span section of span 2, for the passage of Alfa Pendular train at 185 km/h. The experimental records were filtered based on a low-pass digital filter with a cut-off frequency equal to 30 Hz.

The results show a very good agreement between experimental and numerical records after calibration, as well as a significant improvement in relation to the numerical records before calibration. The dynamic responses, in particular in terms of displacement of the deck, are clearly dominated by the frequency associated with the passage of the regularly spaced groups of axles (f) with a spacing (d) of 25.9 m ($f = v/d = 185/3.6/25.9 = 1.98$ Hz).

Regarding the response in terms of acceleration, besides the contribution of the frequency related to the train action, it should be noted the important contribution of the vibration mode 1G at a frequency of 6.47 Hz. This contribution is particularly evident in the record before calibration where inclusively is visible an amplification of the dynamic response. The response is also influenced by the contribution of higher frequencies, above 15 Hz, mainly related to the track irregularities [15, 18].

6.5 Conclusions

In this chapter the computational implementation of an iterative method for calibration of FE numerical models based on genetic algorithms, and its application to the dynamic models of two railway bridges, São Lourenço bridge and Alverca viaduct, was described.

The developed computational implementation allows performing an automatic and efficient calibration of numerical models based on experimental data, particularly modal parameters, and based on the interoperability between three software packages: Ansys, Matlab and OptiSlang.

The calibration results of the numerical models of both bridges conducted to stable estimates of the main numerical parameters and modal responses, considering different initial populations, and therefore demonstrating the robustness and efficiency of genetic algorithms in complex optimization problems with a large number of variables. The results also showed a significant improvement of the numerical models before calibration, as attested by the important reduction in the mean value of the errors associated with the frequencies of vibration and MAC parameters.

The validation of the numerical models involved the comparison of the measured responses from dynamic tests under railway traffic, with the numerical responses obtained through TBI software, based on train-bridge dynamic interaction models including track irregularities. The experimental and numerical records of displacements and accelerations obtained on the decks of both bridges showed a very good agreement, especially for the models after calibration.

In future works, the calibrated numerical models of São Lourenço bridge and Alverca viaduct may serve as basis for the implementation of advanced damage

detection techniques. These techniques, based on numerical simulations, will rely on dynamic performance indicators associated to the train-track-bridge system, namely those associated to the traffic stability, track stability and passengers comfort. Based on these studies some recommendations that can support the decisions of infrastructure managers, with impact on reducing the costs of inspection and maintenance of bridges and on increased safety in operation, will be established.

Acknowledgments The authors would also like to thank to REFER, the company responsible for the management of the Portuguese railway network, for all the information provided about São Lourenço bridge and Alverca viaduct and for all the support given during the experimental tests, as well as the support provided by CSF, Centre of Competence in Railways of the Faculty of Engineering of the University of Porto.

References

1. ANSYS (2007) Structural analysis guide—release 11.0. In: ANSYS Incorporate, editor
2. Brehm M (2011) Vibration-based model updating: reduction and quantification of uncertainties. Ph.D. thesis, Bauhaus-Universitat Weimar, Weimar, Germany
3. Brehm M, Zabel V, Bucher C (2010) An automatic mode pairing strategy using an enhanced modal assurance criterion based on modal strain energies. J Sound Vib 329:5375–5392
4. Brownjohn J, Xia P, Hao H, Xia Y (2001) Civil structure condition assessment by FE model updating: methodology and case studies. Finite Elem Anal Des 37:761–775
5. Cantieni R, Brehm M, Zabel V, Rauert T, Hoffmeister B (2008) Ambient modal analysis and model updating of a twin composite filler beam railway bridge for high-speed trains with continuous ballast. In: IMAC-XXVI conference on structural dynamics. Orlando, USA
6. Chan THT, Li ZX, Yu Y, Sun ZH (2009) Concurrent multi-scale modeling of civil infrastructures for analyses on structural deteriorating—Part II: model updating and verification. Finite Elem Anal Des 45:795–805
7. Chellini G, Salvatore W (2007) Updated models for steel-concrete composite HS railway bridges. In: Conference experimental vibration analysis for civil engineering structures (EVACES). Porto, Portugal
8. Cunha Á, Caetano E, Calçada R, Roeck GD, Peeters B (2003) Dynamic measurements on bridges: design, rehabilitation and monitoring. Bridge Eng 156:135–148
9. Ewins DJ (1984) Modal testing: theory and practice. Research Studies Press, London
10. Friswell MI, Mottershead JE (1995) Finite element model updating in structural dynamics. Kluwer Academic Publishers, Boston
11. Frýba L, Pirner M (2001) Load tests and modal analysis of bridges. Eng Struct 23:102–109
12. Huang M, Guo W, Zhu H, Li L (2008) Dynamic test and finite element model updating of bridge structures based on ambient vibration. Front Archit Civ Eng China 2:139–144
13. Jaishi B, Ren W-X (2005) Structural finite element model updating using ambient vibration test results. J Struct Eng 45:617–628
14. Liu W, Gao W, Sun Y, Xu M (2008) Optimal sensor placement for spatial lattice structure based on genetic algorithms. J Sound Vib 317:175–189
15. Malveiro J, Ribeiro D, Calçada R, Delgado R (2014) Updating and validation of the numerical model of a railway viaduct with precast deck based on dynamic tests. Struct Infrastruct Eng 10(11):1484–1509
16. Mares C, Friswell M, Mottershead J (2002) Model updating using robust estimation. Mech Syst Signal Process 16(1):169–183
17. Mathworks (2011) MATLAB—Getting started guide. Natick, USA

18. Meixedo A, Ribeiro D, Calçada R, Delgado R (2014) Global and local dynamic effects on a railway viaduct with precast deck. In: Railways 2014—the second international conference on railway technology: research, development and maintenance. Ajaccio, France
19. Mottershead J, Mares C, James S, Friswell M (2006) Stochastic model updating: Part 2—application to a set of physical structures. Mech Syst Signal Process 20(8):2171–2185
20. OptiSLang (2008) OptiSLang—the optimizing Structural Language, 3.0 edn. In: GmbH D (ed) Weimar, Germany, 2008
21. Ren W-X, Chen H-B (2010) Finite element model updating in structural dynamics by using the response surface method. Eng Struct 32:2455–2465
22. Ribeiro D (2012) Efeitos Dinâmicos Induzidos por Tráfego em Pontes Ferroviárias: Modelação Numérica, Calibração e Validação Experimental. Ph.D. thesis, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal [in Portuguese]
23. Ribeiro D, Calçada R, Delgado R, Brehm M, Zabel V (2012) Finite element model updating of a bowstring-arch railway bridge based on experimental modal parameters. Eng Struct 40:413–435
24. Ribeiro D, Calçada R, Delgado R, Brehm M, Zabel V (2013) Finite element model updating of a railway vehicle based on experimental modal parameters. Veh Syst Dyn 51(6):821–856
25. Schlune H, Plos M, Gylltoft K (2009) Dynamic finite element model updating of prestressed concrete continuous box-girder bridge. Earthq Eng Eng Vib 31:1477–1485
26. Teughels A (2003) Inverse modelling of civil engineering structures based on operational modal data. Ph.D. thesis, Katolic University of Leuven, Leuven, Belgium
27. Teughels A, Roeck GD, Suykens J (2003) Global optimization by coupled local minimizers and its application to FE model updating. Comput Struct 81:2337–2351
28. Wenzel H, Pichler D (2005) Ambient vibration monitoring. Wiley, Chichester
29. Yang YB, Chen YJ (2009) A new direct method for updating structural models based on measured modal data. Eng Struct 31:32–42
30. Yu X, Gen M (2010) Introduction to evolutionary algorithms. Springer, Heidelberg
31. Zabel V, Brehm M (2009a) System identification of high-speed railway bridges. In: Gmbh D (ed) Weimar optimization and stochastic days 50, Weimar
32. Zabel V, Brehm M (2009b) Stochastic model updating methods. In: IMAC XXVI conference on structural dynamics, Orlando, USA
33. Zong Z-H, Jaishi B, Ge J-P, Ren W-X (2005) Dynamic analysis of a half-through concrete-filled steel tubular arch bridge. Eng Struct 27:3–15

Chapter 7
A Procedure for Improving the Acoustic Efficiency of Top-Edge Devices on Noise Barriers: An Application of Genetic Algorithms and Boundary Elements

Rayco Toledo, Juan J. Aznárez, Orlando Maeso and David Greiner

Abstract A procedure for improving the acoustic efficiency of barriers featuring top-edge devices is conducted. This methodology is based on the maximization of the *insertion loss* of candidate profiles proposed by an evolutionary algorithm. The complexity normally associated with these devices raises the need to consider some geometric simplifications in order to ease the shape optimization processes. In this way, the overall barrier configuration is modeled as both thickness and non-thickness bodies (the boundary thickness is neglected), as representatives of very thin elements. Such an idealization requires a complementary formulation to the classical Boundary Element Method (BEM) that allows the problem to be solved. Numerical results are presented both validating the formulation and on the basis of simulations on noise barriers with complex diffuser-type tops by using a 2D Dual BEM code. Results obtained show the suitability of idealizing complex barrier configurations as models featuring thickness and non-thickness boundaries.

Keywords Noise barriers · Thick and very thin bodies · Shape optimization · Genetic algorithms · Dual boundary element formulation

7.1 Introduction

The inclusion of sound barriers for abating the negative effects of road traffic noise near residential areas is a broadly used strategy. Considerable research works and studies focused on sound diffraction around barriers have been carried

R. Toledo (✉) · J.J. Aznárez · O. Maeso · D. Greiner
Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería—SIANI,
Edificio Central del Parque Científico Tecnológico del Campus Universitario de Tafira,
Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas, Spain
e-mail: rayco.toledo102@alu.ulpgc.es

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*,
Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2_7

out in the past two decades, specifically in the prediction of the performance and the development of more efficient designs. Among all of the different theoretical methods proposed concerning the issue, the Boundary Element Method (BEM hereinafter) has been previously used by the authors of this work [14] in the analysis of complex barrier configurations.

Evolutionary Algorithms (EA) have been widely used for Shape Design Optimization problems in numerous Engineering fields. The combined use of optimization problems using EA with a BEM code has been implemented in sound barriers design problems within the institute where this work is developed [6, 8–10, 27]. Many other noteworthy works concerning the issue can be found in the literature (see e.g. [1, 7, 11]).

A procedure for the shape design optimization of noise barriers by coupling BEM with an evolutionary algorithm is conducted in the work here presented. Two-dimensional sound propagation problems concerning an infinite, coherent monofrequency source of sound, placed parallel to an infinite noise barrier that stands on a flat plane (ground) of uniform admittance are studied. The sound propagation analysis is performed in the frequency domain with the usual assumptions: the medium (air) is modeled as homogeneous, elastic and isotropic with no viscosity, under small disturbances and initially at rest with no wind effects. Expression of the objective function to be maximized throughout the shape optimization process is written in terms of this response.

With the purpose of raising the acoustic performance, numerous and innovative designs have been proposed and studied in the literature for compensating the limitations normally associated with the parameter with greatest influence on the barrier efficiency: the effective height. For their unquestioned benefits for scattering sound field, the use of diffusers has been the subject of many reviews and studies in indoor acoustic projects. However, despite their indoor-oriented application, the use of such devices on noise barriers in outdoor acoustic problems has evidenced a good performance when compared with both a vertical screen and other top configurations. Some noteworthy works concerning the use of diffusers installed on the barrier top can be found in the literature. Such is the case of those based on mathematical number sequence such as Quadratic Residue Diffusers (QRDs) [16–18, 21] and Primitive Roots Difussers (PRDs) [19]. Other outdoor-oriented diffusers featuring elaborated configurations can be found in [22, 23] (see Fig. 7.1).

As an application, numerical results on the basis of barrier configurations featuring complex diffuser-type tops modeled as both thickness and non-thickness bodies are performed. Such an idealization requires a complementary formulation (based on the Hypersingular Boundary Integral Equation) to the classical BEM that allows the problem to be solved. This simplification of reality greatly facilitates the geometric definition of barrier profiles, having no major influence on the acoustic performance [5]. The coupling of an evolutionary algorithm with the Dual BEM code allows us to obtain interesting acoustic solutions avoiding the complexity associated with the geometric generation of real structures.

The main benefit of the methodology here presented lies in the fact that it is truly convenient when dealing with complex configurations eligible for some sort

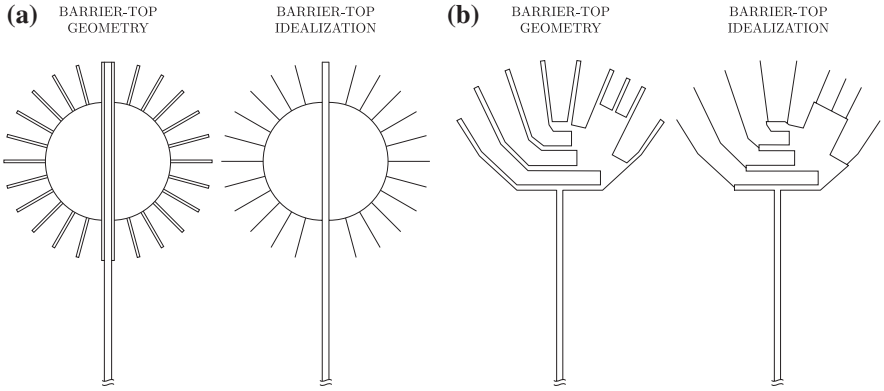

Fig. 7.1 Examples of complex designs eligible for undergoing geometric idealizations. **a** Waterwheel-top barrier from Okubo and Fujiwara [22]. **b** Complex barrier-top from Okubo and Fujiwara [23]

of geometric simplification. Such interesting models are largely present in the literature. Figure 7.1 shows some of them featuring elaborated top barrier designs.

The procedure for the geometric definition of the studied noise barrier, the fundamental aspects of the Dual BEM formulation implemented and the main features of the evolutionary algorithm software used are thoroughly and clearly explained in the next sections. Results concerning the validation of the formulation are also presented.

7.2 Schroeder Diffusers. Quadratic Residue Diffuser

A Quadratic Residue Diffuser (QRD) is just one of many possible forms of reflection phase gratings (RPGs) and the term was first used by Schroeder to describe an optimum acoustic diffuser device [25]. Its configuration is on the basis of a periodic mathematical sequence that aims at both scattering the incident wave into a wide range of directions and producing some phase variation. To accomplish this, the surface of the grating is devised with repeated structures of a series of wells equally separated from one another by mean of very thin fins. The overall number of the wells of each periodic structure is dependent on the diffusion frequency range (based on the nearest upper prime number N from the relation between the highest and the lowest value), while the separation between them and their depths are function of the highest frequency and both the lowest frequency and the pattern marked by the aforementioned numerical series, respectively. According to this pattern, the well depths are function of $i^2 \, mod \, N$, or in other words, of the modulo or *reminder* from such operation, with i being the ith well. This way, the word *quadratic* refers to the fact that the sequence is derived by use of a squared

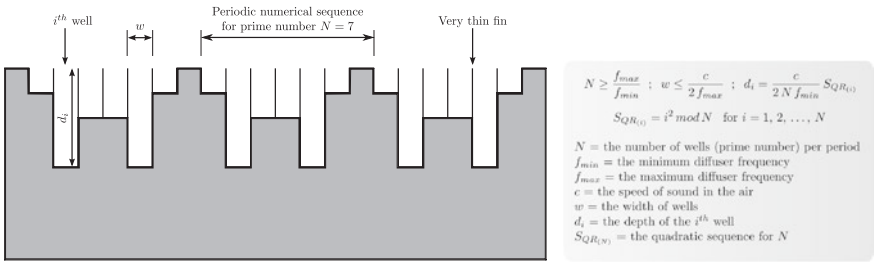

Fig. 7.2 Section of a diffuser comprised of three periods based on a quadratic sequence for N = 7. The proportional relationship of the well depths is given, in this case, by $S_{QR_{(7)}} = 0, 1, 4, 2, 2, 4, 1$. Note that the well widths are equal and the depth variation is symmetric within a period and periodic along the configuration

term, while the word *residue* is derived from the *reminder* of the mathematical operation. Further details concerning the terminology, the use, the history and the acoustic principles of these devices are provided in [4, 20].

Figure 7.2 shows a QRD section consisting of three periods based on the quadratic residue sequence derived from that of a prime number $N = 7$.

7.3 Modeling and Discretization by Implementing a Dual BEM Formulation

The next lines are focused on the description of the implemented Dual BEM formulation to make possible the numerical treatment of barriers featuring thick and very thin bodies that can be idealized as boundaries with null thickness. The special nature of these types of barriers makes necessary the addition of a complementary formulation (hyper-singular) that coupled with the conventional BEM formulation yields a compatible system of equations.

7.3.1 Singular Boundary Integral Equation (SBIE). Classical BEM Formulation

The SBIE for the boundary point i to be solved by BEM can be expressed as follows:

$$c_i p_i + \int_{\Gamma} p \frac{\partial G(k,r)}{\partial n_j} \, d\Gamma = G_0(k,r) + \int_{\Gamma} \frac{\partial p}{\partial n_j} G(k,r) \, d\Gamma \tag{7.1}$$

This integral equality just involves the boundary of the barrier under investigation. The $\oint$ symbol represents the integral along the boundary to be understood in the Cauchy principal value sense, once the singularity around the collocation point i has been extracted (c_i). In (7.1), p is the acoustic pressure field over the barrier surface and $G(k, r)$ is the half-space fundamental solution (the acoustic pressure field when the source is placed at the collocation point i over a plane with admittance β_g (ground admittance)) and c_i is the free term. As a general rule: $c_i = \theta/2\pi$, where θ represents the inner angle to the boundary measured in radians. It is easily shown that $c_i = 0.5$ for smooth boundaries.

The expressions of the fundamental solution and its derivative for a perfectly reflecting ground ($\beta_g = 0$) for bi-dimensional, harmonious problems are:

$$G(k, r) = \frac{1}{2\pi}[K_0(ikr) + K_0(ik\bar{r})] \tag{7.2}$$

$$\frac{\partial G(k, r)}{\partial n_j} = -\frac{ik}{2\pi}\left[K_1(ikr)\frac{\partial r}{\partial n_j} + K_1(ik\bar{r})\frac{\partial \bar{r}}{\partial n_j}\right] \tag{7.3}$$

being i the imaginary unit, k the wave number, and r, $\bar{r}$ the distances to the observation point from the collocation point and its symmetric point with respect to the ground plane, respectively. K_0 and K_1 are the Bessel modified functions of order 0 and 1, respectively.

By applying (7.1) on a node i of the boundary discretization and interpolating the variable with quadratic elements, the following can be written:

$$c_i \cdot p_i + \sum_{j=1}^{NE}\sum_{k=1}^{3} p_k^j \cdot h_k^{ij} = p_0^* + \sum_{j=1}^{NE}\sum_{k=1}^{3} q_k^j \cdot g_k^{ij} \tag{7.4}$$

with NE being the overall number of elements. The repeated application of (7.1) on each node of the boundary discretization leads to the following system of equations:

$$\left(\mathbf{C}^s + \mathbf{H}\right) \cdot \mathbf{P} = \mathbf{G} \cdot \mathbf{Q} + \mathbf{G_0} \tag{7.5}$$

where $\mathbf{C}^s$ is a diagonal matrix whose entries involve the free term values at the nodes of the discretization, $\mathbf{P}$, $\mathbf{Q}$, are the pressure and the flux (the derivative of the pressure with respect to the normal at each boundary node) vectors, $\mathbf{G_0}$ vector stores the values of the fundamental solution concerning the external noise source, and $\mathbf{H}$, $\mathbf{G}$ are matrices whose entries are associated with the integration cores of the BEM formulation involving just the variables of the problem along the barrier boundary:

$$h_k^{ij} = \int_{\Gamma_j} \frac{\partial G(k, r)}{\partial n_j}\phi_k \, d\Gamma_j; \quad g_k^{ij} = \int_{\Gamma_j} G(k, r)\,\phi_k \, d\Gamma_j \tag{7.6}$$

with i being the collocation point, j the observation point and ϕ_k the shape functions with quadratic approximation of the local variable ξ along the element under integration.

7.3.2 Hyper-Singular Boundary Integral Equation (HBIE)

The HBIE for the boundary point i to be solved by BEM can be written as follows:

$$c_i\left(\frac{\partial p_i}{\partial n_i}\right)+\oint\!\!\!\!\!\!=_{\Gamma} p\frac{\partial^2 G(k,r)}{\partial n_i \partial n_j}d\Gamma=\int\!\!\!\!\!\!-_{\Gamma}\frac{\partial G(k,r)}{\partial n_i}\frac{\partial p}{\partial n_j}d\Gamma+\frac{\partial G_0(k,r)}{\partial n_i} \tag{7.7}$$

where the $\oint\!\!\!\!\!\!=$ and $\int\!\!\!\!\!\!-$ symbols represent the integral along the boundary to be understood in the Hadamard finite part integral and in the Cauchy principal value sense, respectively. As the Hölder condition [15] must be satisfied at the collocation point i, the numerical treatment of the hyper-singular formulation can be conducted either (1) by using discontinuous boundary elements or (2) by moving the source towards inside the element in a non-nodal point (non-nodal collocation). In both strategies and in all situations $c_i = 0.5$ in (7.7).

Expressions (7.8) and (7.9) show the values of the derivatives of the fundamental solution implied in the hyper-singular integral Eq. (7.7):

$$\frac{\partial G(k,r)}{\partial n_i}=-\frac{ik}{2\pi}\left[K_1(ikr)\frac{\partial r}{\partial n_i}+K_1(ik\bar{r})\frac{\partial \bar{r}}{\partial n_I}\right] \tag{7.8}$$

$$\begin{aligned}\frac{\partial^2 G(k,r)}{\partial n_i \partial n_j}=\frac{(ik)^2}{2\pi}\Bigg[&\left(K_2(ikr)\frac{\partial r}{\partial n_i}\frac{\partial r}{\partial n_j}+\frac{1}{r}K_1(ikr)\,n_i\cdot n_j\right)\\&+\left(K_2(ik\bar{r})\frac{\partial \bar{r}}{\partial n_I}\frac{\partial \bar{r}}{\partial n_j}+\frac{1}{\bar{r}}K_1(ik\bar{r})\,n_I\cdot n_j\right)\Bigg]\end{aligned} \tag{7.9}$$

As in (7.1), i is the imaginary unit, k the wave number and r, $\bar{r}$ the distances to the observation point from the collocation point and its symmetric point with respect to the ground plane, respectively. It is worth making a distinction here regarding the normal vectors involved in the expressions above. n_j is the normal to the boundary at the integration point and $n_i\,(n_x^i, n_y^i)$, $n_I\,(n_x^i, -n_y^i)$ represent the normal vectors to the real boundary at the collocation point (i) and at its symmetric point (I) placed on a fictitious, symmetric boundary with respect to the ground plane, respectively. K_1 and K_2 represent the Bessel modified functions of order 1 and 2, respectively.

By applying (7.7) on a node i of the boundary discretization and interpolating the variable with quadratic elements, the following can be written:

$$c_i \cdot q_i + \sum_{j=1}^{NE} \sum_{k=1}^{3} p_k^j \cdot m_k^{ij} = \frac{\partial G_0(k,r)}{\partial n_i} + \sum_{j=1}^{NE} \sum_{k=1}^{3} q_k^j \cdot l_k^{ij} \tag{7.10}$$

with *NE* being the overall number of elements. The repeated application of (7.7) on each node of the boundary discretization, as in the classical formulation, leads to the following system of equations:

$$\mathbf{M} \cdot \mathbf{P} = (\mathbf{L} - \mathbf{C}^h) \cdot \mathbf{Q} + \frac{\partial \mathbf{G_0}}{\partial n_i} \tag{7.11}$$

where $\mathbf{C}^h$ is a diagonal matrix with entry values of 0.5, **P**, **Q** are the pressure and the flux (the derivative of the pressure with respect to normal at each boundary node) vectors, $\frac{\partial \mathbf{G_0}}{\partial n_i}$ array stores the values of the derivative of the fundamental solution concerning the external noise source, and **M**, **L** are matrices whose entries are associated with the integration cores of the hyper-singular BEM formulation involving just the variables of the problem along the barrier boundary:

$$m_k^{ij} = \int_{\Gamma_j} \frac{\partial^2 G_0(k,r)}{\partial n_i n_j} \phi_k \, d\Gamma_j; \quad l_k^{ij} = \int_{\Gamma_j} \frac{\partial G_0(k,r)}{\partial n_i} \phi_k \, d\Gamma_j \tag{7.12}$$

The numerical strategies employed in the evaluation of both the singular and the hyper-singular BEM integrals have been developed and implemented in a computer code by following the patterns proposed by Sáez et al. [24].

7.3.3 Dual BEM Formulation

Two are the Dual BEM formulations depending on the nature of the problem under study and the benefit concerning the use of such a strategy. The nature of these problems can be categorized as follows: (1) the mitigation of the fictitious eigenfrequencies; (2) the numerical modeling of thin geometries. Both difficulties have strong presence in the barrier models studied in this work. The features of the formulation for both problems are described in detail in the next subsections.

7.3.3.1 Dual BEM for Avoiding Fictitious Eigenfrequencies

Some undesirable problems may arise at certain frequencies when dealing with non-thin boundaries. These mathematically-related effects reveal the eigenfrequencies of the indoor acoustic problem (the eigenvalues of the classical BEM matrices) and may seriously affect the barrier performance. With the purpose of solving this problem, Burton and Miller [2] propose a Dual BEM formulation

based on the combined use of both the SBIE and the HBIE related by means of a frequency-related complex value. The expression for the boundary point i to be solved by BEM can be written then:

$$c_i\,(p_i + \alpha\, q_i) + \sum_{j=1}^{N} \left(h_j + \alpha\, m_j\right) p_k^j = \sum_{j=1}^{N} \left(g_j + \alpha\, l_j\right) q_j + \left(G_0 + \alpha\, \frac{\partial G_0}{\partial n_i}\right) \quad (7.13)$$

being N the overall nodes number of the discretization over the boundary. The most commonly used value of $\alpha = i/k$ [12], being i the imaginary unit and k the wave number. For smooth boundaries $c_i = 0.5$. For the SBIE, a non-nodal collocation at unbound extremes of boundaries is demanded. Furthermore, the hypersingular formulation of the method demands the collocation point j to be inside the element. This way, the free term is assumed as 0.5 in all cases.

The absorptive capacity of the barrier boundary is usually determined by means of the Robin boundary condition, so the pressure value and its derivative at each node are related:

$$q_j = -ik\beta_\Gamma p_j \quad (7.14)$$

This way, (7.13) can be written matricially:

$$\left(\frac{1}{2}\,(1+\beta)\,\mathbf{I} + \mathbf{H} + (i/k)\mathbf{M} + (ik\,\mathbf{G} - \mathbf{L})\,\beta\right) \cdot \mathbf{P} = \mathbf{G}_0 - (i/k)\,\frac{\partial \mathbf{G}_0}{\partial n_i} \quad (7.15)$$

with $\mathbf{I}$ being the identity matrix.

7.3.3.2 Dual BEM for Very Thin Bodies

The nature of the issue is different when dealing with very thin boundaries. In this case, numerical integration problems may appear affecting, equally, to the barrier performance. The idealization of such boundaries as non-thickness bodies not only solves the problem but also contributes to ease their geometric representation. With this aim, the SBIE and the HBIE are applied separately.

Figure 7.3a represents an idealization of a generic thin body to be solved by the Dual BEM formulation. After a discretization process, each node holds the values of pressure and flux with respect to the boundary normal (p^+, q^+, p^-, q^- hereinafter). Figure 7.3b represents the strategy used to isolate the singularity of the method in this type of domains. Thus, the expression of the BEM formulation for these boundaries can be written as follows:

$$c_i\left(p_i^+ + p_i^-\right) + \sum_{j=1}^{N} \left(H_j^+ p_j^+ + H_j^- p_j^-\right) = \sum_{j=1}^{N} \left(G_j^+ q_j^+ + G_j^- q_j^-\right) + G_0(k, r) \quad (7.16)$$

being N the overall nodes number of the discretization over the boundary. Taking into account that $n^+ = -n^-$, it is easily shown that:

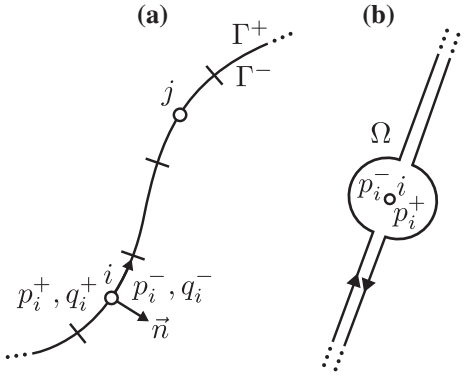

Fig. 7.3 **a** Idealization of a generic thin-cross section noise barrier profile as null thickness boundaries. **b** Strategy used to avoid the singularity around the collocation point in the dual BEM formulation (see e.g. [13])

$$H_j^+ = -H_j^-; \quad G_j^+ = G_j^- \tag{7.17}$$

Considering the Robin boundary condition, the following matrix expression is obtained:

$$\left(\frac{1}{2}\mathbf{I}^* + \mathbf{H} + i\,k\,\beta\,\mathbf{G}\right) \cdot \mathbf{P} = \mathbf{G}_0 \tag{7.18}$$

being $\mathbf{I}^*$ a matrix with the following form:

$$\mathbf{I}^* = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 \end{bmatrix} \tag{7.19}$$

that allows us to consider the contribution of the free term at both sides of the discretization nodes.

According to Fig. (7.3), the hyper-singular expression concerning these types of geometries can be written as follows:

$$c_i\left(\frac{\partial p_i^+}{\partial n_i^+} + \frac{\partial p_i^-}{\partial n_i^+}\right) + \sum_{j=1}^{N}\left(M_j^+ p_j^+ + M_j^- p_j^-\right) = \sum_{j=1}^{N}\left(L_j^+ q_j^+ + L_j^- q_j^-\right) \tag{7.20}$$

where:

$$\frac{\partial p_i^-}{\partial n_i^+} = -q_i; \quad M_j^+ = -M_j^-; \quad L_j^+ = L_j^- \tag{7.21}$$

Considering the Robin boundary condition, the following matrix expression is obtained:

$$\left[ik\beta\left(\frac{1}{2}\mathbf{I}^{*}+\mathbf{L}\right)+\mathbf{M}\right]\cdot\mathbf{P}=\frac{\partial\mathbf{G}_0}{\partial n_i} \tag{7.22}$$

7.3.3.3 Dual BEM for Barriers Featuring Both Thick and Very Thin Bodies

According to the previous subsection and calling:

$$\mathbf{A}_1=\frac{1}{2}(1+\beta)\,\mathbf{I}+\mathbf{H}+(i/k)\,\mathbf{M}+(ik\,\mathbf{G}-\mathbf{L})\,\beta \tag{7.23}$$

$$\mathbf{A}_2=\frac{1}{2}\mathbf{I}^{*}+\mathbf{H}+ik\beta\mathbf{G};\quad \mathbf{A}_3=ik\beta\left(\frac{1}{2}\mathbf{I}^{*}+\mathbf{L}\right)+\mathbf{M} \tag{7.24}$$

$$\mathbf{B}_1=\mathbf{G}_0-(i/k)\frac{\partial\mathbf{G}_0}{\partial n_i};\quad \mathbf{B}_2=\mathbf{G}_0;\quad \mathbf{B}_3=\frac{\partial\mathbf{G}_0}{\partial n_i} \tag{7.25}$$

the final Dual BEM matrix expression for barriers with thin and non-thin bodies may be written as follows:

$$\begin{bmatrix}\mathbf{A}_1\\ \mathbf{A}_2\\ \mathbf{A}_3\end{bmatrix}\cdot\begin{Bmatrix}\mathbf{P}\\ \\ \end{Bmatrix}=\begin{Bmatrix}\mathbf{B}_1\\ \mathbf{B}_2\\ \mathbf{B}_3\end{Bmatrix} \tag{7.26}$$

In (7.26), $\mathbf{A}_1$ is a $N_{Thick}\times N_{Total}$ matrix and $\mathbf{A}_2$, $\mathbf{A}_3$ are $N_{Thin}\times N_{Total}$ ones, with N_{Thick} and N_{Thin} being the number of nodes concerning the discratization of thick- and very thin-type barrier boundaries, respectively, and N_{Total} the total number of nodes of the overall discretization. Consequently, $\mathbf{P}$ is a N_{Total}-dimension vector that stores the pressure value of each variable of the problem, $\mathbf{B}_1$ is a N_{Thick}-dimension vector and $\mathbf{B}_2$, $\mathbf{B}_3$ are N_{Thin}-dimension ones.

Once the variables of the problems are known, their corresponding values at any point of the domain can be easily obtained by applying

$$p^{i}=G_0(k,r)-\sum_{j=1}^{N_{Total}}\left(h_j+i\,k\,\beta\,g_j\right)p_j \tag{7.27}$$

7.4 Noise Barrier Models

The models investigated here deal with two-dimensional problems concerning an infinite, coherent mono-frequency source of sound placed on a ground with a perfectly reflecting surface ($\beta_g=0$) at 5.0 m from the axis of an infinite thin

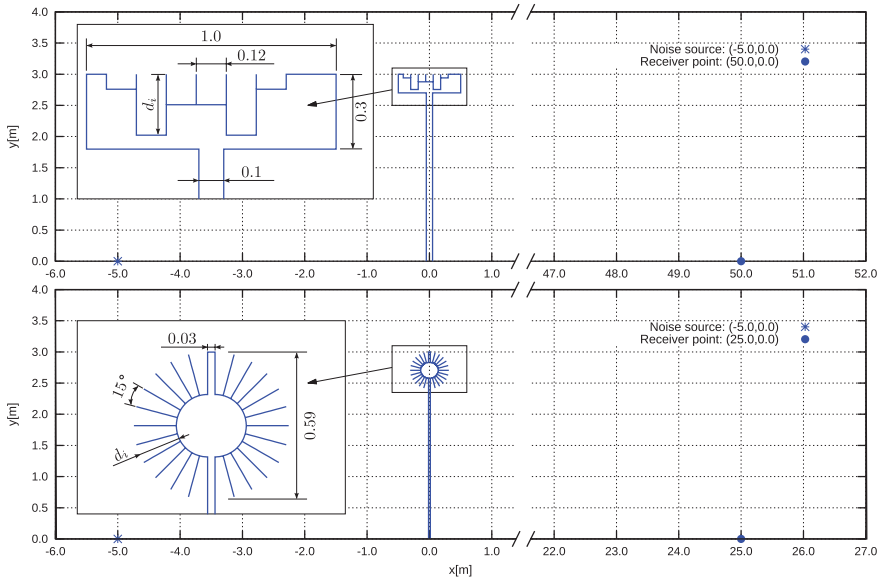

Fig. 7.4 Bi-dimensional configurations to be used in the optimization process of complex diffuser tops. *Up* model (a). Noise barrier design inspired on that by Monazzam et al. [16]: $d_1 = 0.06\,\text{m}$, $d_2 = 0.2445\,\text{m}$, $d_3 = 0.12225\,\text{m}$, $d_4 = 0.12225\,\text{m}$, $d_5 = 0.2445\,\text{m}$, $d_6 = 0.06\,\text{m}$. *Bottom* model (b). Diffuser top design based on the models by Okubo et al. [22]: $d_i = 0.17\,\text{m}$. Measurements in meters

cross-section noise barrier of 3.0 m height with a perfectly rigid surface ($\beta_b = 0$). Just one receiver point is considered, placed on the ground either at 50.0 m or at 25.0 m from the barrier axis, depending on the case (see Fig. 7.4).

Figure 7.4 represents the models under study. The upper model is a diffuser derived from the QRD configuration studied by Monazzam et al. [16]. Installed on the top of a 0.1 m width vertical stem, it deals with a 1.0 m width, 0.30 m height box comprised of six wells of 0.12 m width and different depths (d_i) which presents a symmetric configuration with respect to the median vertical axis of the diffuser. The very thin elements which separate two consecutive wells are idealized as null thickness-type. The model at the bottom is inspired on the so called *waterwheel cylinder*-diffuser types studied by Okubo et al. [22]. It is based on a constant-radial diffuser top of 0.59 m diameter comprised of two semi-circular cores from which a uniform distribution of very thin elements idealized as null bodies are born for a specific well depth d_i. The diffuser lies on the top of a 0.03 m width vertical stem.

In the harmonic problem, for every frequency from the analyzed noise source, the effectiveness of the barrier design under study is given in terms of the *insertion loss* (IL), defined as usual:

$$\text{IL} = -20 \cdot \log_{10}\left(\frac{\text{P}_\text{B}}{\text{P}_\text{HS}}\right)[\text{dB}] \tag{7.28}$$

on every frequency of the band spectrum, and represents the difference of sound pressure levels at the receiver points in the situation with (P_B) and without (P_{HS}) considering the barrier.

With the purpose of conducting an optimization process where the excitation is represented by a noise source pulsing at every frequency of the band spectrum, the efficiency of the barrier for a specific receiver can be written as:

$$\overline{IL} = -10 \cdot \log_{10}\left(\frac{\sum_{i=1}^{NF} 10^{(A_i - IL_i)/10}}{\sum_{i=1}^{NF} 10^{A_i/10}}\right)[\text{dBA}] \tag{7.29}$$

being NF the studied spectrum number of frequencies, A_i the spectrum A-weighted noise level and IL_i the insertion loss value for sources pulsing at every frequency of the spectrum, according to (7.28). For guaranteeing the acoustic intensity along the spectrum to be the same, the normalized traffic noise for 1/3-octave band center frequencies proposed by the UNE-EN 1793-3:1998 [28] (the same used by the Spanish Technical Building Code (CTE) [3]) is expanded to the 1/15-octave bands both to fit the spectrum applied in the works of reference in this paper [16, 22] and to reproduce as accurately as possible the high frequency dependence of such designs along the optimization process. The noise source then is characterized according to this latter spectrum ranging from 100 to 5 000 Hz. The expression for the objective function to be maximized in such cases can be written as:

$$OF = \max(\overline{IL}) \tag{7.30}$$

7.5 Shape Optimization

Shape design optimization is carried out by the combined use of an evolutionary algorithm and a code that implements a Dual BEM formulation. The evolutionary algorithm software used in this work applies the GAlib package [29]. This library is a collection of C++ genetic algorithm (GA) components from which it is possible to quickly construct GA's to attack a wide variety of problems.

In this paper, preliminary results with limited computational resources have been executed. Therefore, a high exploitative strategy with high selection pressure has been taken into account: a steady-state genetic algorithm [26, 30] is used replacing the worst individual (in terms of its objective function) at each generation, with a population size of 50 individuals. A single-point crossover operator is used in this study, with a crossover rate of 0.9. The considered mutation rate is $1/n_{ch}$, where n_{ch} is the chromosome length. Given the symmetric nature of the models under study, $n_{ch} = 3 \times n$ for model (a) and $n_{ch} = 12 \times n$ for model (b), being n the precision of the binary variables. Five independent runs of the

optimization process are considered for each model. The stopping criterion condition is met for 250 generations. Even in this scenario, however, the results obtained are able to improve the cases of reference.

7.6 Application of the Proposed Methodology to the Assessment of the Acoustic Efficiency of Different Barrier Designs

The proposed methodology is applied to the study of two elaborated top barrier designs featuring both thick and very thin bodies in this section. With the purpose of highlighting the robustness and flexibility of the methodology, the range of possible geometric designs to be assessed and validating the formulation here presented, such models have been chosen on the basis of complex diffuser tops previously studied in other works. Model (a) deals with a QRD-inspired diffuser top by Monazzam et al. [16] (upper graph of Fig. 7.4). Model (b) is based on the so called *waterwheel cylinder* tops studied by Okubo et al. [22] (bottom graph of Fig. 7.4). In both cases, the optimization process aims at searching for the optimum design according to the design variable d_i, or in other words, the depth of each well. This make, for example, model (b) present complex configurations with semi-circular cores featuring different arches for each well. The symmetric-nature of the designs from which these models are inspired remains along the shape design optimization of the diffuser tops.

Due to the limited computational resources and the high demanding cost within the frame of the considered 1/15 octave band center frequencies, a discrete shape optimization arises as an adequate, valid engineering optimum design process. It is performed in both models by selecting discrete variables from a list representing the set of possible diffuser well depths d_i (see Fig. 7.4) to be considered:

The well depths range with precision of centimeters for model (a) has been chosen with the purpose of ensuring that a barrier profile similar than that from Monazzam et al. [16] is contemplated as one more possible design within the discrete optimization process. On the other hand, it is a matter of fact that the well depth is directly related to higher shielding efficiency for values equal to n-quarter of the wavelength and, then, is inversely proportional to the so called *cut-off* or *design* frequencies. Therefore, this desired effect does unfortunately occur just for particular frequencies. The set of four different well depths for model (b) has been chosen with the aim of detecting these events within the frequency range proposed in this work (Table 7.1).

Table 7.1 Diffuser well depths d_i (in meters) used in the discrete shape optimization

Model	d_i (m)
(a)	0.25, 0.24, 0.23, 0.22, 0.21, 0.20, 0.19, 0.18, 0.17, 0.16, 0.15, 0.14, 0.13, 0.12, 0.11, 0.10, 0.09, 0.08, 0.07, 0.06, 0.05, 0.04, 0.03, 0.02, 0.01, 0.00
(b)	0.25, 0.17, 0.10, 0.05

7.7 Results and Discussion

7.7.1 Validation of the Methodology

Numerical model validations of the introduced methodology on the basis of results present in the reference works from which model (a) and (b) are inspired are carried out. The analysis is conducted by using the introduced Dual BEM formulation to predict and compare the acoustic performance of a QRD (Fig. 7.5) and a waterwheel diffuser top (Fig. 7.6) with the classical BEM results from [16, 22], respectively. In accordance with the results presented in such works, the validation is performed on the basis of the IL for model (7.1) and the normalized sound pressure level (SPL) (that referenced at a distance of 1 m from the noise source in the free field) for model (b), similarly to the results presented in the work from which this model is derived, both for 1–15 octave band center frequencies. Both reference barriers are 3 m height and other measurements can be consulted in Fig. 7.4.

Figures 7.5 and 7.6 display the comparison results between the Dual BEM formulation and numerical outcomes (using the classical BEM formulation) from the aforementioned works. It can be proved that the boundary thickness plays an important role in the differences observed in the numerical results depicted. This issue is particularly relevant for barriers featuring a large number of very thin boundaries, as in the case of the sticks arranged around the top surface of model (b). However, the similarity in tendencies exhibited by both graphs suggests that

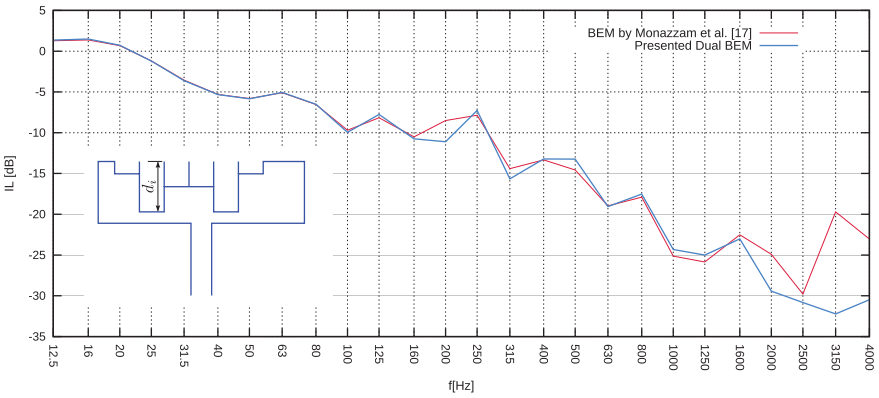

Fig. 7.5 Validation of the presented dual BEM formulation. Comparative results with those by Monazzam et al. [16]: $d_1 = 0.06\,\text{m}$, $d_2 = 0.2445\,\text{m}$, $d_3 = 0.12225\,\text{m}$, $d_4 = 0.12225\,\text{m}$ m, $d_5 = 0.2445\,\text{m}$, $d_6 = 0.06\,\text{m}$. Noise source at $(-5.0, 0.0)$. Receiver at $(50.0, 0.0)$

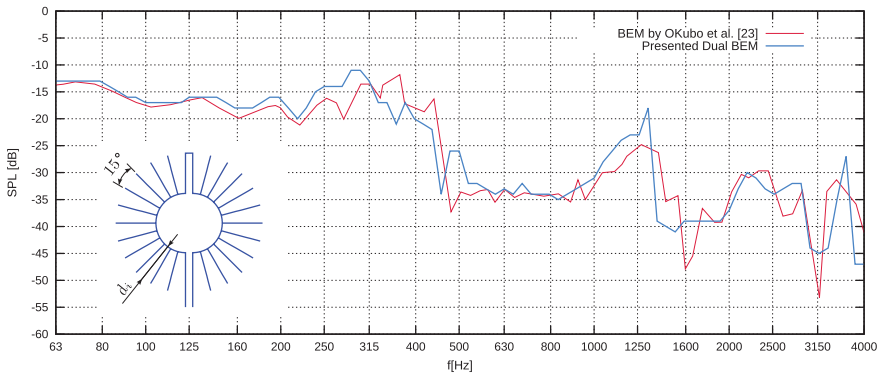

Fig. 7.6 Validation of the presented Dual BEM formulation. Comparative results with those by Okubo et al. [22]: $d_i = 0.17\,\text{m}$. Noise source at $(-5.0, 0.0)$. Receiver at $(25.0, 0.0)$

the Dual BEM formulation is a convenient, adequate alternative to the classical BEM when implemented in methodologies like the one introduced in this work.

7.7.2 Acoustic Performance of the Introduced Models

Results are shown for the best optimum individuals from the optimization processes for both models. Figure 7.7 shows the model (a) optimum profile for the given noise source-receiver scheme and its IL evolution along the considered spectrum in comparison with both a 3 m-height simple barrier and the barrier from which this model is inspired. Figure 7.8 shows the results concerning the model (b). As a complement, Fig. 7.9 shows the influence of well depths d_i (the semi-circular cores have a constant radius $\rho = [(0.59/2) - d_i]$ on the cut-off frequencies of this radial diffuser barrier top.

Table 7.2 collects the well depths of the left symmetric part of the optimum designs as well as their corresponding shielding efficiency according to (7.29), for each model and run. The $\overline{\text{DIL}}_{\text{Ref.}}$ column shows the shielding gain of the optimum designs with respect to the reference barries in this work, that is, those from [16, 22] for model (a) and model (b), respectively. Column $\overline{\text{DIL}}_{\text{Simple}}$ collects the acoustic efficiency gain of such optimums in comparison with a 3 m-height vertical screen for the considered noise source-receiver scheme in each case. The optimum profiles are depicted in Fig. 7.10.

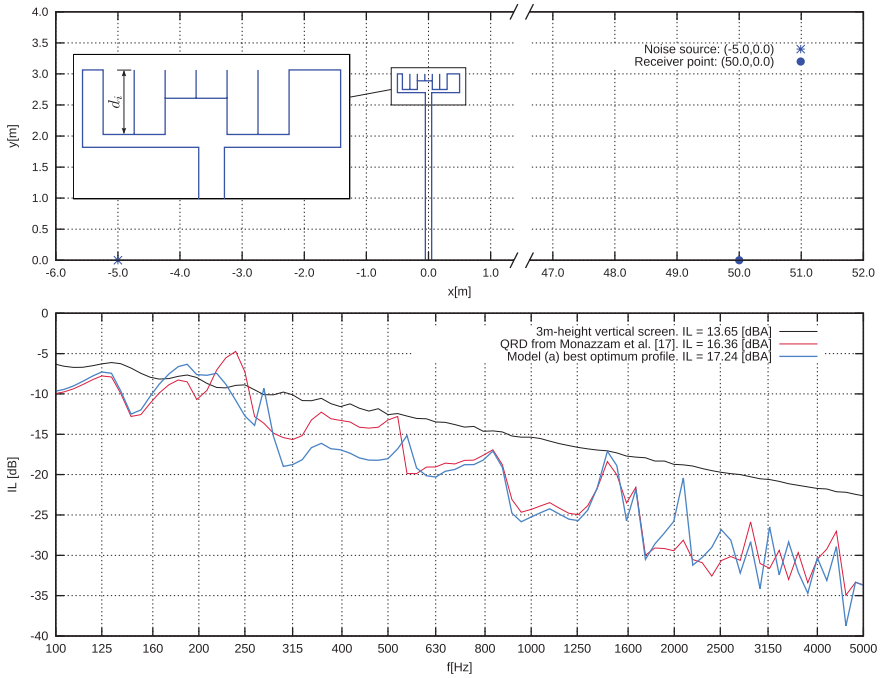

Fig. 7.7 Model (a). *Up* best optimum profile found along all discrete optimization runs (see Run #1 data in Table 7.2). *Bottom* comparative IL values between the optimum profile and that from Monazzam et al. [16]

7.7.3 Discussion

From the analysis of the results obtained some conclusions on the response of the models studied in this work may be drawn:

- The shielding efficiency of the diffuser top models studied here clearly outperform the acoustic behavior of the reference 3 m-height vertical screen. The no-so-near placement of the receiver point in both cases make this issue more remarkable.
- The strong frequency dependence-nature of these well-shaped devices poses the need of further considerations. In order to get the best affordable designs in terms of their overall screening behavior a multi-objective optimization will be performed in future work (adding the optimization of the IL curve smoothness as a new objective), as the sharp IL peaks at the cut-off frequencies may be misleading and not representatives of the corresponding octave band. Furthermore, the use of width octave bands (one-third, one-fifth, etc.) may lead to unrealistic shielding results.
- The configuration of the waterwheel has a direct impact on the cut-off frequencies, showing that the smaller the well depth the higher the frequencies with sharp IL peaks.

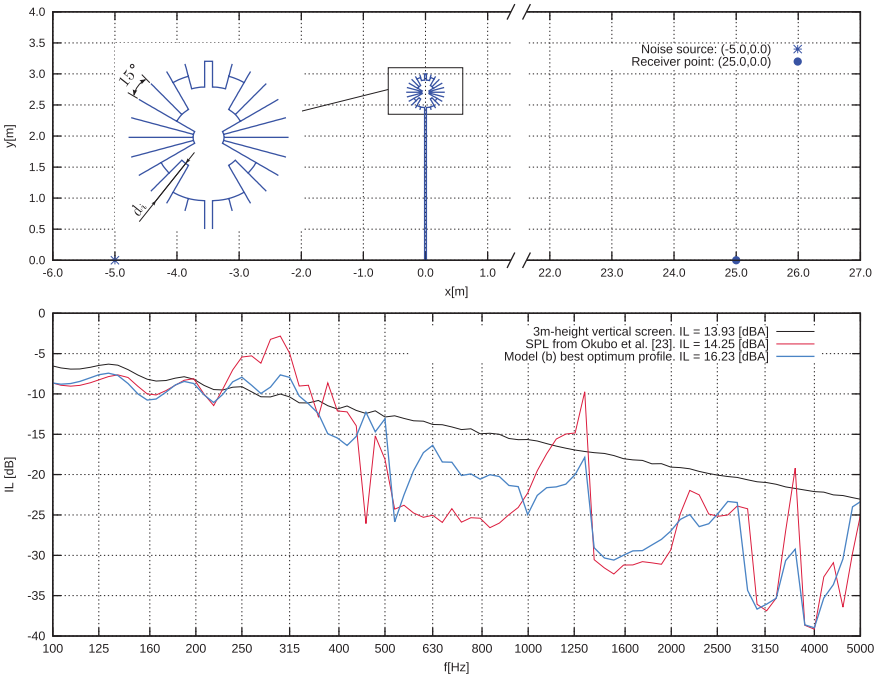

Fig. 7.8 Model (b). *Up* best optimum profile found along all discrete optimization runs (see Run #1 data in Table 7.2). *Bottom* comparative IL values between the optimum profile and that from Okubo et al. [22]

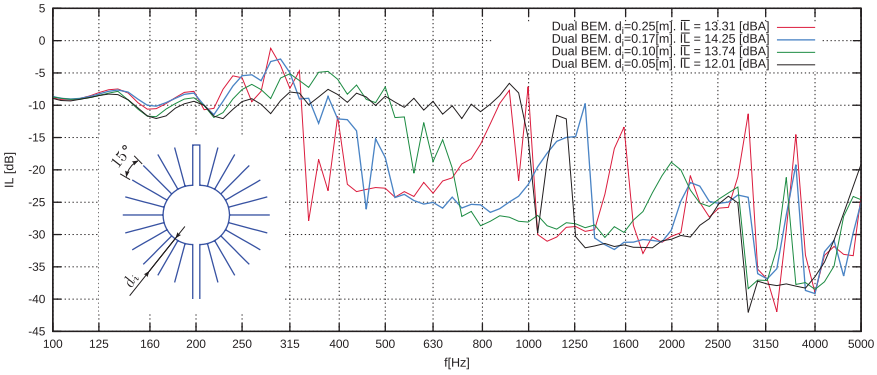

Fig. 7.9 Model (b). Influence of the well depths d_i on the cut-off frequencies. Noise source at (−5.0, 0.0) m. Receiver at (25.0, 0.0) m

Table 7.2 Design variables of the left symmetric part of the optimum diffuser configuration and shielding efficiency gain for each run and model

Model	Run	Well depth (m)												$\overline{\text{IL}}$ (dBA)	
		d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}	d_{12}	$\Delta\overline{\text{IL}}_{\text{Ref.}}$	$\Delta\overline{\text{IL}}_{\text{Simple}}$
(a)	#1	0.25	0.25	0.11	–	–	–	–	–	–	–	–	–	+0.88	+3.59
	#2	0.25	0.25	0.11	–	–	–	–	–	–	–	–	–	+0.88	+3.59
	#3	0.21	0.25	0.12	–	–	–	–	–	–	–	–	–	+0.86	+3.57
	#4	0.25	0.25	0.11	–	–	–	–	–	–	–	–	–	+0.88	+3.59
	#5	0.24	0.24	0.12	–	–	–	–	–	–	–	–	–	+0.84	+3.55
(b)	#1	0.05	0.05	0.17	0.10	0.25	0.25	0.25	0.25	0.17	0.17	0.05	0.10	+2.27	+2.30
	#2	0.10	0.05	0.10	0.17	0.25	0.25	0.25	0.25	0.25	0.10	0.17	0.05	+2.27	+2.30
	#3	0.05	0.05	0.10	0.25	0.25	0.25	0.25	0.25	0.17	0.25	0.17	0.10	+2.26	+2.29
	#4	0.05	0.05	0.17	0.10	0.25	0.25	0.25	0.25	0.25	0.17	0.17	0.10	+2.23	+2.26
	#5	0.05	0.05	0.05	0.25	0.25	0.25	0.25	0.25	0.10	0.25	0.17	0.10	+2.25	+ 2.28

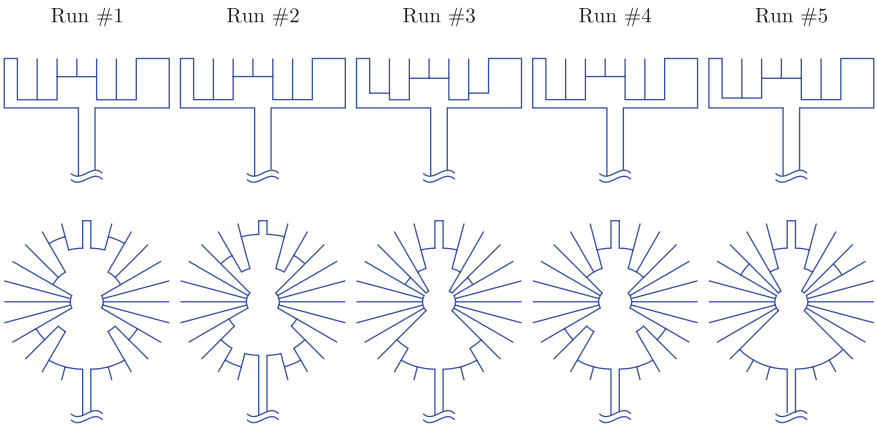

Fig. 7.10 Optimum profiles for each run. *Up* model (a). *Bottom* model (b). Please note that model (b) representation is over-dimensioned with respect to model (a) in this drawing. Further details in Table 7.2

7.8 Conclusions

A procedure for the discrete shape optimization of top barrier devices featuring both thick and very thin bodies idealized as null-thickness type has been presented. With the purpose of highlighting the robustness and flexibility of the methodology, the range of possible geometric designs to be assessed and validating the formulation here presented, two complex diffuser tops on the basis of models previously studied in other works have been proposed: a QRD-inspired [16] and a waterwheel diffuser top [22]. The strongly frequency dependent-nature of diffusers is a well known matter of fact in the literature, and its negative incidence on the overall shielding efficiency requires a thorough consideration in design optimization problems. A study concerning the cut-off frequencies of the waterwheel model has been performed for different well depths showing that the smaller the well depth the higher the frequencies with sharp IL peaks.

The versatility of the algorithm responsible for the geometry generation of the barrier makes the building of the profile to be easily accomplished. The Dual Boundary Element formulation here presented allows a simple treatment of the complex configurations. This is a significant advantage over the case when dealing with geometries of real barrier profiles, as the evaluation process for the feasibility of the design proposed by an evolutionary algorithm is often cumbersome and difficult to establish.

Acknowledgments This work was supported by the Ministerio de Economía y Competitividad (MINECO) of Spain and FEDER through research project BIA2010-21399-C02-01 and also by the Agencia Canaria de Inv., Inn. y Soc.de la Inf. (ACIISI) of the Government of the Canary Islands and FEDER through research project ProID20100224. R. Toledo is a recipient of a fellowship from the Subprogram of Predoctoral Fellowships of Research Personnel in Trainning (FPI-MICINN), granted by Ministerio de Ciencia e Innovación of Spain. The authors are grateful for this support.

References

1. Baulac M, Defrance J, Jean P (2008) Optimisation with genetic algorithm of the acoustic performance of T-shaped noise barriers with a reactive top surface. Appl Acoust 69:332–342
2. Burton AJ, Miller GF (1971) The application of integral equation methods to the numerical solution of some exterior boundary-value problems. Proc Roy Soc A 323:201–210
3. Código Técnico de la Edificación. Documento Básico HR. Protección frente al ruido. Ministerio de Fomento de España
4. Cox TJ, D'Antonio P (2009) Acoustic absorbers and diffusers theory, design and application, 2nd edn. CRC Press, London and New York
5. de Lacerda L, Wrobel L, Mansur W (1997) A dual boundary element formulation for sound propagation around barriers over an impedance plane. J Sound Vib 202(2):235–247
6. Deb K, Bandaru S, Greiner D, Gaspar-Cunha A, Celal Tutum C (2014) An integrated approach to automated innovization for discovering useful design principles: Case studies from engineering. Appl Soft Comput 15: 42–56
7. Duhamel D (2006) Shape optimization of noise barriers using genetic algorithms. J Sound Vib 297:432–443
8. Greiner D, Aznárez JJ, Maeso O, Winter G (2006) Shape design of noise barriers using evolutionary optimization and boundary elements. In: The fifth international conference on engineering computational technology, vol 43. Civil-Comp Press, Stirlingshire, UK
9. Greiner D, Galván B, Aznárez JJ, Maeso O, Winter G (2009) Robust design of noise attenuation barriers with evolutionary multiobjective algorithms and the boundary element method. In: Ehrgott M et al (eds) Evolutionary multi-criterion optimization, Lecture Notes in Computer Science, vol 5467. Springer, pp 261–274
10. Greiner D, Aznárez JJ, Maeso O, Winter G (2010) Single- and multi-objective shape design of Y-noise barriers using evolutionary computation and boundary elements. Adv Eng Softw 41:368–378
11. Grubeša S, Jambrošić K, Domitrović H (2012) Noise barriers with varying cross-section optimized by genetic algorithms. Appl Acoust 73: 1129–1137
12. Li S, Huang Q (2010) An improved form of the hypersingular boundary integral equation for exterior acoustic problems. Eng Anal Boundary Elem 34:189–195
13. Luo J, Liu Y, Berger E (1998) Analysis of two dimensional thin structures (from micro- to nano-scales) using the boundary element method. Comput Mech 22:404–412
14. Maeso O, Aznárez JJ (2005) Estrategias para la reducción del impacto acústico en el entorno de carreteras. Una aplicación del Método de los Elementos de Contorno. Universidad de Las Palmas de Gran Canaria. ISBN: 84-689- 0340-X, http://hdl.handle.net/10553/1500
15. Martin P, Rizzo F, Cruse T (1998) Smoothness-relaxation stratregies for singular and hypersingular integral equations. Int J Numer Meth Eng 42:885–906
16. Monazzam M, Lam Y (2005) Performance of profiled single noise barriers covered with quadratic residue diffusers. Appl Acoust 66:709–730
17. Monazzam M, Lam Y (2008) Performance of T-shape barriers with top surface covered with absorptive quadratic residue diffusers. Appl Acoust 69(2):93–109

18. Monazzam M, Nassiri P (2009) Contribution of quadratic residue diffusers to efficiency of tilted profile parallel highway noise barriers. Iran J Environ Health Sci Eng 6(4):271–284
19. Monazzam M, Naderzadeh M, Nassiri P, Fard S (2010) Performance of Environmental T-shape Noise Barriers Covered with Primitive Root Diffusers. Archives of Acoustics. 35(4):565–578
20. MuSonics (2009) Consultation and designs in acoustics, sound and music. Technical Note: RPGs & QRDs
21. Naderzadeh M, Monazzam M, Nassiri P, Fard S (2011) Application of perforated sheets to improve the efficiency of reactive profiled noise barriers. Appl Acoust 72:393–398
22. Okubo T, Fujiwara K (1998) Efficiency of a noise barrier on the ground with an acoustically soft cylindrical edge. J Sound Vib 216(5):771–790
23. Okubo T, Fujiwara K (1999) Efficiency of a noise barrier with an acoustically soft cylindrical edge for practical use. J Acoust Soc Am 105:3326
24. Sáez A, Gallego R, Domínguez J (1995) Hypersingular quarter-point boundary elements for crack problems. Int J Numer Meth Eng 38:1681–1701
25. Schroeder MR (1979) Binaural dissimilarity and optimum ceilings for concert halls: more lateral sound diffusion. J Acoust Soc Am 65(4):958–963
26. Syswerda G (1991) A study of reproduction in generational and steady-state genetic algorithms. In: Rawling GJE (ed) Proceedings of the foundations of genetic algorithms (FOGA). Morgan Kaufmann Publishers, San Mateo, California
27. Toledo R, Aznárez JJ, Maeso O, Greiner D (2015) Optimization of thin noise barrier designs using evolutionary algorithms and a dual BEM formulation. J Sound Vib 334:219–238
28. UNE-EN 1793-3:1998 (1997) Road traffic noise reduction devices. Test method for determining the acoustic performance. Part 3: normalized traffic noise spectrum
29. Wall M (ed) (1996) GAlib: a C++ library of genetic algorithm components, Mechanical Engineering Department, Massachusetts Institute of Technology. http://lancet.it.edu/ga/
30. Whitley D, Kauth J (1998) Genitor: a different genetic algorithm. Technical Report CS-88-101, Colorado State University

Author Index

J. Magalhães-Mendes and D. Greiner (eds.), *Evolutionary Algorithms and Metaheuristics in Civil Engineering and Construction Management*, Computational Methods in Applied Sciences 39, DOI 10.1007/978-3-319-20406-2