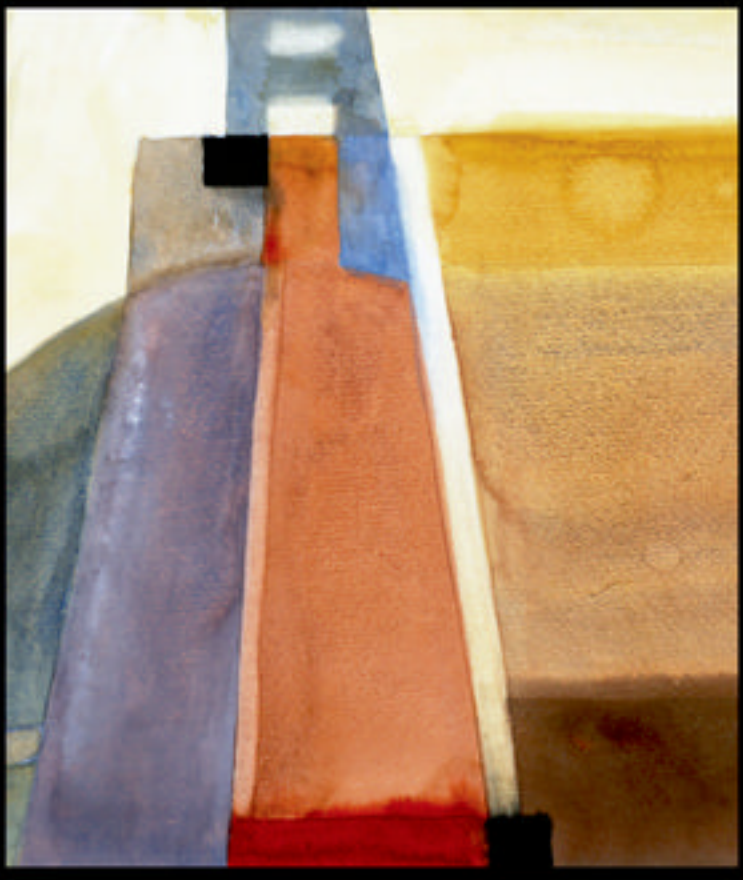

OXFORD SERIES ON COGNITIVE MODELS AND ARCHITECTURES

ANATOMY OF THE MIND

Exploring Psychological Mechanisms and Processes with the Clarion Cognitive Architecture

RON SUN

OXFORD

Anatomy of the Mind

OXFORD SERIES ON COGNITIVE MODELS AND ARCHITECTURES

Integrated Models of Cognitive Systems
Edited by Wayne D. Gray

In Order to Learn: How the Sequence of Topics Influences Learning
Edited by Frank E. Ritter, Joseph Nerb, Erno Lehtinen, and Timothy O'Shea

How Can the Human Mind Occur in the Physical Universe?
By John R. Anderson

Principles of Synthetic Intelligence PSI: An Architecture of Motivated Cognition
By Joscha Bach

The Multitasking Mind
By David D. Salvucci and Niels A. Taatgen

How to Build a Brain: A Neural Architecture for Biological Cognition
By Chris Eliasmith

Minding Norms: Mechanisms and Dynamics of Social Order in Agent Societies
Edited by Rosaria Conte, Giulia Andrighetto, and Marco Campennì

Social Emotions in Nature and Artifact
Edited by Jonathan Gratch and Stacy Marsella

Anatomy of the Mind: Exploring Psychological Mechanisms and Processes with the Clarion Cognitive Architecture
By Ron Sun

Anatomy of the Mind

Exploring Psychological Mechanisms and Processes with the Clarion Cognitive Architecture

Ron Sun

OXFORD
UNIVERSITY PRESS

OXFORD
UNIVERSITY PRESS

Oxford University Press is a department of the University of Oxford. It furthers the University's objective of excellence in research, scholarship, and education by publishing worldwide. Oxford is a registered trade mark of Oxford University Press in the UK and certain other countries.

Published in the United States of America by Oxford University Press
198 Madison Avenue, New York, NY 10016, United States of America.

First Edition published in 2016

Library of Congress Cataloging-in-Publication Data
Sun, Ron, 1960–
Anatomy of the mind : exploring psychological mechanisms and processes with the Clarion cognitive architecture / Ron Sun.
pages cm. — (Oxford series on cognitive models and architectures)
Includes bibliographical references and index.
ISBN 978–0–19–979455–3
1. Cognitive science. 2. Cognitive neuroscience. 3. Computer architecture. 4. Cognition—Computer simulation. I. Title.
BF311.S8148 2016
153—dc23
2015018557

9 8 7 6 5 4 3 2 1

Printed by Sheridan, USA

Contents

Preface

This book aims to understand psychological (cognitive) mechanisms, processes, and functionalities through a comprehensive computational theory of the human mind, namely, a computational "cognitive architecture," or more specifically, the Clarion cognitive architecture. The goal of this work is to develop a unified framework for understanding the human mind, and within the unified framework to develop process-based, mechanistic explanations of a substantial variety of psychological phenomena.

The book describes the essential Clarion framework, its cognitive-psychological justifications, its computational instantiations, and its applications to capturing, simulating, and explaining various psychological phenomena and empirical data. The book shows how the models and simulations shed light on psychological mechanisms and processes, through the lens of a unified framework (namely, Clarion).

While a forthcoming companion volume to this book will fully describe the technical details of Clarion (along with hands-on examples), the present book concentrates more on a conceptual-level exposition and explanation, but also describes, in a more accessible way, essential technical details of Clarion. It covers those technical details that are necessary for explaining the psychological phenomena discussed in this book.

The following may be considered the features of the present book:

- A scope broader than any other cognitive architecture, pointing to new possibilities for developing comprehensive computational cognitive architectures.

- Integration of multiple approaches and perspectives within this broad scope.
- Exploration of empirical data and phenomena through computational models and simulations, examining a variety of data from a variety of empirical fields.
- Balance of formal modeling and readability (i.e., accessibility to a multidisciplinary readership).

These features were designed with potential readers of the book in mind, who may include (in no particular order): (1) cognitive scientists (especially cognitive modeling researchers, or "computational psychologists" as one might call them) who might be interested in a new theoretical framework, a new generic computational model, as well as new interpretations of data through computational modeling; (2) experimental psychologists who might be interested in new possibilities of interpreting empirical data within a unified framework, new conceptual interpretations (or existing interpretations for that matter) being substantiated through computational modeling, and also new possibilities for further empirical explorations; (3) researchers from adjacent fields who might be interested in work on computational psychology (cognitive modeling) and how such research may shed light on the mind; (4) interested lay readers who might want to explore computational psychology and its implications for understanding the human mind . . . and so on. To put it simply, this book is for those who are interested in exploring and understanding the human mind through computational models that capture and explain empirical data and phenomena in a unified framework.

In fields ranging from cognitive science (especially cognitive modeling), to psychology, to artificial intelligence, and even to philosophy, academic researchers, graduate and undergraduate students, and practitioners of various kinds may have interest in topics covered by this book. The book may be suitable for graduate-level seminars or courses on cognitive architectures or cognitive modeling, but might also be suitable for the advanced undergraduate level.

A little history is in order here. The general ideas of a pair of books (this one and a companion technical book) on Clarion were drawn up in February 2009 after much rumination. I worked more on the ideas for the two books in May of that year. In November, between two trips, I wrote two book proposals. They were submitted to Oxford University

Press in January 2010. After a round of very thorough reviews of the book proposals by the publisher, the contracts for the two books were signed in May 2010. The writing of this book was sporadic and largely put off until the summer of 2011. Since that time, efforts were made to finish the book. The manuscript was sent to the publisher at the end of 2013.

The history of the Clarion cognitive architecture started, of course, much earlier than that. Back in the summer of 1994, the ONR cognitive science basic research program issued a call for proposals, which prompted me to put together a set of ideas that had been brewing in my head. That was the beginning of Clarion. The grant from the ONR program enabled the development and the validation of the initial version of Clarion. During the 1998–1999 academic year, I had my sabbatical leave at the NEC Research Institute. A theoretically oriented book on Clarion took shape during that period, which was subsequently published. Starting in 2000, research grants from ARI enabled the further development of a number of subsystems within Clarion. Then, from 2008 on, new grants from ONR enabled the extension of the work to social simulation and other related topics.

I would like to thank Frank Ritter for his solicitation of thorough reviews of the two book proposals and for his suggestions regarding the organizations of the books. Thanks also go to the eight reviewers of the book proposals for their helpful suggestions. Later I received detailed critiques of the entire book manuscript from Frank Ritter and two anonymous reviewers, whom I gratefully acknowledge as well. I would also like to acknowledge useful discussions that I have had with many colleagues, including Paul Bello, Michael Zenzen, Larry Reid, Jeff White, Jun Zhang, and Deliang Wang, regarding motivation, emotion, personality, ethics, learning, modeling, and so on. I am also indebted to my many collaborators, past and present, including Sebastien Helie, Bob Mathews, Sean Lane, Selmer Bringsjord, Michael Lynch, and their students. I also want to acknowledge my past and current graduate students: Jason Xi Zhang, Isaac Naveh, Nick Wilson, Pierson Fleischer, and others. Some other students contributed to the work on Clarion as well. The work described in this book is theirs as well as mine.

Clarion has been implemented as Java and C# libraries, available at (courtesy of Nick Wilson and Michael Lynch):

http://www.clarioncognitivearchitecture.com

Finally, the work described here has been financially supported, in part, by ONR grants N00014-95-1-0440, N00014-08-1-0068, and N00014-13-1-0342, as well as ARI grants DASW01-00-K-0012 and W74V8H-05-K-0002. Without these forms of support, this work could not have come into being.

Ron Sun
Troy, New York

Anatomy of the Mind

1

What Is A Cognitive Architecture?

In this chapter, as an introduction to what is to be detailed in this book, I will attempt to justify the endeavor of developing a generic computational model (theory) of the mind (i.e., a computational cognitive architecture), through addressing a series of questions. Then I will discuss a few issues fundamental to such an endeavor.

1.1. A Theory of the Mind and Beyond

Before embarking on this journey, it might help to make clear at the outset that what is to be described and discussed in the present book, including concepts, theories, models, and simulations, is centered on a particular theoretical framework—namely, the Clarion framework. It is worth noting that Clarion, in its full-fledged form, is a generic and relatively comprehensive theory of the human mind,[1] along with a computational implementation of the theory. It is thus a computational "cognitive

1. "Mind" is a complex notion. Rather than engaging in a philosophical discourse on the notion, the focus here is instead on mechanisms and processes of the mind. In turn, "mechanism" here refers to physical entities and structures and their properties that give rise to certain characteristics of the mind. Although living things often appear to have certain characteristics that have no counterpart in the physical universe, one may aim to go beyond these appearances (Thagard, 1996).

architecture" as is commonly referred to in cognitive science, cognitive psychology, or more generally in the "cognitive sciences".[2] In general, a cognitive architecture is a broad domain-generic cognitive-psychological model implemented computationally.

Clarion has been in continuous development for a long time, at least since 1994 (although its predecessors have had a longer history). It has been aimed to capture, explain, and simulate a wide variety of cognitive-psychological phenomena within its unified framework, thus leading (hopefully and ultimately) to unified explanations of psychological (and even other related) phenomena (as advocated by, e.g., Newell, 1990). The exact extent of cognitive-psychological phenomena that have been captured and explained within its framework will be discussed in detail in subsequent chapters. It is not unreasonable to say that Clarion constitutes a (relatively) comprehensive theory of the mind (or at least an initial version of such a theory).

In fact, Clarion, within itself, contains several different kinds of theories. First, it contains a core theory of the mind at a conceptual level. It posits essential theoretical distinctions such as implicit versus explicit processes, action-centered versus non-action-centered processes, and so on, as well as their relationships (Sun, 2002, 2012). With these distinctions and other high-level constructs, it specifies a core theory of the essential structures, mechanisms, and processes of the mind, at an abstract, conceptual level (Sun, Coward, and Zenzen, 2005).

Second, it also contains a more detailed (but still generic) computational model implementing the abstract theory. This implementation constitutes what is usually referred to as a computational cognitive architecture: that is, a generic computational cognitive (i.e., psychological) model describing the architecture of the mind, which by itself also constitutes a theory of the mind, albeit at a more detailed and computational level (as will be argued later; see also Sun, 2009b).

2. In the narrow sense, "cognition" refers to memory, learning, concepts, decision making, and so on—those aspects of the individual mind that are not directly related to motivation, emotion, and the like. In the broadest sense, it may refer to all aspects of the individual mind, especially when methods and perspectives from contemporary cognitive science are used in studying these aspects. In the latter case, I often use a hyphenated form, "cognition-psychology", to make it clear. However, the plural form, "cognitive sciences," is often used to refer to all fields of cognitive, behavioral, and psychological sciences, applying the broadest sense of the term. Similarly, in the term "cognitive architecture," the word "cognitive" should be interpreted in the broadest sense.

Third, with the generic computational cognitive architecture, one may construct specific models and simulations of specific psychological phenomena or processes. That is, one may "derive" specific computational models (namely, specific computational theories) for specific psychological phenomena or processes, from the generic computational model (theory). So, the generic theory leads to specific theories.

Clarion encompasses all of the above simultaneously. Thus, it synthesizes different types of theories at different levels of theoretical abstraction (Sun, 2009b). Below I will refer, alternately, to Clarion in these different senses, at different levels of abstraction, as appropriate.

1.2. Why Computational Models/Theories?

Why would one want computational models for the sake of understanding the human mind? Why are computational models useful exactly?

Generally speaking, models of various forms and complexities may be roughly categorized into computational, mathematical, and verbal-conceptual varieties (Sun, 2008). Computational models present algorithmic descriptions of phenomena, often in terms of mechanistic and process details. Mathematical models present (often abstract) relationships between variables using mathematical equations. Verbal-conceptual models describe entities, relations, or processes in informal natural languages (such as English). A model, regardless of its genre, might often be viewed as a theory of whatever phenomena that it purports to capture. This point has been argued extensively before (by, e.g., Newell, 1990 and Sun, 2009b).

Although each of these types of models has its role to play, I am mainly interested in computational modeling. The reason for this preference is that, at least at present, computational modeling appears more promising in many respects. It offers the expressive power that no other approach can match, because it provides a wider variety of modeling techniques and methodologies. In this regard, note that mathematical models may be viewed as a subset of computational models, because normally they can lead readily to computational implementations (even though some of them may be sketchy, not covering sufficient mechanistic or process details). Computational modeling also supports practical applications (see, e.g., Pew and Mavor, 1998; Sun, 2008).

Computational models are mostly mechanistic and process oriented. That is, they are mostly aimed at answering the questions of how human performance comes about, by what psychological structures, mechanisms, and processes, and in what ways.[3] The key to understanding cognitive-psychological phenomena is often in fine details, which computational modeling can describe and illuminate (Newell, 1990; Sun, 2009b). Computational models provide algorithmic specificity: detailed, exactly specified, and carefully worked-out steps, arranged in precise and yet flexible sequences. Thus, they provide clarity and precision (see, e.g., Sun, 2008).

Computational modeling enables and, in fact, often forces one to think in terms of mechanistic and process details. Instead of verbal-conceptual theories, which may often be vague, one has to think clearly, algorithmically, and in detail when dealing with computational models/theories. Computational models are therefore useful tools. With such tools, researchers must specify a psychological mechanism or process in sufficient detail to allow the resulting models to be implemented on computers and run as simulations. This requires that all elements of a model (e.g., all its entities, relationships, and so on) be specified exactly. Thus it leads to clearer, more consistent, more mechanistic, more process-oriented theories. Richard Feynman once put it this way: "What I cannot create, I do not understand." This applies to the study of human cognition-psychology. To understand is to create, in this case on a computer at least.

Computational models may be necessary for understanding a system as complex and as internally diverse as the human mind. Pure mathematics, developed mainly for describing the physical universe, may not be sufficient for understanding a system as different as the human mind. Compared with theories developed in other disciplines (such as physics), computational modeling of the mind may be mathematically less "elegant", but the human mind itself may be inherently less mathematically elegant when compared with the physical universe (as argued by, e.g., Minsky, 1985). Therefore, an alternative form of theorizing may be necessary—a form that is more complex, more diverse, and more algorithmic in nature. Computational modeling provides a viable way

3. It is also possible to formulate so called "product theories", which provide a functional account of phenomena but do not commit to a particular psychological mechanism or process. Thus, product theories can be evaluated mainly by product measures. One may also term product theories *black-box theories* or *input-output theories*.

of specifying complex and detailed theories of cognition-psychology. Therefore, they may be able to provide unique explanations and insights that other experimental or theoretical approaches cannot easily provide.

A description or an explanation in terms of computation that is performed in the mind/brain can serve either as a fine-grained specification of cognitive-psychological processes underlying behavior (roughly, the mind), or as an abstraction of neurobiological and neurophysiological data and discoveries (roughly, the brain), among other possibilities that may also exist. In general, it is not difficult to appreciate the usefulness of a computational model in this regard, in either sense, especially one that summarizes a body of data, which has been much needed in psychology and in neuroscience given the rapid growth of empirical data.

In particular, understanding the mind (at the psychological level) through computational modeling may be very important. One would naturally like to know more about both the mind and the brain. So far at least, we still know little about the biology and physiology of the brain, relatively speaking. So for this reason (and others), we need a higher level of abstraction; that is, we need to study the mind at the psychological level in order to make progress toward the ultimate goal of fully understanding the mind and the brain.

Trying to fully understand the human mind purely from observations of human behavior (e.g., strictly through behavioral experiments) is likely untenable (except perhaps for small, limited task domains). The rise and fall of behaviorism is a case in point. This point may also be argued on the basis of analogy with the physical sciences (as was done in Sun, Coward, and Zenzen, 2005). The processes and mechanisms of the mind cannot be understood purely on the basis of behavioral experiments, which often amount to tests that probe relatively superficial features of human behavior, further obscured by individual and cultural differences and other contextual factors. It would be extremely hard to understand the human mind in this way, just like it would be extremely hard to understand a complex computer system purely on the basis of testing its behavior, if one does not have any prior ideas about the inner workings and theoretical underpinnings of that system (Sun, 2007, 2008, 2009b).

Experimental neuroscience alone may not be sufficient either, at least for the time being. Although much data has been gathered from empirical work in neuroscience, there is still a long way to go before all the details of the brain are identified, let alone the psychological functioning on that basis. Therefore, as argued earlier, at least at present, it is important to

understand the mind/brain at a higher level of abstraction. Moreover, even when we finally get to know all the minute details of the brain, we would still need a higher-level, yet precise, mechanistic, process-based understanding of its functioning. Therefore, we still need a higher level of theorizing. In an analogous way, the advent of quantum mechanics did not eliminate the need for classical mechanics. The progress of chemistry was helped by the discoveries in physics, but chemistry was not replaced by physics. It is imperative that we also investigate the mind at a higher level of abstraction, beyond neuroscience. Computational modeling has its unique, indispensable, and long-term role to play, especially for gaining conceptually clear, detailed, and principled understanding of the mind/brain.

It might be worth mentioning that there have been various viewpoints concerning the theoretical status of computational modeling. For example, many believed that a computational model (and computational simulation on its basis) may serve as a generator of phenomena and data. That is, they are useful media for hypothesis generation. In particular, one may use simulation to explore process details of a psychological phenomenon. Thus, a model is useful for developing theories, constituting a theory-building tool. A related view is that computational modeling and simulation are suitable for facilitating a precise instantiation of a preexisting verbal-conceptual theory (e.g., through exploring possible details for instantiating the theory) and consequently detailed evaluations of the theory against data. These views, however, are not incompatible with a more radical position (e.g., Newell 1990; Sun 2009b) that a computational model may constitute a theory by itself. It is not the case that a model is limited to being built on top of an existing theory, being applied for the sake of generating data, being applied for the sake of validating an existing theory, or being used for the sake of building a future theory. According to this more radical view, a model may be viewed as a theory by itself. In turn, algorithmic descriptions of computational models may be considered just another language for specifying theories (Sun, 2009b; Sun, 2008).[4] The reader is referred to Sun (2009b) for a more in-depth discussion of this position.

4. Constructive empiricism (van Fraasen, 1980) may serve as a philosophical foundation for computational cognitive modeling, compatible with the view of computational models as theories (Sun 2009b).

In summary, computational models (theories) can be highly useful to psychology and cognitive science, when viewed in the light above (and when the issues discussed below are properly addressed).

1.3. Questions about Computational Models/Theories

There are, of course, many questions that one can, and should, ask about any computational model before "adopting" it in any way.

One important question about any particular computational model is this: how much light can it really shed on the phenomena being modeled? There are a number of aspects to this question, for instance:

- Do the explanations provided by the computational model capture accurately human "performance" (in a Chomskian sense; Chomsky, 1980)? That is, does it capture and explain sufficiently the subtleties exhibited in the empirical data? If an explanation is devoid of "performance" details as observed in empirical data, it will be hard to justify the appropriateness of such an explanation, especially when there are other possible ways of describing the data.[5]
- Does the model take into consideration higher-level or lower-level constraints (above or below the level of the model in question)? There are usually many possible models/theories regarding some limited data. Higher-level or lower-level considerations, among other things, may be used to narrow down the choices.
- Does the model capture in a detailed way psychological mechanisms and processes underlying the data? If a model lacks mechanistic, process-oriented details, it may be less likely to bring new insights into explaining the dynamics underlying the data.
- Do the primitives (entities, structures, and operations) used in the model provide some descriptive power and other advantages over and above other possible ways of describing human behavior and performance (but without being overly generic)?

5. This is not the case for Noam Chomsky's theory of language, which thus serves as an exception.

- Does the model provide a basis for tackling a wide set of cognitive-psychological tasks and data? If a model is insufficient in terms of breadth of coverage, it cannot claim to be a "general" theory.

It should be noted that, in relation to the issue of generality, one should be aware of the danger of over-generality. That is, a model might be so under-constrained that it may match practically any possible data, realistic or unrealistic. To address this problem, many simulations in a wide range of domains are needed, in order to narrow down choices and to constrain parameter spaces (more on this in Chapter 8).

From the point of view of the traditional cognitive science, a model/theory at the computational or knowledge level (in Marr's [1982] or Newell's [1990] sense) can provide a formal language for describing a range of cognitive-psychological tasks. Indeed, in the history of cognitive science, some high-level formal theories were highly relevant (e.g., Chomsky's theory of syntactic structures of language). So, a further question is:

- How appropriate is the model/theory in terms of providing a "formal language" for a broad class of tasks or data? Does it have realistic expressiveness (sufficient for the target tasks or data, but not much more or less) and realistic constraints (of various types, at various levels)?

Furthermore, what is more important than a formal (e.g., mathematical or computational) language for describing cognition-psychology is the understanding of the "architecture" of the mind, especially in a mechanistic (computational) sense. That is, one needs to address the following question:

- How do different components of the mind interact and how do they fit together? Correspondingly, how do different components of a computational model/theory interact and how do these different components fit together, instead of just a mere collection of limited models?

Studying architectural issues may help us to gain new insight, narrow down possibilities, and constrain the components involved.

Moreover, different components and different functionalities of the mind, for example, perception, categorization, concepts, memory,

decision making, reasoning, problem solving, planning, communication, action, learning, metacognition, and motivation, all interact with and depend on each other. Their patterns of interaction change with changing task demands, growing personal experiences, varying sociocultural contexts and milieus, and so on. Some argue that cognition-psychology represents a context-sensitive, dynamic, statistical structure that, on the surface at least, changes constantly—a structure in perpetual motion. However, complex dynamic systems may be attributed to its constituting elements. Thus, one may strive for a model that captures the dynamics of cognition-psychology through capturing its constituting elements and their interaction and dependency. So, an important question is:

- How does a model/theory account for the dynamic nature of cognition-psychology?

Finally, one has to consider the cost and benefit of computational modeling:

- Is the complexity of a model/theory justified by its explanatory utility (considering all the questions above)?

These questions cannot be addressed in abstraction. My specific answers to them, in the context of Clarion, will emerge in subsequent chapters, as details of Clarion emerge in these chapters.

1.4. Why a Computational Cognitive Architecture?

Among different types of computational cognitive-psychological models/theories, computational cognitive architectures stand out. A computational cognitive architecture, as commonly termed in cognitive science, is a broadly scoped, domain-generic cognitive-psychological model, implemented computationally, capturing the essential structures, mechanisms, and processes of the mind, to be used for broad, multiple-level, multiple-domain analysis of behavior (e.g., through its instantiation into more detailed computational models or as a general framework; Newell, 1990; Sun, 2007).

Let us explore this notion of cognitive architecture with a comparison. The architecture for a building consists of its overall structural design and major constituting structural elements such as external walls, floors, roofs,

stairwells, elevator shafts, and so on. Furniture can be easily rearranged or replaced and therefore may not be part of the architecture. By the same token, a cognitive architecture includes overall structures, essential divisions of modules (e.g., subsystems), essential relations between modules, basic representations and algorithms within modules, and a variety of other major aspects (Sun, 2007; Langley, Laird & Rogers, 2009). In general, a cognitive architecture includes those aspects that are relatively invariant across time, domains, and individuals. It deals with them in a structurally and mechanistically well-defined way.

A cognitive architecture can be important to understanding the human mind. It provides concrete computational scaffolding for more detailed modeling and exploration of cognitive-psychological phenomena and data, through specifying essential computational structures, mechanisms, and processes. That is, it facilitates more detailed modeling and exploration of the mind. As discussed earlier, computational cognitive modeling explores cognition-psychology through specifying computational models of cognitive-psychological mechanisms and processes. It embodies descriptions of cognition-psychology in computer algorithms and program codes, thereby producing runnable models. Detailed simulations can then be conducted. In this undertaking, a cognitive architecture can be used as the unifying basis for a wide range of modeling and simulation. Note that here I am mainly referring to psychologically oriented cognitive architectures (as opposed to software engineering oriented cognitive architectures, which are quite different in terms of purpose).

A cognitive architecture serves as an initial set of (relatively) generic assumptions that may be applied in further modeling and simulation. These assumptions, in reality, may be based on empirical data, philosophical arguments, or computational considerations. A cognitive architecture is useful and important because it provides a (relatively) comprehensive yet precise foundation that facilitates further modeling in a wide variety of domains (Cooper, 2007).

In exploring cognitive-psychological phenomena, the use of cognitive architectures forces one to think in terms of mechanistic and process-oriented details. Instead of using often vague and underspecified verbal-conceptual theories, cognitive architectures force one to think more clearly. Anyone who uses cognitive architectures must specify a cognitive-psychological mechanism or process in sufficient detail to allow the resulting models to run as simulations. This approach encourages more detailed and clearer theories. It is true that more specialized,

narrowly scoped computational models may also serve this purpose, but they are not as generic and as comprehensive. Consequently, they are not as generally useful. Cognitive architectures are thus crucial tools (Pew and Mavor, 1998; Sun, 2007).

A cognitive architecture may also provide a deeper level of explanation (Sun, 2007). Instead of a model specifically designed for a specific task (which is often ad hoc), a cognitive architecture naturally encourages one to think in terms of the mechanisms and processes available within a generic model that are not specifically designed for a particular task, and thereby to generate explanations of the task that are not centered on superficial, high-level features of the task (as often happens with specialized, narrowly scoped models)—that is, to generate explanations of a deeper kind. To describe a task in terms of available mechanisms and processes of a cognitive architecture is to generate explanations based on primitives of cognition-psychology envisioned in the cognitive architecture, thereby leading to deeper explanations.

Because of the nature of such deeper explanations, this approach is also more likely to lead to unified explanations for a wide variety of data and phenomena, because potentially a wide variety of tasks, data, and phenomena can be explained on the basis of the same set of primitives provided by the same cognitive architecture (Sun, 2007). Therefore, a cognitive architecture is more likely to lead to a unified, comprehensive theory of the mind, unlike using more specialized, narrowly scoped models (Newell, 1990).

Although the importance of being able to reproduce the nuances of empirical data is evident, broad functionalities in cognitive architectures are even more important (Newell, 1990). The human mind needs to deal with all of the necessary functionalities: perception, categorization, memory, decision making, reasoning, planning, problem solving, communication, action, learning, metacognition, motivation, and so on. The need to emphasize generic models capable of broad functionalities arises also because of the need to avoid the myopia often resulting from narrowly scoped research.

For all of these reasons above, developing cognitive architectures is an important endeavor in cognitive science. It is of essential importance in advancing the understanding of the human mind (Sun, 2002, 2004, 2007). Existing cognitive architectures that have served this purpose include ACT-R, Soar, Clarion, and a number of others (see, e.g., Taatgen and Anderson, 2008 for a review).

In addition, cognitive architectures also, in a way, support the goal of general AI, that is, building artificial systems that are as capable as human beings. In relation to building intelligent systems, a cognitive architecture may provide the underlying infrastructure, because it may include a variety of capabilities, modules, and subsystems that an intelligent system needs. On that basis, application systems may be more readily developed. A cognitive architecture carries with it theories of cognition-psychology and understanding of intelligence gained from studying the human mind. In a way, cognitive architectures reverse engineer the only truly intelligent system around—the human mind. Therefore, the development of intelligent systems on that basis may be more cognitively-psychologically grounded, which may be advantageous in some circumstances. The use of cognitive architectures in building intelligent systems may also facilitate the interaction between humans and artificially intelligent systems because of the relative similarity between humans and cognitively-psychologically based intelligent systems. It was predicted a long time ago that "in not too many years, human brains and computing machines will be coupled together very tightly and the resulting partnership will think as no human brain has ever thought . . ." (Licklider, 1960). Before that happens, a better understanding of the human mind is needed, especially a better understanding in a computational form.

There are, of course, questions that one should ask about cognitive architectures, in addition to or instantiating questions about computational modeling in general as discussed earlier. For instance, a cognitive architecture is supposed to include all essential psychological capabilities and functionalities. As mentioned before, those functionalities may include perception, categorization, memory, decision making, reasoning, problem solving, communication, action, and learning. They may involve all kinds of representation (in a broad sense). There are also motivational and metacognitive processes. However, currently, most cognitive architectures do not yet support all of these functionalities, at least not fully. So, what is minimally necessary? How should these functionalities interact? To what extent are they separate? And so on. There are no simple answers to these questions, but they will be addressed along the way in this book.

In this regard, a question concerning any capability in a cognitive architecture is whether the cognitive architecture includes that capability as an integral part or whether it includes sufficient basic functionalities that allow the capability to emerge or to be implemented later on.

This may be determined by what one views as an integral part of a cognitive architecture and what one views as a secondary or derived capability. Sun (2004) provides a discussion of the relation between a cognitive architecture and the innate structures in the human mind and the notion of minimality in a cognitive architecture. These ideas may help to sort out what should or needs to be included in a cognitive architecture (Sun, 2004). The outcomes of the deliberation on this and other questions will be presented in the subsequent chapters.

1.5. Why Clarion?

Among existing cognitive architectures, why should one choose Clarion? In a nutshell, one might prefer Clarion, for (the totality of) the following reasons:

- Clarion is a cognitive architecture that is more comprehensive in scope than most other cognitive architectures in existence today (as will become clear later).
- Clarion is psychologically realistic to the extent that it has been validated through simulating and explaining a wide variety of psychological tasks, data, and phenomena (as detailed in chapters 5, 6, and 7).
- Its basic principles and assumptions have been extensively argued for and justified, in relation to a variety of different types of evidence (as detailed in chapters 2, 3, and 4).
- It has major theoretical implications, as well as some practical relevance. It has provided useful explanations for a variety of empirical data, leading to a number of significant new theories regarding psychological phenomena (e.g., Sun, Slusarz, & Terry, 2005; Helie & Sun, 2010).
- In addition to addressing problems at the psychological level, it has also taken into account higher levels, for example, regarding social processes and phenomena, as well as lower levels (Sun, Coward, & Zenzen, 2005).

More specifically, Clarion has been successful in computationally modeling, simulating, accounting for, and explaining a wide variety of psychological data and phenomena. For instance, a number of well-known

skill-learning tasks have been simulated using Clarion that span the entire spectrum ranging from simple reactive skills to complex cognitive skills. The simulated tasks, for example, include serial reaction time tasks, artificial grammar learning tasks, dynamic process control tasks, alphabetical arithmetic tasks, and Tower of Hanoi (e.g., Sun, Slusarz, & Terry, 2005; Sun, 2002). In addition, extensive work has been done in modeling complex and realistic skill-learning tasks that involve complex sequential decision making (Sun et al., 2001). Furthermore, many other kinds of tasks not usually dealt with by cognitive architectures—reasoning tasks, social simulation tasks, as well as metacognitive and motivational tasks—have been tackled by Clarion. While accounting for various psychological tasks, data, and phenomena, Clarion provides explanations that shed new light on underlying cognitive-psychological processes. See, for example, Sun et al. (2001), Sun, Slusarz, and Terry (2005), Sun, Zhang, and Mathews (2006), and Helie and Sun (2010) for various examples of such simulations and explanations.

These simulations, more importantly, provided insight that led to some major new theories concerning a number of important psychological functionalities. Some new theories resulting from Clarion include:

- The theory of bottom-up learning (from implicit to explicit learning), as developed in Sun et al. (2001).
- The theory of the implicit-explicit interaction and their synergistic effects on skill learning, as developed in Sun, Slusarz, and Terry (2005).
- The theory of creative problem solving, as described in Helie and Sun (2010).
- The theory of human motivation and its interaction with cognition, as described in Sun (2009), as well as in related simulation papers (e.g., Wilson, Sun, & Mathews, 2009; Sun & Wilson, 2011; Sun & Wilson, 2014)
- The theory of human reasoning (based on implicit and explicit representation and their interaction), as developed in Sun (1994, 1995) and Sun and Zhang (2006).

These theories are standalone, conceptual-level theories of psychological phenomena. However, these theories are also an integral part of Clarion. They have been computationally instantiated. They have led not only to

numerical (quantitative) simulations but also to major qualitative (theoretical) predictions.

I should mention here that two meta-principles have guided the development of this cognitive architecture: (a) completeness of functionalities (to include as many functionalities as possible), but (b) parsimony of mechanisms (to reduce the number of distinct mechanisms and their complexity as much as possible). Or to put it another way, the goal for Clarion has been: maximum scope and minimum mechanism. That goal and the associated meta-principles have led to the aforementioned theories and explanations by Clarion.

Given all of the above, Clarion is worthy of further exploration and examination. In particular, its comprehensive scope should be examined in more detail. Thus a book-length treatment is required.

1.6. Why This Book?

Although a substantial number of articles, including journal and conference papers, have been published on Clarion and its modeling of psychological data of various kinds, there is currently no one single volume that contains all of the information, especially not in a unified and accessible form. Therefore, it seems a good idea to put together a single volume for the purpose of cataloguing and explaining in a unified and accessible way what has been done with regard to Clarion and why it might be of interest.

Furthermore, a book may contain much more material than a typical journal or conference paper. It may describe not only details of Clarion but also many detailed models of psychological phenomena based on Clarion. It may summarize materials published previously, in addition to new materials. A book may also provide theoretical and meta-theoretical discussions of issues involved. Above all, a book may provide a gentler introduction to Clarion and its exploration of psychological mechanisms and processes, which may be of use to some readers.

The present book will present a unified (albeit preliminary and still incomplete) view of the human mind, and interpret and explain empirical data on the basis of that view. The focus will be on broad interpretations of empirical data and phenomena, emphasizing unified explanations of a wide variety of psychological tasks and data. Thus

exact parameter values and other minute technical details will be minimized.

For the sake of clarity, I will proceed in a hierarchical fashion. In other words, there will be a series of progressively more detailed descriptions. First, a high-level conceptual sketch will be given; then a more detailed description will be provided. After that, there will be an even more detailed, more technical description. (However, the most technically exact and complete description, with a code library, can be found in a forthcoming companion technical book on Clarion.) In this way, the reader may stop at any time, up to the level where he or she feels comfortable.

I will start with the overall Clarion framework and then move on to individual components or aspects. To achieve clarity, I will limit the amount of details discussed to only those that are minimally necessary. (Fortunately, the technical book will provide full technical specifications.) With regard to technical details, especially in relation to simulations, I will have to strike a balance between conceptual clarity and technical specificity. Of course, both are important. To achieve conceptual clarity, a high-level conceptual explanation will be provided. To achieve some technical specificity, a more technical (computational) description or explanation will also be provided, corresponding to the high-level conceptual explanation.

1.7. A Few Fundamental Issues

To start, I will quickly sketch a few foundational issues. My stands on these issues form the meta-theoretical basis of Clarion. (Details of the cognitive architecture will be explained in subsequent chapters.)

1.7.1. Ecological-Functional Perspective

The development of a cognitive architecture needs to take into consideration of what I have called the ecological-functional perspective. As discussed in Sun (2012) and Sun (2002), the ecological-functional perspective includes a number of important considerations on human cognition-psychology, especially in relation to ecological realism of

cognitive-psychological theories or models. They may be expressed as dictums such as:

- taking into account ecological niches (evolutionarily or at the present), and focusing attention on characteristics of everyday activities that are most representative of the ecological niches (Sun, 2002; more later);
- taking into account the role of function, because cognitive-psychological characteristics are often, if not always, functional, useful in some way for everyday activities within an ecological niche;
- taking into account cost-benefit trade-offs of cognitive-psychological characteristics (such as implicit versus explicit processes)[6], as psychological characteristics are often selected based on cost-benefit considerations (evolutionarily or at the present).

In particular, these dictums imply that human cognition-psychology is mostly activity-based, action-oriented, and embedded in the world. They also seem to point toward implicit (subconscious or unconscious) psychological processes, as opposed to focusing exclusively on explicit processes. Humans often interact with the world in a rather direct and unmediated way (Heidegger, 1927; Dreyfus, 1992; Sun, 2002).

These dictums, serving as meta-heuristics for developing cognitive architectures, will become clearer in the next chapter, when the justifications for the essential framework of Clarion are discussed.

1.7.2. Modularity

Fodor (1983) argued that the brain/mind was modular and its modules worked largely independently and communicated only in a limited way. However, evidence to the contrary has accumulated that modules and subsystems in the brain/mind may instead be more richly interconnected, anatomically and functionally (Damasio, 1994; Bechtel, 2003).

Nevertheless, starting off with a modular organization might make the task of understanding the architecture of the human mind more tractable.

6. For instance, compared with implicit processes, explicit processes may be more precise but may be more effortful. See more discussions in Chapter 3.

Connections, communications, and interactions, if necessary, may be added subsequently. At a minimum, some cognitive-psychological functionalities do appear to be specialized and somewhat separate from others (in a sense). They may be so either because they are functionally encapsulated (their knowledge, mechanisms, and processes do not transfer easily into other domains) or because they are physically (neurophysiologically) encapsulated. Modularity can be useful functionally, for example, to guarantee efficiency or accuracy of important or critical behaviors and routines (whether they are a priori or learned), or to facilitate parallel operations of multiple processes (Sun, 2004). Hence we start with a (circumscribed) modular view.

1.7.3. Multiplicity of Representation

With modularity (i.e., with the co-existence of multiple modules), multiple different representations (either in terms of form or in terms of content) may co-exist.

Here I use the term "representation" to denote any form of internal encoding, either explicitly and individually encoded or embodied/enmeshed within a complex mechanism or process. Thus this notion of "representation" is not limited to explicit, individuated symbolic entities and their structures (as often meant by "representationalism"). Because it is not limited to symbolic forms, it includes, for example, connectionist encoding, dynamic system content, and so on. So the term should be interpreted broadly here.

In terms of representational form, there are, for example, symbolic-localist representation and distributed connectionist representation. Symbolic-localist representation implies representing each unique concept by a unique basic representational entity (such as a node in a network). Distributed representation involves representing each concept by an activation pattern over a shared set of nodes in a network (Rumelhart et al., 1986). Different forms of representations have different computational characteristics: for example, crisp versus graded, rule-based versus similarity-based, one-shot learning versus incremental learning, and so on, as will be discussed in more detail later.

In terms of representational content, there may be the following types: procedural representation, declarative representation, metacognitive representation, motivational representation, and so on. Each of these types

is necessary for a full account of the human mind. In subsequent chapters when I discuss each of these types in turn, I will present arguments why each of them is needed. Each type may in turn involve multiple representational forms within.

On the other hand, one may question why an individual needs multiple representational forms after all. There are a number of potential advantages that may be gained by involving multiple representational forms. For example, in incorporating both symbolic-localist and distributed representation (for capturing explicit and implicit knowledge, respectively, as will be detailed later), one may gain

- synergy in skill learning from dual procedural representation
- synergy in skill performance from dual procedural representation
- synergy in reasoning from dual declarative representation

and so on. These advantages have been demonstrated before in previous publications; I will elaborate on these advantages in later chapters when I revisit these points.

1.7.4. Dynamic Interaction

In a cognitive architecture, various modules (in the previously discussed sense) have to work with each other to accomplish psychological functioning. Modules of different kinds and sizes (e.g., subsystems and components within each subsystem) interact with each other dynamically.

At the highest level, the interaction among subsystems may include metacognitive monitoring and regulation of other processes (i.e., the interaction between the metacognitive subsystem and the other subsystems). The interaction among subsystems may also involve motivated action decision making (i.e., the interaction between the motivational subsystem and the action-centered subsystem). Within each subsystem, many component modules exist and they also interact with each other, necessary for accomplishing cognitive-psychological functioning.

Note that these characteristics may not have been sufficiently captured by most existing cognitive-psychological models (including cognitive architectures). Compared with these other models, Clarion is unique in terms of containing (well-developed, built-in) motivational constructs and (well-developed, built-in) metacognitive

constructs. These are not commonly found in existing cognitive architectures. Nevertheless, I believe that these features are crucial to a cognitive architecture because they capture important or indispensable elements of the human mind, necessary in the interaction between an individual and his or her physical and social world (Sun, 2009). Details will be presented in subsequent chapters.

1.8. Concluding Remarks

So far I have covered only some preliminary ideas, which are necessary background regarding cognitive architectures. The questions that have been addressed include: Why should one use computational modeling for studying cognition-psychology? Why should one use cognitive architectures among other computational models? Why should one use the Clarion cognitive architecture, among other possible cognitive architectures? And other questions and issues.

More importantly, the basic "philosophy" in regard to a number of fundamental issues has been outlined. In particular, the principles of modularity, multiplicity of representation, and dynamic interaction (include that among motivation, cognition, and metacognition) are of fundamental importance to Clarion.

The rest of the book is divided into eight chapters. They include three chapters for presenting various theoretical, conceptual, and technical aspects of Clarion, three chapters on various simulations using Clarion, and additional materials in the remaining two chapters.

Finally, a note for the interested reader: for general surveys, discussions, and comparisons of computational cognitive architectures in the context of cognitive-psychological modeling, covering other well-known cognitive architectures (such as ACT-R and Soar), see Pew and Mavor (1998), Ritter et al. (2003), Sun (2006), Chong, Tan, and Ng (2007), Taatgen and Anderson (2008), Langley et al. (2009), Thórisson and Helgasson (2012), Helie and Sun (2014b), among other existing publications (see also Chapter 9).

2

Essential Structures of the Mind

In this chapter, I introduce the basic framework (i.e., the relatively abstract conceptual-level theory) of Clarion, and discuss the justifications for this framework.

In a way, this chapter presents a worldview—an essential, overarching framework for understanding the mind. One should view it as the more abstract general theory of Clarion, as opposed to the more detailed computational theory of Clarion, which will be presented in chapters 3 and 4, or as opposed to the specific computational simulation models derived from Clarion, which will be presented in chapters 5, 6, and 7.

Below I will first review and justify the essential desiderata that have been driving the development of Clarion. Then, on that basis, the overall structure of Clarion will be sketched.

2.1. Essential Desiderata

As has been characterized earlier, Clarion is a computational cognitive architecture: it is a generic and comprehensive model of cognitive-psychological structures, mechanisms, processes, functionalities,

and so on, specified and implemented computationally. As such, it needs substantial justifications.

Clarion has indeed been justified extensively on the basis of empirical data (see, e.g., Sun, 2002, 2003; see also Sun, Merrill, & Peterson, 2001; Sun, Slusarz, & Terry, 2005; Helie & Sun, 2010), as well as theoretical (fundamental, philosophical) considerations. In particular, a number of essential (philosophical and psychological) desiderata have been central to the conception of the framework. These essential desiderata include those described below (along with others described elsewhere, e.g., in Sun, 2002, 2004, 2012). Together, they present a situated/embodied view of the mind in a generalized sense (Sun, 2013b), consistent with the ecological-functional perspective discussed in Chapter 1, in addition to the other considerations outlined there (e.g., representational multiplicity, modularity, and dynamic interaction).

Sequentiality. Everyday activities are sequential: they are often carried out one step at a time. Temporal structures are essential to such activities and form the basis of behaviors in different circumstances (Sun, 2002).

Routineness. Everyday activities are largely made up of reactive routines (skills), or habitual sequences of behavioral responses (on a moment-to-moment basis mostly). They are, generally speaking, gradually formed and subject to continuous modification (with the possible exception of some innate routines or instincts). Therefore, human everyday activities may be viewed as comprised of forming, adapting, and following routines (or skills; Sun, 2002; Tinbergen, 1951; Timberlake and Lucas, 1989).

Trial-and-error adaptation. Learning of reactive routines (and other behaviors) is often a trial-and-error process. Such learning has been variously studied under the rubric of law of effect, classical conditioning, instrumental conditioning, probability learning, and implicit learning (Reber, 1989). Such learning is essential to human everyday activities (Sun, 2002).

Implicit versus explicit processes. Reactive routines are mostly implicit. Implicit processes are (relatively) inaccessible and "holistic," while explicit processes are more accessible and more precise (e.g., Reber, 1989). These two types interact with each other. This dichotomy is related to some other well-known dichotomies: the conscious versus the unconscious, the conceptual versus the subconceptual, and so on (Evans & Frankish, 2009; Sun, 2002).

Synergistic interaction. It was hypothesized that one reason for having these two types of processes, implicit and explicit, was that these processes worked together synergistically, supplementing and complementing each other in a variety of ways (Sun, Slusarz, & Terry, 2005). These two types have qualitatively different characteristics, thus often generating better overall results when they interact (Sun, 2002).

Bottom-up and top-down learning. The interaction between implicit and explicit processes allows for a gradual transfer of knowledge (memory) from one type to the other (besides separate, standalone learning within each type). Learning resulting from the implicit-explicit interaction includes top-down learning (explicit learning first and implicit learning on that basis) and bottom-up learning (implicit learning first and explicit learning on that basis; Sun, 2002).

Procedural versus declarative processes. Procedural processes are specifically concerned with actions in various circumstances (i.e., how to do things). Declarative processes are not specifically concerned with actions but are more about objects, persons, events, and so on, in generic terms. This distinction has provided useful insight in interpreting a wide range of psychological data in the past (Proctor & Dutta, 1995). Furthermore, the procedural-declarative distinction is orthogonal to the implicit-explicit distinction (based on empirical evidence as summarized in Sun, 2012).

Motivational control. A full account of behavior must address why one does what one does.[1] Hence motivational processes need to be understood (Sun, 2009). An individual's essential motivations (needs) arise prior to deliberative cognition (Sun, 2009) and are the foundation of cognition and action. In a way, cognition has evolved to serve the essential needs (motives) of an individual, and bridges the needs (motives) of an individual and his or her environments.

Metacognitive control. Metacognition regulates cognition. For need fulfillment, metacognitive monitoring and regulation are necessary. They help to set goals, to assess progresses, and to adopt or change various parameters and strategies (large or small) for goal achievement. The importance of metacognition has been well established (see, e.g., Reder, 1996, and Sun & Mathews, 2012).

1. Simply saying that one chooses actions to maximize rewards or reinforcement is not sufficient. It leaves open the question of what determines them.

Table 2.1. Fundamental issues relevant to Clarion (see Chapter 1 for details).

Ecological-functional perspective
Modularity
Multiplicity of representation
Dynamic interaction

Table 2.2. Some essential desiderata for Clarion (see text for details).

Sequentiality
Routineness
Trial-and-error adaptation
Implicit versus explicit processes
Synergistic interaction
Bottom-up and top-down learning
Procedural versus declarative processes
Motivational and metacognitive control

For justifying these desiderata (see tables 2.1 and 2.2), more supporting arguments and evidence are needed. But before that, an example that illustrates how these desiderata might be tied together is in order. The example, in a way, also justifies the desiderata above.

2.2. An Illustration of the Desiderata

According to the framework of Clarion, when an individual is born into the world, that is, when an agent is instantiated into the system, little information, skill, or knowledge is readily available. For instance, the individual comes with no explicit knowledge, either about the self or about the world. But the individual does come with evolutionarily hard-wired instincts (e.g., reflexes). Moreover, the individual has needs, such as hunger and thirst, which constitute innate motives driving actions and reactions. The individual certainly has no explicit knowledge of how to meet these needs but does have hard-wired instinctual responses, including primitive behavioral routines, which may be applied in attempts to satisfy the needs.

The individual is endowed with sensory inputs regarding environmental states and internal states. Whenever there is a growing physiological deficit, an internal change may lead to heightened activation of a motive (need). It may therefore lead to a goal to address the need (i.e., to reduce

the deficit), which may then lead to corresponding actions (based on innate behaviors initially). In the process, even the perception of the individual might be modulated somewhat so that, for example, it focuses more on the perceptual features that are relevant to the pressing needs.

Similar processes happen when there is a growing "deficit" in terms of a socially oriented need, such as the need for interaction with others (the need for affiliation and belongingness). In such a case, the individual may similarly generate a corresponding goal, which in turn leads to corresponding actions (initially based on whatever primitive behavioral repertoire that is available, for example, by crying).

Gradually, with trial and error, the individual learns more and more how to meet various needs, in part based on successes or failures in attending to these needs. The individual learns what actions to perform in what situations, in order to fulfill an outstanding need. When an outstanding need is fulfilled to some extent, pleasure is felt—a positive reinforcement. Based on such reinforcement, the individual learns to associate needs with concrete goals and in turn also learns to associate goals with actions that best accomplish the goals. Through the trial-and-error process, the individual increases competence (developing more effective and more complex routines or skills), which helps to deal with similar or more difficult situations in the future.

In this process, the individual may experience a variety of affect states, which facilitate learning and performance of actions: *elation* when goals are accomplished (needs are met and positive reinforcement is received), *frustration* when unable to accomplish goals despite efforts, and *anxiety* when negative consequences (thus negative reinforcement) are expected, and so on.

Moreover, gradually, the individual starts to develop explicit (symbolic) knowledge regarding actions (i.e., explicit procedural knowledge), beyond implicit associations acquired through trial and error (that is, implicit procedural knowledge discussed above). Explicit procedural knowledge may be extracted on the basis of already acquired implicit procedural knowledge (through "bottom-up learning"). Explicit knowledge in turn enables the individual to reflect on the knowledge and the situations, to plan ahead, to communicate the knowledge to others, and so on. Thus, implicit and explicit procedural knowledge together may lead to more effective coping with the world (i.e., a synergy effect).

Furthermore, even general knowledge that is not directly tied to actions (namely, declarative knowledge) may be generated over time. It

may be generated on the basis of acquired procedural knowledge (which may involve bottom-up learning) or from information provided by others (which may involve top-down learning). Such declarative knowledge adds more capabilities to the individual.

So, drawing lessons from this scenario, according to the Clarion framework, an individual starts small: there are only minimum initial structures. Some of these initial structures have to do with behavioral predispositions (e.g., evolutionarily pre-wired instincts and reflexes); some others have to do with learning capabilities; yet some others have to do with motivation. Together they constitute the genetic and biological pre-endowment.

Most of the mental contents within an individual have to be "constructed" (learned) during the course of individual ontogenesis. Development occurs through interacting with the world (physical and sociocultural). It leads to the formation of various implicit, reactive behavioral routines (skills), which in turn lead to explicit (symbolic) representation. The generation of explicit representation is, to a significant extent, determined by implicit mental contents within an individual. Of course, there is also another source: sociocultural influence, including through symbols employed in a culture.

Overall, the mind of an individual is mostly activity-based, action-oriented, and embedded in the world. An individual often interacts with the world in a rather direct and immediate way (Heidegger, 1927; Dreyfus, 1992), although more explicit, more contemplative, less direct ways may develop within the individual.

In Chapter 3, another example will pick up from here, continuing the learning processes discussed thus far, adding more details. But now, I will explore further the desiderata that were identified and illustrated above by examining the relevant empirical literature.

2.3. Justifying the Desiderata

Here I will not attempt to address all of the desiderata enumerated earlier, but instead will focus on some more controversial ones. Some points such as sequentiality, routineness, and trial-and-error adaptation have been thoroughly discussed in Sun (2002), and they seem almost self-evident by now. These will not be discussed again here.

2.3.1. Implicit-Explicit Distinction and Synergistic Interaction

To justify the Clarion worldview, I will start by examining in detail the distinction between implicit and explicit processes, which is the foundation of the Clarion framework. The theoretical distinction between implicit and explicit processes, as well as its ecological-functional significance, has been argued in the past in many psychological theories. See, for example, Reber (1989), Seger (1994), and Sun (1994, 2002).

First, the distinction of implicit and explicit processes has been empirically demonstrated in the implicit memory literature (e.g., Roediger, 1990; Schacter, 1987). The early work on amnesic patients showed that these patients might have intact implicit memory while their explicit memory was severely impaired. Warrington and Weiskrantz (1970), for example, demonstrated that when using "implicit measures," amnesic patients' memory was as good as normal subjects; but when using "explicit measures," their memory was far worse than normal subjects. The explicit measure used included free recall and recognition, while the implicit measures used included word-fragment naming and word completion. It has been argued that the implicit measures reflected unconscious (implicit) processes because amnesic patients were usually unaware that they knew the materials (Warrington & Weiskrantz, 1970). Such results demonstrating dissociations between implicit and explicit measures have been replicated in a variety of circumstances.

Second, Jacoby (e.g., Jacoby, 1983) demonstrated that implicit and explicit measures might be dissociated among normal subjects as well. Three study conditions were used: generation of a word from a context, reading aloud a word in a meaningful context, and reading aloud a word out of context. The explicit measure used was recognition (from a list of words), while the implicit measure was perceptual identification (from fast presentations of words). The results showed that, using the explicit measure, generated words were remembered the best and words read out of context were remembered the least. However, using the implicit measure, the exact opposite pattern was found. Other dissociations were also found from other manipulations (see, e.g., Roediger, 1990; Schacter, 1987). Toth, Reingold, and Jacoby (1994) devised an inclusion-exclusion procedure for assessing implicit and explicit contributions, which also provided strong indications of dissociation.

Third, the distinction of implicit and explicit processes has also been empirically demonstrated in the implicit learning literature (Reber, 1989;

Seger, 1994; Cleeremans et al., 1998). For example, serial reaction time tasks involve learning of a repeating sequence, and it was found that there was a significant reduction in response time to repeating sequences (compared to random sequences). However, subjects were often unaware that a repeating sequence was involved (e.g., Lewicki, Czyzewska, & Hoffman, 1987). Similarly, dynamic process control tasks involve learning of a relation between the input and the output variables of a controllable system, through interacting with the system. Although subjects often did not recognize the underlying relations explicitly, they nevertheless reached a certain level of performance in these tasks (e.g., Berry & Broadbent, 1988). In artificial grammar learning tasks, subjects were presented with strings of letters that were generated in accordance with a finite state grammar. After memorization, subjects recognized new strings that conformed to the artificial grammar, although subjects might not be explicitly aware of the underlying grammar (except for some fragmentary knowledge; Reber, 1989). In all, these tasks shared the characteristic of implicit learning processes being involved to a significant extent.

Generally speaking, explicit processing may be described mechanistically as being based on rules in some way, while implicit processing is more associative (Sun, 2002). Explicit processing may involve the manipulation of symbols, while implicit processing involves more instantiated knowledge that is more holistically associated (Sun, 1994, 2002; Reber, 1989). While explicit processes require attention, implicit processes often do not (Reber, 1989). Explicit processes may compete more for resources than implicit processes. Empirical evidence in support of these differences can be found in, for example, Reber (1989), Seger (1994), and Sun (2002).

Similar distinctions have been proposed by other researchers, based on similar or different empirical or theoretical considerations (Grossberg, 1982; Milner & Goodale, 1995; McClelland, McNaughton, & O'Reilly, 1995; Erickson & Kruschke, 1998). There have also been many other tasks that may be used for demonstrating implicit processes, such as various concept learning, reasoning, automatization, and instrumental conditioning tasks (for a review, see Sun, 2002). In particular, it is worth noting that in social psychology, there have been a number of dual-process models that are roughly based on the coexistence of implicit and explicit processes (see, e.g., Chaiken & Trope, 1999). Evans and Frankish (2009) included a collection of theories and models based on this kind of distinction. Taken together, the distinction between explicit and implicit

processes may be supported in many ways, although details of some of these proposals might be different (or even contradictory to each other in some way). Although some researchers disputed the existence of implicit processes based on the imperfection and incompleteness of tests for explicit knowledge (e.g., Shanks & St. John, 1994), there is an overwhelming amount of evidence in support of the distinction (see Sun, 2002 for further arguments).

Now the question is whether these different types of processes reside in separate memory stores (memory modules or systems) or not. There have been debates in this regard, and differing views exist (Roediger, 1990). Squire (1987) proposed that memory be divided into declarative and procedural memory, with the former further divided into episodic and semantic memory and the latter into skills, priming, classical conditioning, and so on. According to Squire (1987), declarative memory was explicit while procedural memory was implicit. Tulving and Schacter (1990) incorporated some features of the one-system view on memory while preserving the separation of explicit and implicit memory. They proposed that there should be multiple priming systems in the implicit memory so that dissociations among different implicit measures could be accounted for. This proposal addressed some objections raised by the proponents of the one-system view. Sun, Slusarz, and Terry (2005) provided a theoretical interpretation of a variety of learning data (related to process control, serial reaction time, and other tasks, as mentioned earlier), based on the multiple memory stores view.

Work in neuroscience shows some evidence for the existence of distinct brain circuits for implicit and explicit processes (i.e., separate memory stores). For instance, the work of Schacter (1990), Buckner et al. (1995), Posner, DiGirolamo, and Fernandez-Duque (1997), Goel, Bruchel, Frith, and Dolan (2000), Lieberman (2009), and so on provided some such indications. There have also been arguments that implicit memory represents a phylogenetically older system. This system may be more primitive but yet powerful on behavior.

However, as pointed out by Hintzman (1990), "once the model has been spelled out, it makes little difference whether its components are called systems, modules, processes, or something else; the explanatory burden is carried by the nature of the proposed mechanisms and their interactions, not by what they are called" (p.121). The debates regarding whether dissociations and distinctions of various kinds mentioned above

point to different processes or difference systems should be seen in this light. Sun (2012) provided further arguments in this regard.

In relation to the ecological-functional perspective articulated before, it should be noted that there have been some indications that explicit processes are evolutionarily newer than implicit processes (Reber, 1989). But the juxtaposition of the two is functional. It is functional and thus evolutionarily advantageous, especially because the interaction between the two types of processes may lead to synergy in the form of better, more accurate, and/or faster performance in a variety of circumstances (as I have extensively argued in prior work). Further discussions of synergy from the interaction, both in an empirical sense and in a computational sense, can be found in Section 2.5, as well as in Sun (2002), Sun, Slusarz, and Terry (2005), Helie and Sun (2010), and so on. Synergy, although not universal (i.e., not present in all circumstances), has been amply demonstrated in a wide variety of situations. Therefore, the division of implicit and explicit processes may conceivably be favored by natural selection. In addition, the separation of the two types of information, knowledge, mechanisms, and processes enables the adoption of each type as appropriate for different types of situations. For example, highly complex situations may be better handled by implicit processes, while explicit processes operating in a more precise way may be better for more clear-cut situations (Sun, 2002; Sun & Mathews, 2005; Lane, Mathews, Sallas, Prattini, & Sun, 2008). Furthermore, the division also enables parallel applications of the two types for different purposes simultaneously. So, putting everything together, the separation and the interaction of these two types of processes are psychologically advantageous.

2.3.2. Separation of the Implicit-Explicit and the Procedural-Declarative Distinction

I now turn to the distinction between procedural and declarative processes (i.e., action-centered and non-action-centered processes in the action-centered and the non-action-centered subsystem, respectively, as will be explained later) and its orthogonality with the implicit-explicit distinction (which might be a more controversial point).

The distinction between procedural and declarative processes has been advocated by Anderson (1983), Squire (1987), and many others (although some details vary across different proposals). Procedural

processes involve knowledge that is specifically concerned with actions (and action sequences) in various circumstances, that is, how to do things. Declarative processes involve knowledge that is not specifically concerned with actions but more about objects, persons, events, and so on, in more generic terms (i.e., the "what", not the "how"). The major factor that distinguishes procedural and declarative processes seems to be the action-centeredness or the lack thereof—in other words, the procedural versus nonprocedural nature.

Evidence in support of this distinction includes many studies of skill acquisition in both high- and low-level skill domains (e.g., Kanfer & Ackerman, 1989; Anderson & Lebiere, 1998; see also Proctor & Dutta, 1995). These studies included both experimental work on human subjects, as well as modeling/simulation and other work aimed at theoretical interpretations. They showed that this distinction provides useful insight in interpreting a range of data and phenomena. For instance, Anderson (1983) used this distinction to account for changes in performance resulting from extensive practice, based on data from a variety of skill-learning studies. According to Anderson, the initial stage of skill development is characterized by the acquisition of declarative knowledge. During this stage, the learner must explicitly attend to this knowledge in order to perform a task. Through practice, procedures develop that may accomplish the task without declarative knowledge.

Let us examine the relation between the procedural-declarative distinction and the implicit-explicit distinction. In Anderson (1983), declarative knowledge was assumed to be consciously accessible (i.e., explicit): subjects could report on and manipulate such knowledge. Procedural knowledge was assumed not: it led to actions without explicit accessibility. Thus, in Anderson (1983), the two dichotomies were merged into one.

On the other hand, in ACT-R as described by Anderson and Lebiere (1998), each individual piece of knowledge, be it procedural or declarative, involved both subsymbolic and symbolic representation. Symbolic representation was used for denoting semantic labels and structural components of each concept, while subsymbolic representation was used for expressing its activation and other numerical factors. One interpretation was that the symbolic representation was explicit while the subsymbolic representation was implicit (either for declarative knowledge, or for both declarative and procedural knowledge). This view constituted another take on the relationship between the two dichotomies.

According to the first view above, the difference in action-centeredness (i.e., the procedural versus nonprocedural nature) seems the main factor in distinguishing the two types of knowledge, while accessibility (i.e., implicitness versus explicitness) is a secondary factor. I believe that this view unnecessarily confounds two aspects: action-centeredness and accessibility, and can be made clearer by separating the two dimensions. Action-centeredness does not necessarily go with implicitness (inaccessibility), as shown by, for example, the experiments of Stanley, Mathews, Buss, and Kotler-Cope (1989), Willingham, Nissen, and Bullemer (1989), or Sun et al. (2001). Likewise, non-action-centeredness does not necessarily go with explicitness (accessibility) either, as shown by conceptual priming and other implicit memory experiments (e.g., Schacter, 1987; Moscovitch & Umilta, 1991) or by experiments demonstrating implicit information (e.g., Hasher & Zacks, 1979; Nisbett & Wilson, 1977). Some might choose to group all implicit memory (including semantic, associative, and conceptual priming) under procedural memory, but such views confound the notion of "procedural" and thus are not adopted here. In light of the above, these two dimensions need to be separated.

The alternative view that each individual piece of knowledge (either procedural or declarative, or both) involves both implicit and explicit parts is also problematic. Such a view entails a close coupling between implicit and explicit processes, which is highly questionable. The underlying assumption that every piece of knowledge (either declarative or procedural, or both) has an explicit part contradicts the fact that some knowledge may be completely implicit (e.g., Lewicki et al., 1987; Cleeremans, Destrebecqz, & Boyer 1998). This raises the question of whether such a tight coupling or a more separate organization, for example, having these two types of knowledge in separate memory stores, makes better sense.

Squire (1987) proposed that memory should be divided into declarative and procedural memory, with the former further divided into episodic (working) and semantic (reference) memory and the latter into skills, priming, classical conditioning, and other memory stores; declarative memory was explicit while procedural memory was implicit. However, this view would have trouble accounting for implicit declarative memory (which was clearly not procedural; e.g., conceptually driven priming; Roediger, 1990). Explicit procedural memory was also not accounted for in this view (Sun, Slusarz, & Terry, 2005). This view unnecessarily confounded the notions of "procedural" and "declarative."

As a more natural, more intuitively appealing alternative to those views above, I proposed the separation of the two dichotomies—treating them as logically separate from (i.e., orthogonal to) each other (Sun, Zhang, & Mathews, 2009; Sun, 2012). Arguments in favor of this view can be found in the literature. For example, Willingham (1998) argued based on empirical data that motor skills (a type of procedural process) consisted of both implicit and explicit processes. Rosenbaum et al. (2001) argued based on empirical data that intellectual skills and perceptual-motor skills alike were made up of implicit and explicit knowledge. In other words, procedural (action-centered) processes, ranging from high-level intellectual skills to perceptual-motor skills, may be divided into implicit and explicit processes.

Similarly, declarative (non-action-centered) processes may also be divided (Tulving & Schacter, 1990). There is no reason to believe that all implicit knowledge is procedural (as implied by some of the aforementioned views). Some implicit knowledge may be declarative (i.e., non-action-centered). In terms of functional consideration, having separate implicit and explicit declarative memory stores may allow different tasks to be tackled simultaneously in these separate memory stores (e.g., while thinking explicitly about one task, letting intuition work on another). Sun (1994) and Sun and Zhang (2006) showed that through dividing declarative memory into explicit and implicit modules, some reasoning data could be naturally accounted for. Furthermore, Helie and Sun (2010) showed that this division accounted well for creative problem solving (which otherwise would be difficult to account for).

On this view, procedural and declarative knowledge reside separately in procedural and declarative memory stores respectively, which are representationally different (Sun et al., 2009; Sun, 2012). Procedural knowledge (e.g., in procedural memory located in the action-centered subsystem as will be detailed later) may be represented by either action rules (explicit) or action neural networks (implicit), both of which are centered on situation-action mappings. Declarative knowledge (e.g., in declarative memory located in the non-action-centered subsystem as will be detailed later), on the other hand, is represented by either associative rules (explicit) or associative neural networks (implicit), in both of which knowledge is represented in a non-action-centered way.

As mentioned before, in a similar fashion but orthogonally, implicitness/explicitness is also distinguished by representation. Implicit knowledge may be represented using connectionist distributed representation

(such as in a hidden layer of a Backpropagation network), which is less accessible to an individual possessing it, relatively speaking (Sun, 1999, 2002), while explicit knowledge may be represented using symbolic-localist representation, which is relatively more accessible (Kirsh, 1990). Implicit and explicit knowledge thus reside in different memory modules with different representations. Moreover, in this way, the two dichotomies are separate from each other: that is, there are both implicit and explicit procedural (action-centered) memory stores, and both implicit and explicit declarative (non-action-centered) memory stores.

This four-way division is functional according to the ecological-functional perspective, because of (1) the division of labor between explicit and implicit memory (one for storing explicit knowledge that is more crisp and the other for storing implicit knowledge that is more complex), and (2) the division of labor between declarative and procedural memory (one for storing general knowledge and the other for storing knowledge oriented specifically toward action decision making). The divisions of labor led to both the separation and the interaction of these different types of knowledge. The separation ensures that different types of knowledge may be found separately and thus relatively easily, while the interaction helps to bring together different types of knowledge when needed (Klein, Cosmides, Tooby, & Chance, 2002; Sun & Zhang, 2006; Sun et al., 2007, 2009), to ensure performance and synergy as mentioned before and as will be discussed later (Sun, Slusarz, & Terry, 2005). Furthermore, the separation allows different processes to work on different tasks possibly simultaneously and thus enhances the overall functionality. Thus, the four-way division is in keeping with the ecological-functional considerations.

The orthogonality of the procedural-declarative distinction and the implicit-explicit distinction will be further argued for later in Chapter 3 when addressing the issue of semantic versus episodic memory.

2.3.3. Bottom-Up and Top-Down Learning

The interaction between implicit and explicit processes during learning includes top-down learning (explicit learning first and implicit later), bottom-up learning (implicit learning first and explicit later), and parallel learning (simultaneous implicit and explicit learning).

However, bottom-up learning may be more essential to everyday activities (Sun et al., 2001). There are various indications and arguments for bottom-up learning, including (1) philosophical arguments, such as Heidegger (1927) and Dewey (1958), in which the primacy of direct interaction with the world (in a mostly implicit way) is emphasized, and (2) psychological evidence of the acquisition and the delayed explication of implicit knowledge. Let us look into some psychological findings below.

It has been found empirically that in skill learning, subjects' ability to verbalize is often independent of their performance (Berry & Broadbent, 1988). Furthermore, performance typically improves earlier than explicit knowledge that can be verbalized by subjects (Stanley et al., 1989). For instance, in a process control task, although the performance of subjects quickly rose to a high level, their verbal knowledge improved more slowly: subjects could not provide usable verbal knowledge until near the end of their training (Stanley et al., 1989). This phenomenon has also been demonstrated by Reber and Lewis (1977) in artificial grammar learning. A study of bottom-up learning was carried out by Sun et al. (2001) using a complex minefield navigation task. In all of these tasks, it appears easier to acquire implicit skills than explicit knowledge and hence the delay in the development of explicit knowledge. The delay indicates that explicit learning may be triggered by implicit learning. Explicit knowledge may be in a way "extracted" from implicit skills. (However, in some other tasks, explicit and implicit knowledge appear to be more closely associated.)

In the context of discovery tasks, Bowers, Regehr, Balthazard, and Parker (1990) showed evidence of the explication of implicit knowledge. When subjects were given patterns to complete, they showed implicit recognition of what a proper completion might be, although they did not have explicit recognition of a correct completion. The implicit recognition improved over time until an explicit recognition was achieved. Siegler and Stern (1998) also showed in an arithmetic problem that children's strategy changes often occurred several trials earlier than their explicit recognition of strategy changes. Stanley et al. (1989), Seger (1994), and Sun et al. (2001) suggested that because explicit knowledge lagged behind but improved along with implicit knowledge, explicit knowledge could be viewed as obtained from implicit knowledge.

Several developmental theorists considered a similar delayed explication process. Karmiloff-Smith (1986) suggested that developmental

changes involved "representational redescription." In children, first low-level implicit representations of stimuli were formed. Then when more knowledge was accumulated and stable behavior patterns developed, it was through a redescription process that more abstract representations were formed that transformed low-level representations and made them more explicit. This redescription process was repeated a number of times, and a verbal form of representation emerged. Mandler (1992) proposed another kind of redescription. From perceptual stimuli, relatively abstract "image-schemas" were extracted that coded several basic types of movements. Then, on top of such image-schemas, concepts were formed using information therein. In a similar vein, Keil (1989) viewed conceptual representations as composed of an associative component and a theory component. Developmentally, there was a shift from associative to theory-based representations. These ideas and the empirical data on which they were based testify to the ubiquity of the implicit-to-explicit transition.

In the other direction, top-down learning usually occurs when explicit knowledge is available from external sources, or when it is relatively easy to learn such knowledge (compared with learning corresponding implicit knowledge). Explicit knowledge, directly received from external sources or otherwise learned, is then assimilated into an implicit form. For example, learning to play chess would be a good illustration. One often first learns the basic rules of chess and some essential guidelines as to what to do in prototypical situations. One may then develop more complex and more nuanced knowledge that may be largely implicit. See, for example, detailed discussions in Dreyfus and Dreyfus (1987) on such a process.

2.3.4. Motivational and Metacognitive Control

Motivational and metacognitive control captures important elements in the interaction between an individual and his or her physical and social worlds. Motivation and metacognition interact closely with cognition (in its narrow sense; Simon, 1967). For instance, when interacting with the world, an individual must attend to his or her own basic needs (such as hunger and thirst) and must also know to avoid danger and so on (Toates, 1986). Actions are chosen in accordance with the individual's needs for survival and functioning in the world. In other words, innate motives must be there (Reiss, 2004). On that basis, specific goals may be chosen. The individual must be able to focus activities with respect to specific

goals. However, the individual needs to be able to give up some goals when necessary (Toates, 1986). In order to address these considerations, motivational and metacognitive processes are necessary.

Without motivational mechanisms and processes, an individual would be literally aimless. One would have to rely solely on external "feedback" (reinforcement, reward, punishment, and so on) in order to learn. But such external feedback begs the question of how it may be obtained in the real world in general (aside from a few simple cases). With a more sophisticated motivational mechanism incorporating innate motives, feedback may be generated internally in response to the condition of the world. An individual may be able to learn on that basis. A sophisticated motivational mechanism is also important for facilitating social interaction (with innate, socially oriented motives; Sun, 2006).

Similarly, without metacognitive monitoring and regulation, an individual might be blindly single minded. One might not be able to flexibly adjust behavior. The ability to reflect on and to modify dynamically one's own behaviors is important to function effectively in complex environments (Reder, 1996; Mazzoni & Nelson, 1998; Sun & Mathews, 2012). Social interaction is made possible by the (at least partially innate) ability of individuals to reflect on and to modify their own behaviors. Metacognition enables individuals to interact with each other more effectively, for example, by avoiding social impasses—impasses that are created because of the radically incompatible behaviors of different individuals (Sun, 2006).

The duality of representation is present in the motivational and metacognitive processes, as in other processes. But, the questions of exactly how internal needs and motives are represented, how they affect performance, and how one exerts control over one's own cognitive processes will be addressed in subsequent chapters.

2.4. Four Subsystems of Clarion

2.4.1. Overview of the Subsystems

Based on the desiderata (and their justifications) discussed above, Clarion was conceived as a comprehensive cognitive architecture consisting of a number of distinct but interacting subsystems. These subsystems capture distinct types of representational contents discussed in these desiderata

above, in a functionally somewhat separate but mutually dependent and dynamically interacting fashion. These subsystems include

- the action-centered subsystem (the ACS)
- the non-action-centered subsystem (the NACS)
- the motivational subsystem (the MS)
- the metacognitive subsystem (the MCS)

See Figure 2.1 for a sketch of these four subsystems. Clearly, these subsystems correspond to the types of representational contents enumerated in Chapter 1 and further discussed above.

The respective roles of these subsystems may be briefly summed up as follows:

- The role of the ACS is to control actions with procedural knowledge, regardless of whether the actions are for external

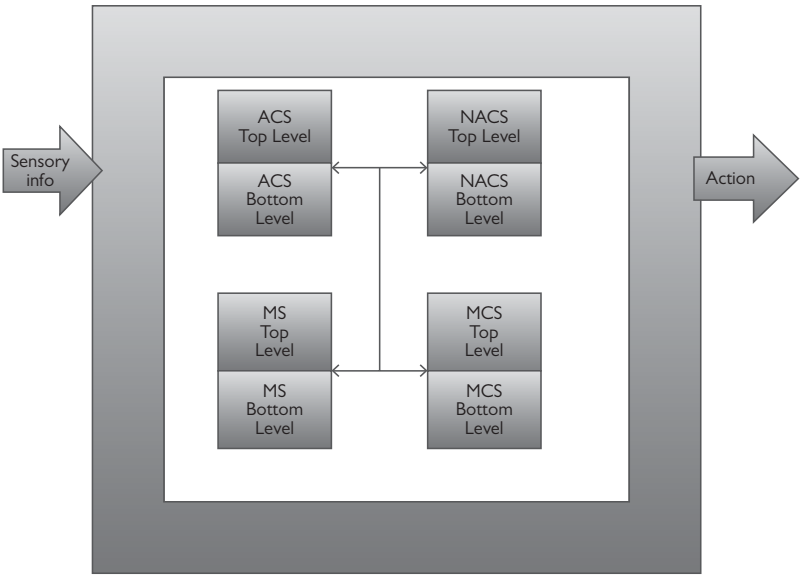

Figure 2.1. The subsystems of the Clarion cognitive architecture. The major information flows are shown with arrows. ACS stands for the action-centered subsystem. NACS stands for the non-action-centered subsystem. MS stands for the motivational subsystem. MCS stands for the metacognitive subsystem. See the text for more explanations. See chapters 3 and 4 for technical details of the subsystems.

physical movements or internal mental operations (e.g., "executive control").
- The role of the NACS is to maintain general (i.e., declarative) knowledge for retrieval of appropriate information in different circumstances and to perform inferences on that basis (ultimately in the service of action decision making by the ACS).
- The role of the MS is to provide underlying motivations for perception, action, and cognition.
- The role of the MCS is to monitor and regulate the operations of the other subsystems on the fly.

Each of the subsystems serves a unique function, and together they form a functioning cognitive architecture.

Each of these subsystems consists of two "levels" of representation—that is, a dual-representational structure as discussed earlier (and extensively argued for in Sun, 2002, and Sun, Slusarz, & Terry, 2005). Generally speaking, in each subsystem, the top "level" encodes explicit knowledge, using symbolic-localist representations, and the bottom "level" encodes implicit knowledge, using connectionist distributed representations (more on this in Chapter 3; Sun, 1994).

The relatively inaccessible nature of implicit knowledge at the bottom level (i.e., in implicit memory) may be captured computationally by subsymbolic, distributed representation. This is because distributed representational units (e.g., in a hidden layer of a Backpropagation network) are capable of accomplishing computations but are subsymbolic and generally not individually meaningful (Rumelhart et al., 1986; Sun, 1994). This characteristic of distributed representation, which renders the representational form less accessible computationally, accords well with the relative inaccessibility of implicit knowledge in a phenomenological sense (Reber, 1989; Seger, 1994). Note that phenomenological accessibility refers to the direct and immediate availability of mental content for the major operations that are responsible for, or concomitant with, consciousness, such as introspection, higher-order thoughts, and verbal reporting (Sun, 1999).

In contrast, explicit knowledge at the top level (in explicit memory) may be captured in computational modeling by symbolic or localist representation, in which each unit is more easily interpretable and has a clearer conceptual meaning. This computational characteristic of symbolic or localist representation captures the phenomenological characteristic

of explicit knowledge being more accessible and more manipulatable (Sun, 1994).

The dichotomous difference in the representations of the two different types of knowledge leads naturally to a two-level structuring whereby each level uses one kind of representation and captures one corresponding kind of knowledge, memory, and process (implicit or explicit).

The two levels interact, for example, by cooperating in action decision making, through an integration of the action recommendations from the two levels respectively, as well as by cooperating in learning through a bottom-up and a top-down learning process (more in Chapter 3).

Below, I will briefly sketch the working of these subsystems, to give a general idea about each subsystem, without elaboration and without further justifications. Details, including technical specifics and justifications, will follow in the next two chapters.

2.4.2. The Action-Centered Subsystem

The action-centered subsystem (the ACS) is the most important part of Clarion. The ACS controls actions with procedural knowledge. That is, the ACS embodies procedural processes.

Within the ACS, the process for action decision making is essentially as follows (Sun, 2002):

> Observing the current state of the world (the observable input state), the two levels of processes within the ACS (implicit and explicit) make their decisions in accordance with their respective knowledge, and their outcomes are integrated. Thus, a final selection of an action is made and the action is then performed. The action changes the world in some way. Comparing the changed input state with the previous input state, learning occurs. The cycle then repeats itself.

In the bottom level of the ACS, implicit reactive routines are formed. An action has a value that is an evaluation of the overall "quality" of the action in a given state. This value may be the cumulative reinforcement that can be received (see Chapter 3). The state may include the chosen goal. In any state, an action may be selected based on the values of all possible actions, for example, by choosing the action with the highest value.

To acquire the values, various learning algorithms may be used (when the values are not innate), especially reinforcement learning algorithms implemented in neural networks with subsymbolic representation (Sun & Peterson, 1998). A reinforcement learning algorithm (such as *Q-learning*) may compare the values of successive actions and adjusts a value function on that basis (Bertsekas and Tsitsiklis, 1996; Sutton and Barto, 1998). Through gradual learning, the bottom level of the ACS develops implicit reactive routines (implicit sequential reactive skills). (Details will be discussed in Chapter 3.)

The bottom level of the ACS is modular; that is, a number of relatively small neural networks (e.g., Backpropagation networks; Rumelhart et al., 1986) coexist, each of which is adapted to specific modalities, tasks, or groups of input stimuli. While some of these may be innate, others learn through interaction with the world. This is consistent with the modularity hypothesis that much processing is done by specialized and (to some extent) encapsulated processors (e.g., Fodor, 1983, as discussed in Chapter 1).

In the top level of the ACS, explicit knowledge is captured in the form of individual nodes (representing concepts) and links connecting them (representing rules). In Clarion, concepts are generally "redundantly" represented: denoted by a unitary (localist) representation at the top level as well as specified through multiple (micro)features at the bottom level (i.e., through distributed representation). At the top level, a single node is set up to represent a concept, known as a *chunk node*, which is specifically for representing that concept. The chunk node connects to its corresponding multiple (micro)feature nodes (in a distributed representation) at the bottom level. Together they constitute a *chunk*: that is, the representation of a concept. At the top level of the ACS, *action rules* are represented by directed links from one chunk node (the condition chunk node) to another (the conclusion or action chunk node).

There are many ways in which explicit knowledge at the top level may be learned, including independent hypothesis-testing learning and bottom-up learning (as touched upon before). Explicit representation at the top level can also be assimilated into implicit reactive routines at the bottom level through top-down learning. Details of learning will be discussed in Chapter 3.

Note that the term "rule" has a very specific meaning here. It denotes explicit encoding of knowledge, in the form of explicit association from

one explicit concept to another, where "explicit" refers to direct computational accessibility. As outlined before, in my theoretical interpretation, explicit representation amounts to symbolic-localist encoding (Sun, 1994, 2002). This notion of "rule" is not necessarily relevant to the philosophical discourses on rule following and related issues.

Note also that the term "(micro)feature" is used to indicate that, in a distributed representation, basic elements of the representation may or may not be interpretable; that is, they may or may not have clear conceptual meanings. Generally, in the connectionist literature, the term "microfeature" is used to denote features that are not individually meaningful but together capture contents of concepts. Here the parenthesis is added to suggest that it may or may not have this characteristic.

2.4.3. The Non-Action-Centered Subsystem

The non-action-centered subsystem (the NACS) is for representing declarative knowledge and for performing declarative memory retrievals and inferences of various kinds.

At the bottom level of the NACS, implicit non-action-centered knowledge is encoded by "associative memory" networks with distributed representation. An association is formed by mapping an input to an output. For instance, Backpropagation networks or Hopfield networks may be used to establish associations (Rumelhard et al., 1986).

On the other hand, at the top level of the NACS, explicit non-action-centered knowledge is encoded. A single node is set up in the top level to represent a concept, known as a *chunk node* (a localist representation). The chunk node at the top level connects to its corresponding (micro)feature nodes (i.e., distributed representation) in the bottom level. Additionally, at the top level of the NACS, links between chunk nodes encode explicit associations between concepts, known as *associative rules*.

In addition to applying associative rules, similarity-based reasoning may be automatically carried out within the NACS through the interaction of the two levels (i.e., through cross-level links and the resulting top-down and bottom-up activation flows; more on this in Chapter 3).

Within the NACS, there exists the distinction between semantic memory and episodic memory. This distinction has been fairly well established

(see, e.g., Tulving, 1985). Semantic knowledge (in the semantic memory of the NACS) is not tied to specific experiences, although it might be a result of past experiences (see Chapter 3). In contrast, episodic knowledge (in the episodic memory of the NACS) is directly tied to specific past experiences, with specific episodic information included as part of the encoding.

As in the ACS, top-down or bottom-up learning may take place in the NACS as well, either to extract explicit knowledge at the top level from implicit knowledge in the bottom level or to assimilate explicit knowledge of the top level into implicit knowledge at the bottom level.

The NACS is normally under the control of the ACS, through its actions. Clarion embodies the belief that cognition-psychology is activity-based, action-oriented, and embedded in the world (as indicated by the desiderata earlier). Therefore, one overarching principle in Clarion is: action first and reasoning in the service of action. Furthermore, executive function is embodied in part by the ACS, which is thus responsible for controlling both internal actions and external actions (at a high level at least). Thus, in Clarion, the ACS directs the NACS, in addition to deciding on external actions.[2]

2.4.4. The Motivational Subsystem

An individual's behavior is normally far from being random. It may be traced to deep-rooted needs (motives), either innate or acquired (Tolman, 1932; Murray, 1938; Maslow, 1943; Toates, 1986; Weiner, 1992; Masmoudi, Dai, & Naceur, 2012). The motivational subsystem (the MS) is thus a necessary part of a psychologically realistic cognitive architecture, necessary for proper psychological functioning of an individual.

The MS of Clarion is centered on basic human motives, termed *drives*, and their interactions and competitions (Sun, 2009), which lead to specific goals and in turn to actions that accomplish goals. So the MS is concerned with why an individual does what he or she does. Simply saying that one chooses actions to maximize gains, rewards, or payoffs leaves open

2. Note that the ACS often makes fairly "high-level" external action decisions, which do not involve details of motor control; for example, "I should move to the right side to avoid the speeding car." Internal actions may be decided similarly; for example, "I don't know which action to take, so let me think about it" (i.e., applying the NACS). Therefore, at a "high level," internal and external action decisions are similarly made. Both are controlled by the ACS.

the question of what determines these things. Through the MS, Clarion addresses the essential human motives (drives), formed by a long evolutionary history, especially in relation to survival and continuation, as well as functioning in the social and physical world. These basic motives (drives) have been explored in social psychology and psychology of motivation (e.g., Murray, 1938; Maslow, 1943; Weiner, 1992).

Among these drives, low-level drives are concerned mostly with physiological needs, such as hunger, thirst, physical danger, and so on. High-level drives are more socially oriented. Some of these drives are primary, in the sense of being evolutionarily "hard-wired." Low-level primary drives include hunger, thirst, and so on. High-level primary drives include, for example, seeking social status, following social code, maintaining fairness, and so on (see Chapter 4). Although such primary drives are relatively unalterable, there are also "derived" drives, which are secondary, more changeable, and acquired mostly in the process of satisfying primary drives (Sun, 2009).

Like dual representations in the other subsystems, dual motivational representation is in place in the MS (Tolman, 1932; Murray, 1938; Deci, 1980; Sun, 2009). Beyond drives, explicit goals are also important, for instance, to the working of the ACS. Explicit goals (e.g., *find food*) may be generated based on internal drive activation (e.g., a high activation of the hunger drive).

2.4.5. The Metacognitive Subsystem

The existence of a variety of drives and goals that they give rise to leads to the need for metacognitive regulation and control. The metacognitive subsystem (the MCS) of Clarion is closely tied to the MS. The MCS monitors and regulates cognitive processes based on information from the MS as well as from other sources.

In the MCS, regulation and control may be in the forms of setting goals (for the ACS) based on drive activation, choosing explicit or implicit processes or their combination thereof (for the ACS), filtering input and output information (for the ACS or the NACS), choosing learning or reasoning methods, interrupting and changing ongoing processes, setting essential parameters, and so on. Regulation and control can also be carried out through providing reinforcement for reinforcement learning within the ACS. All of the functions above may be performed on the basis of the MS.

As argued by, for example, Reder (1996) and Sun and Mathews (2012), metacognition includes both implicit and explicit processes. The MCS thus also includes two levels: the top level and the bottom level. The bottom level is implicit while the top level is explicit, consistent with the overall structure of Clarion (Sun & Mathews, 2012).

2.4.6. Parameters of the Subsystems

The stable structures of these subsystems and their stable interactions (as identified above), as well as the relatively stable parameters of these subsystems (to be identified later in subsequent chapters), make the system behave in a relatively stable and predictable way. They help to capture the (relatively) stable characteristics of an individual (including, e.g., "personality").

With adjustable parameters (as will be identified later), different characteristics of different individuals (i.e., individual differences) may also be captured. Generally speaking, individual differences as captured by different parameters in the subsystems might be attributed to two sources. They may be attributed, in part, to innate (in-born, hardwired) differences (due to biological, including genetic, factors) but also partly to different individual experiences (including different individual experiences of sociocultural influences). Thus, some parameters tend to be fixed, while others are subject to "tuning" to various extents through experiences (especially during ontogenesis).

Note that for fixed parameters within these subsystems, their (default) values are discussed in the companion technical book (see also various technical papers on Clarion). For adjustable parameters, it is important to consider them in broader contexts. These broader contexts, including the important aspect of learning, will be addressed in chapters 3 and 4 and examples shown in chapters 5, 6, and 7.

2.5. Accounting for Synergy within the Subsystems of Clarion

Two important predictions from the Clarion framework outlined above were the synergy effects between implicit and explicit procedural (action-centered) processes within the ACS, and the synergy effects between implicit and explicit declarative (non-action-centered)

processes within the NACS. These two predictions have been validated through examining empirical literature and through simulation studies (Sun, Slusarz, & Terry, 2005; Sun & Zhang, 2006; Helie & Sun, 2010). I briefly examine these two points below.

2.5.1. Accounting for Synergy within the ACS

The Clarion framework predicted that there should be synergy between implicit and explicit procedural (action-centered) processes, resulting from their interaction in learning and in action decision making (Sun, Slusarz, & Terry, 2005). From the ecological-functional perspective, there should be no surprise that the interaction of these two types of processes may be synergistic. One naturally expects that this division should be functional and thus expect some benefit. Judging from the empirical literature, such a synergy may show up, under right circumstances, by speeding up skill learning, improving skill performance, and facilitating transfer of learned skills (Sun, Slusarz, & Terry, 2005; Sun, 2012).

There is some empirical evidence that can be interpreted in support of this prediction. In terms of speeding up learning, Willingham et al. (1989) found that those subjects in serial reaction time tasks who acquired more explicit knowledge appeared to learn faster. This suggested that their explicit procedural processes supplemented their implicit procedural processes. Stanley et al. (1989) reported that in a process control task, subjects' learning improved if they were asked to generate verbal instructions for others during learning. That is, an individual was able to speed up his or her own learning through an explication process that generated explicit knowledge (in addition to implicit knowledge). Sun et al. (2001) showed a similar effect of verbalization in a minefield navigation task and Reber and Allen (1978) in an artificial grammar learning task. Mathews et al. (1989) showed that a better result could be attained if a proper mix of implicit and explicit learning was used (in their case, first implicit learning and later explicit learning were encouraged).

Furthermore, in terms of skill performance, Stanley et al. (1989) found that subjects who verbalized while performing process control tasks were able to attain a higher level of performance than those who did not verbalize, likely because the requirement that they verbalized prompted the formation and utilization of explicit knowledge, which supplemented their implicit knowledge. Sun et al. (2001) also showed that verbalizing subjects were able to attain a

higher level of performance in a minefield navigation task. Squire and Frambach (1990) reported that initially amnesic and normal subjects performed comparably in a process control task and equally lacked explicit knowledge. However, with more training, normal subjects achieved better performance than amnesic subjects and also better scores on explicit knowledge measures, which pointed to the possibility that it was because normal subjects were able to learn better explicit knowledge that they achieved better performance. Consistent with this interpretation, Estes (1986) suggested that implicit learning alone could not attain optimal levels of performance. Even in high-level skill acquisition domains, similar effects were observed. Gick and Holyoak (1980) found that good problem solvers could better state explicit rules that described their actions in problem solving. Bower and King (1967) showed that verbalization improved performance in classification rule learning. Gagne and Smith (1962) showed the same effect of verbalization in learning to solve Tower of Hanoi.

In terms of facilitating transfer of learned skills, Willingham et al. (1989) provided some suggestive evidence that explicit knowledge facilitated transfer. They reported that (1) subjects who acquired explicit knowledge in a training task tended to have faster response times in a transfer task; (2) these subjects were also more likely to acquire explicit knowledge in the transfer task; and (3) subjects who acquired explicit knowledge responded more slowly when the transfer task was unrelated to the training task (suggesting that the explicit knowledge of the previous task might have interfered with the performance of the transfer task). Sun et al. (2001) showed some similar effects. In high-level domains, Ahlum-Heath and DiVesta (1986) found that the subjects who were required to verbalize while solving Tower of Hanoi performed better on a transfer task after training than those who were not required to verbalize.

Of course, synergy effects depend on contexts; they are not universal (Sun, 2002). Under some circumstances, explicit processes might even hurt learning and performance (Sun et al., 2001). Even so, it should be recognized that explicit processes serve important cognitive functions as discussed above. Explicit processes also serve additional functions, such as facilitating verbal communication, or acting as gatekeepers (e.g., enabling conscious veto, as suggested by Libet, 1985).

It has been demonstrated through simulation (e.g., by Sun, Slusarz, & Terry, 2005; Sun et al., 2007) that Clarion can computationally account for the synergy effects as described above, with the interaction between

implicit and explicit procedural processes within the ACS. The synergy effects may be discerned, in various tasks, through comparing different conditions, such as comparing the verbalization condition and the non-verbalization condition (whereby the verbalization condition encourages explicit processes), or comparing the dual-task condition and the single-task condition (whereby the dual-task condition discourages explicit processes). Simulations will be presented in chapters 5, 6, and 7 (see also Sun, Slusarz, & Terry, 2005).

2.5.2. Accounting for Synergy within the NACS

Just like the synergy effects between implicit and explicit procedural processes, the Clarion framework predicted that there should also be synergy effects between implicit and explicit declarative (non-action-centered) processes, resulting from their interaction (Sun and Zhang, 2006). Such synergy effects are expected from the ecological-functional perspective: Like other divisions of processes, knowledge, and memory, this division should be functional also (e.g., as argued in Sun, 2012).

Let us examine some implicit learning experiments (Reber, 1989). Recall that in process control experiments, synergistic results were found between implicit and explicit procedural processes (see the previous subsection). Domangue, Mathews, Sun, Roussel, and Guidry (2004) investigated the effects of similar training variables in artificial grammar learning, examining these effects in situations involving implicit and explicit declarative (as opposed to procedural) processes. The experiments in Domangue et al. (2004) tested implicit training with exemplars (encouraging implicit processes), explicit training with the grammar (encouraging explicit processes), and integrated training providing simultaneous experience with exemplars (encouraging implicit processes) and the grammar (encouraging explicit processes).

The results showed that encouraging explicit processing generally led to slower but more accurate responding on the cued-generate test. Encouraging implicit processing led to faster responding but with lower accuracy. In contrast, the integrated training achieved a balance, having higher accuracy than implicit training, and higher speed than explicit training.

One reasonable interpretation within the Clarion framework is that explicit training (in this particular context) led to the encoding of more grammatical knowledge in the form of explicit rules, while implicit

training led to the encoding of more implicit associative mappings. Neither implicit nor explicit knowledge in this task was action-centered—they likely resided in declarative memory (within the NACS). Often, both kinds of learning occurred, and the differences among different training conditions lay in the proportions of, and the interactions between, the two types of knowledge.

The experimental results above were simulated using Clarion, based on the interpretation above. The simulation correspondingly demonstrated the synergy effect of the integrated training, which achieved higher accuracy than implicit training and higher speed than explicit training. See Sun and Mathews (2005) for details.

There is also other corroborating evidence pointing to synergy between implicit and explicit declarative processes. For example, Berry (1983) showed that in a reasoning task, verbalization during learning improved transfer performance. The result appeared to indicate synergy between implicit and explicit declarative processes. Nokes and Ohlsson (2001) showed related results as well. This phenomenon may also be related, to some extent, to the self-explanation effect reported in the literature (e.g., Chi, Bassok, Lewis, Reimann, & Glaser, 1989): subjects who explained examples in physics textbooks more completely did better in solving new problems. There were also indications of synergy effects in alphabetic arithmetic tasks, categorical inference tasks, and so on. In all these cases, it could be the use of explicit declarative knowledge—in addition to the use of implicit declarative knowledge—that helped the performance. Clarion was used to computationally simulate data and phenomena in some of these tasks (see, e.g., Sun & Mathews, 2005; Sun & Zhang, 2006; Helie & Sun, 2010).

Table 2.3. Some basic constituting ideas of Clarion (see text for details).

- A distinction exists between implicit and explicit processes (though they tend to interact), with different forms of representation and learning.
- Knowledge is often redundantly represented in both explicit and implicit forms. Implicit and explicit processes are often simultaneously involved in a task. Results of explicit and implicit processing are often integrated (leading to synergy possibly).
- A distinction exists between procedural and declarative processes, in terms of their contents and forms (including learning), orthogonal to the distinction between implicit and explicit processes.
- These processes are motivationally driven and modulated. Motivations consist of drives and goals, with goals determined mostly based on drives.
- Metacognition regulates these processes in a number of specific ways based (in part) on motivations.

In addition, similarity-based reasoning can be naturally carried out through the interaction of implicit and explicit declarative processes. The interaction of the two types within the NACS of Clarion can account for important kinds of similarity-based processes in human everyday reasoning (see Sun, 1994, 2003, for details). Furthermore, the combination of similarity-based reasoning and rule-based reasoning (carried out within explicit declarative processes in Clarion) may capture and explain a wide range of common human everyday reasoning patterns (as shown by Sun, 1994, and Sun & Zhang, 2006). Thus, the separation and the interaction of these two types of processes are highly functional in the ecological-functional sense, and Clarion was able to capture and explain their effects.

2.6. Concluding Remarks

This chapter has presented some essential desiderata that have motivated the development of the Clarion framework and some arguments and evidence in support of these desiderata. Based on these desiderata, the basic framework of Clarion has been sketched, including its four major subsystems, before delving into more details in the next two chapters. Table 2.3 contains a summary of some basic constituting ideas of Clarion.

In the next two chapters, I will turn to more detailed descriptions and discussions of the four major subsystems. Chapter 3 will cover the action-centered and the non-action-centered subsystem—their major mechanisms and processes. Chapter 4 will cover the motivational and the metacognitive subsystem. After the detailed exposition of the subsystems of Clarion, chapters 5 and 6 will demonstrate how these subsystems work together to account for a variety of psychological data and phenomena.

3

The Action-Centered and Non-Action-Centered Subsystems

The preceding chapters point to, among other things, the need for the action-centered and non-action-centered subsystems within the Clarion framework, which I now explore in this chapter. (The preceding chapters also point to the need for other subsystems, which will be addressed in the next chapter.)

To account for action decision making as well as for some "executive control" functions, either implicit or explicit, there should be a subsystem for action-centered processes. This subsystem—the action-centered subsystem (the ACS)—stores procedural knowledge (in the procedural memory within the subsystem) for the sake of action decision making.

Separately, to account for reasoning of various sorts, explicit or implicit, there is the need for another subsystem. This subsystem—the non-action-centered subsystem (the NACS)—should account for a variety of reasoning phenomena. Thus this subsystem stores declarative knowledge (both semantic and episodic, in the declarative memory within the subsystem).

In this chapter, first the ACS is described (in Section 3.1). Then, a description of the NACS follows (in Section 3.2). In Section 3.3, learning

that goes across levels or subsystems is discussed, focusing especially on extracting explicit knowledge from implicit knowledge. Note that, as mentioned before, a forthcoming companion technical book will present full technical details (along with hands-on examples). Therefore, in this chapter and the next, I present only technical details essential to the ACS and the NACS, in what I hope is an easily understandable way, avoiding unnecessary or overcomplicated details.

3.1. The Action-Centered Subsystem

3.1.1. Background

The Action-Centered Subsystem (the ACS) captures procedural (i.e., action-centered) processes, knowledge, and memory. It is necessary, for example, for action decision making by an individual, either for everyday routine situations or for novel or difficult circumstances. It also includes some "executive control" functions. Because of the pervasiveness and the importance of these functions, a distinct subsystem (i.e., a somewhat standalone set of modules) is needed for carrying out the functions in an effective manner (recall the ecological-functional perspective referred to in Chapter 1). Hence the ACS exists within Clarion.

This subsystem is predicated on the distinction between, and the separation of, procedural and declarative processes (i.e., action-centered and non-action-centered processes), and consequently the distinction between the ACS and the NACS. The distinction between procedural and declarative processes has been discussed in Chapter 2, as well as in the literatures on human skill acquisition and on human memory. The arguments will not be repeated here; the reader should refer to Chapter 2 for more details (see also Sun, 2012).

The ACS is arguably the most important subsystem in Clarion. The ACS receives inputs from the external and internal environment and generates action decisions or commands. It thereby involves procedural knowledge from procedural memory. In addition to capturing procedural processes, the ACS captures some executive functions based on the same mechanisms and processes. For instance, it directs processes within the NACS.

As discussed earlier, each subsystem of Clarion, including the ACS, consists of two "levels" of representation (i.e., two sets of modules with

different representational forms). Generally, in each subsystem, the top level encodes explicit knowledge (explicit memory) and the bottom level encodes implicit knowledge (implicit memory). Together they constitute a dual representational structure. The distinction between implicit and explicit processes has been made in Chapter 2, based on voluminous data from empirical work on memory, learning, and a variety of other psychological functionalities (Reber, 1989; Seger, 1994; Sun, 2002; Evans and Frankish, 2009). The arguments will not be repeated here (however, in Chapter 5, I will further justify it based on simulations of empirical data). Within this framework, the top level of the ACS captures explicit action-centered (procedural) processes, while the bottom level of the ACS captures implicit action-centered (procedural) processes.

As discussed in Chapter 2, different representational forms (distributed versus symbolic-localist) are used in representing these two types of knowledge (implicit and explicit), in order to capture intrinsically the essential differences between these two types of knowledge through different representational forms. As I have argued before, the interaction between the two levels may lead to synergy in a variety of circumstances (Sun, Slusarz, & Terry, 2005). Therefore, such a division between the two levels is cognitively-psychologically advantageous.

In the bottom level of the ACS, implicit reactive routines reside, as mentioned in Chapter 2 (Tinbergen, 1951; Timberlake & Lucas, 1989). They are either innate or learned. In the top level of the ACS, explicit procedural knowledge exists. Such knowledge may be learned from information in the bottom level, information from external sources, and so on. Together, the two levels capture the interaction and the synergy between the two types of processes, as will be detailed later.

Figure 3.1 shows the division between the explicit and implicit processes within the ACS, as well as a similar division within the NACS (as a comparison). A technical description of the core processes of the ACS is provided below (Sun, 2002, 2003).

I should note here again the distinction between the conceptual-level Clarion theory and its current computational instantiations, as alluded to in Chapter 1. The conceptual-level theory enables multiple possible computational instantiations in many respects. For the sake of readability, I will not exhaustively cover all computational possibilities, although in some cases I will point out multiple possibilities when these possibilities are important. The reader should keep in mind that the current

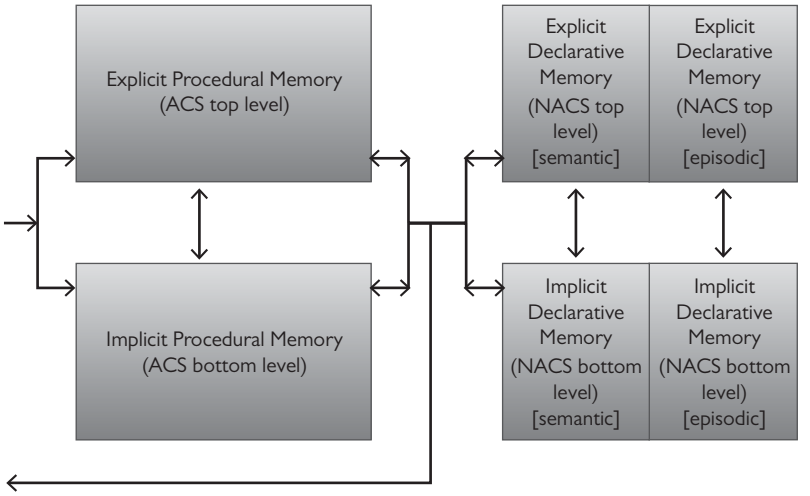

Figure 3.1. The essential memory modules. The leftmost lines show the input information to and output actions from the ACS. The lines between the modules show the information flows. (The working memory, goal structure, and sensory information store are used to facilitate the flows but are omitted above.)

computational implementations constitute only existence proofs of the conceptual-level theory and that other possibilities do exist.

3.1.2. Representation

3.1.2.1. Representation in the Top Level

In general, a basic unit for encoding knowledge is a "chunk," which corresponds to a concept (an object, a person, an event, and so on). A chunk is represented using both levels. At the top level, a chunk is represented by a unitary (localist) chunk node. At the bottom level, a chunk is represented by a distributed representation, which contains features or microfeatures (i.e., values for different dimensions of a concept; Rumelhart et al., 1986; Tsunoda et al., 2001). The distributed and localist representations together (along with their interaction) constitute a chunk (see Figure 3.2).[1]

1. The notion of "chunk" may be traced back to Miller (1956) and Miller et al. (1967). It was also used by Rosenbloom, Laird, and Newell (1993) and Anderson and Lebiere (1998). In general, a chunk is an aggregate of information. Each chunk may contain a

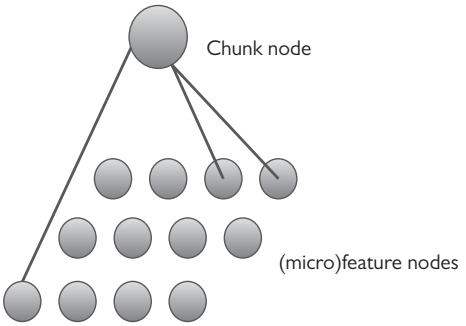

Figure 3.2. A chunk is represented by a chunk node (at the top level) connecting to a set of (micro)feature nodes (at the bottom level).

In the top level of the ACS, a *chunk node* represents either the condition or the action of a rule. Condition chunk nodes are activated by the environment (e.g., sensory inputs from the sensory information store) or other Clarion components (e.g., working memory that stores temporary information). Action chunk nodes represent external actions (e.g., motor programs) or internal actions (e.g., queries or commands to other Clarion components). Each chunk node is individually represented at the top level, has clear conceptual meaning, and constitutes a symbolic-localist representation.

Chunk nodes at the top level of the ACS can be linked to form *action rules* of the form: "*condition-chunk-node* → *action-chunk-node*" (see Figure 3.3).[2] Such a rule, in the simplest case, is captured by a weight (e.g., 1) on the link from a condition chunk node to an action chunk node. (Thus, action rules together constitute a linear, localist connectionist network; Sun, 1994, 2002.) For example, an action rule could be the following: "If there is a large obstacle in front, then turn left."

More specifically, assume that for the ACS the input (denoted as x) is made up of a number of dimensions (denoted as x_1, x_2, ..., x_n). Each dimension (x_i) can have a number of possible values, that is, features

"header" that indicates some key identifying information as well as a variable data section that contains "features". The notion of chuck as used here is different in a number of ways from that of Miller (1956), Rosenbloom, Laird, and Newell (1993), or Anderson and Lebiere (1998).

2. Alternatively, rules can be in the forms of "*condition* → *action new-state*" or "*condition action* → *new-state*". The encoding of the alternative forms of rules is similar to what is described here.

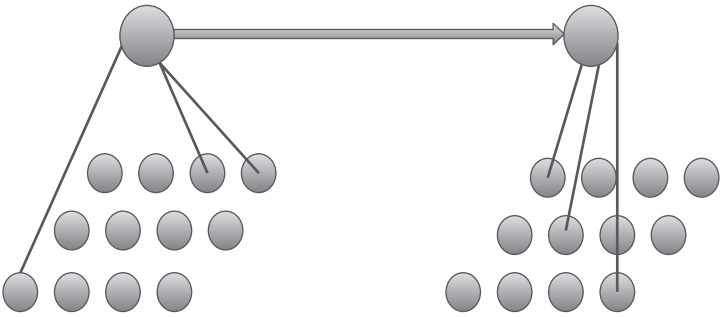

Figure 3.3. An action rule is formed, at the top level of the ACS, by connecting a condition chunk node to an action chunk node. These chunk nodes at the top level are also connected to their corresponding (micro)feature nodes at the bottom level.

or microfeatures (denoted as: v_{i1}, v_{i2}, …., v_{im}). [3] Due to the fact that a chunk node at the top level is connected to, and specified by, a set of (micro)features at the bottom level (i.e., dimensional values, represented as separate nodes at the bottom level), one should interpret an action rule specified at the top level as follows: the left-hand side of the rule, the condition chunk node, actually indicates a set of (micro)features; these (micro)features together constitute the condition for activating the rule. When they are satisfied by the input (in some way), the rule may be applied.[4] The right-hand side of the rule is an action chunk node (representing an action recommendation), which is likewise connected to the (micro)features at the bottom level.

So, the condition at the left-hand side of a rule is indicated by a chunk node at the top level (while each of its dimensional values is represented by a separate node at the bottom level). Similarly, the action at the right-hand side of a rule is also represented by a chunk node at the top level (while each of its dimensional values is represented by a separate node at the bottom level). The dimensional values at the bottom level

3. Each dimension is either ordinal (discrete or continuous) or nominal.

4. Through its connections to (micro)feature nodes at the bottom level, each condition chunk node specifies a conjunction ($\wedge$) of elements (each of which refers to one dimension). Each element within the conjunction specifies an allowable value range for a dimension (x_i) of the input (x), that is, $x_i \in (v_{i1}, v_{i2}, \ldots, v_{ik})$. Each element can thus be expressed as a disjunction: $(x_i\ v_{i1}) \vee (x_i\ v_{i2}) \vee \ldots \vee (x_i\ v_{ik})$. Each dimensional value $(x_i\ v_{ij})$ within the disjunction is represented as a (micro)feature node at the bottom level, which is connected to the chunk node at the top level.

serve as (micro)features of a chunk, and the chunk node at the top level serves to identify and label this set of dimensional values (features or microfeatures) as a whole.[5] The representations at the two levels together constitute a chunk.

3.1.2.2. Representation in the Bottom Level

The bottom level of the ACS captures implicit procedural knowledge. It uses distributed representation, with (micro)features. These (micro)features are connected by neural networks (Rumelhart et al., 1986).

The (micro)feature nodes (at the bottom level) and the chunk node (at the top level) can be activated together because they can be connected to each other. Joint activation can be accomplished through button-up activation flows (when the nodes at the bottom level are activated first, e.g., by sensory inputs) or top-down activation flows (when the chunk nodes at the top level are activated first). Therefore, as stated before, a chunk is represented at the two levels together—a localist chunk node at the top level and a distributed representation at the bottom level, both of which are part of a chunk. Moreover, the distributed and the localist representation are tied together by their close interaction.

For making action decisions implicitly, within the bottom level, (micro)features of conditions are mapped to (micro)features of actions based on several types of connectionist networks (feedforward or recurrent). In particular, a multilayer Perceptron network (an MLP network, also known as a Backpropagation network; Rumelhart et al., 1986) can be used, which decides on an action based on an input state.

The bottom level of the ACS is modular. There can be multiple action decision networks at the bottom level. Each network can be thought of as a behavioral routine or skill (innate or learned) that can be used to accomplish a particular type of task (Tinbergen, 1951; Timberlake and Lucas, 1989).

3.1.2.3. Action Decision Making

In the ACS, as touched upon before, action decision making is essentially as follows: Observing the current input state, the two levels of processes

5. In a chunk, when there are multiple allowable values in a dimension, the relation among them is logical OR (i.e., disjunction, denoted as $\vee$), because only one value needs to be present. The relation across dimensions is logical AND (i.e., conjunction, denoted as $\wedge$), because all of them should be present (at least ideally).

within the ACS (implicit and explicit) make their separate decisions in accordance with their respective knowledge (implicit or explicit) and their outcomes are integrated. Thus, a final selection of an action is made and the selected action is then performed. The action changes the world in some way. Comparing the changed input state with the previous input state, learning occurs. The cycle then repeats itself.

Thus, the overall algorithm for action decision making during an individual's interaction with the world is essentially as follows, assuming that inputs come in the form of (micro)features (dimensional values), which also go up to activate corresponding chunk nodes:

1. Observe the current input state x (including the current goal).
2. Compute in the bottom level the "value" of each of the possible actions (a_i's) within state x: $Q(x, a_1)$, $Q(x, a_2)$, …., $Q(x, a_n)$. Choose one action according to these values (when necessary; to be detailed later).
3. Find out all the possible actions (b_1, b_2, …., b_m) at the top level, based on the condition chunk nodes activated by the current input state (which comes up from the bottom level; more later) and the existing rules in place at the top level. Choose one action (when necessary).
4. Choose an action by selecting the action choice of either the top level or the bottom level, or by combining the values of actions from the two levels respectively and then selecting one action on that basis (to be detailed later).
5. Perform the action, and observe the next input state y and possibly the immediate reinforcement r.
6. Update knowledge at the bottom level in accordance with an appropriate learning algorithm (e.g., Q-learning, to be detailed later).
7. Update the top level using an appropriate learning algorithm (e.g., the RER algorithm, for extracting, refining, and deleting rules, to be detailed later).
8. Go back to Step 2.

Below, let us look into the details of these steps.

As mentioned before, in this subsystem, the bottom level consists of a number of modular neural networks involving distributed representation, and the top level contains explicit action rules with symbolic-localist representation (see Figure 3.1).

At the bottom level of the ACS, the input state (x) consists of three sets of information: the sensory inputs, the current goal (from the goal structure), and the working memory. The output of the bottom level is an action choice. The input and the output are represented as a set of (micro)features (dimensional values) at the bottom level. (Note that in general goals are important for action decision making.)

At the bottom level of the ACS, actions are selected based on their Q values. A Q value is an evaluation of the "quality" of an action in a given input state: $Q(x, a)$ indicates how desirable action a is in state x (which includes the current sensory input, the current goal, and the working memory). Given the current input state, an action is chosen based on Q values of actions (Luce, 1959).

Specifically, at each step, given input state x, the Q values of all the actions are computed: $Q(x, a)$ for all a's. To do so, for instance, an MLP (Backpropagation) network can be used (Rumelhart et al., 1986). In such a network, nodes are divided into multiple layers: the input layer, the hidden layer(s), and the output layer, with feedforward connections from one layer to the next. Each node, in whatever layer, computes its output (activation) with a sigmoidal function:

$$o = \frac{1}{1 + e^{-\sum_{i=0}^{n} w_i z_i}}$$

where z_i is the value of the i^{th} input to the node ($z_0 = 1$), w_i is the weight of the i^{th} input and n is the number of inputs to the node. This function is close to a threshold function (useful for binary decision making), but continuous and thus differentiable (useful for deriving learning algorithms; see the appendix). The nodes on the output layer generate Q values for actions.

Then, the Q values computed by the output layer of the network are used to decide stochastically on an action to be selected, through turning Q values into a probability distribution—a Boltzmann distribution of Q values:

$$p(a \mid x) = \frac{e^{Q(x,a)/\tau}}{\sum_i e^{Q(x,a_i)/\tau}}$$

where $p(a|x)$ is the probability of selecting action a given input state x, τ controls the "temperature" of the action decision making (degree of

randomness or stochasticity), and i ranges over all possible actions. An action is selected in accordance with the probabilities (i.e., $p(a|x)$ for all a's).[6]

At the top level of the ACS, action decision making is based on action rules in place there. As in the case of the input to the bottom level, the input to the top level consists of three groups of information: the sensory inputs, the current goal, and the working memory. The (micro)feature inputs to the bottom level activate, via the bottom-up activation flow, relevant condition chunk nodes at the top level, which in turn transmit activations to corresponding action chunk nodes via action rules, thus leading to action recommendations from the top level. In a given input state, one action can be chosen at the top level by choosing an applicable rule. All applicable rules compete to be used in action decision making.

A number of numerical measures are associated with each rule or each chunk node at the top level. First, the activation (i.e., *strength*) of a condition chunk node is determined from bottom-up activation, that is, from the activations of its constituting (micro)features (dimensional values) at the bottom level:

$$S^c_{c_k} = \sum_{i=1}^{n} A_i^{c_k} \times W_i^{c_k}$$

where $S^c_{c_k}$ is the strength (activation) of chunk c_k (where superscript c indicates that the measure is related to chunks), $A_i^{c_k}$ is the activation of the ith dimension of chunk c_k (which is the maximum of the activations within the dimension[7]), and $W_i^{c_k}$ is the weight of the ith dimension of chunk c_k ($1/n$ by default, where n is the number of dimensions of chunk c_k). The weights should sum to 1 or less. This weighted sum computation, roughly speaking, allows one to weigh different pieces of evidence in

6. This method turns a set of values into a probability distribution, with an added parameter ("temperature") that controls the degree of randomness of the distribution. When the temperature is low, the degree of randomness is low and the highest value tends to be selected. When the temperature becomes higher, the distribution becomes more random. This method is also known as Luce's choice axiom (Luce, 1959; Watkins 1989) and has been psychologically justified.

7. Bottom-up activation needs to take into consideration multiple allowable values in any dimension of a chunk. So, first, the activations of the multiple allowed values of a dimension are combined by taking the maximum (i.e., by using the function *max*), and then, across dimensions, the weighted sum specified above is applied to generate the activation of the chunk node.

determining the strength of a "conclusion," with a simple, linear combination of evidence.[8]

Then, from the strength of a condition chunk node, *rule support* is computed as follows:

$$s_k^r = s_{c_k}^c \times w_k^r$$

where s_k^r is the support for rule k (where k indicates a rule at the top level), $s_{c_k}^c$ is the strength of condition chunk c_k (representing the condition of rule k), and w_k^r is the weight of rule k (where the default is 1). Superscript r indicates that the corresponding measures are related to rules.

To stochastically select an action rule to apply (in order to generate an action recommendation at the top level), a Boltzmann distribution is constructed from the rule support values for all rules (similar to the stochastic selection at the bottom level):

$$p(j|x) = \frac{e^{s_j^r/\tau}}{\sum_i e^{s_i^r/\tau}}$$

where $p(j|x)$ is the probability of selecting rule j given input state x, s_i^r is the rule support for rule i, τ is the "temperature", and i ranges over all applicable rules.

Another numerical measure associated with action rules is *rule utility*, which measures the effectiveness of a rule, in terms of cost and benefit. Utility may be determined on the fly. The utility for rule j may be determined by:

$$U_j^r = benefit_j - v \times cost_j$$

where v is a scaling factor balancing measurements of *benefit* and *cost*.[9]

At the top level, to stochastically select an action rule to apply, a Boltzmann distribution is constructed either from the rule support values

8. The weighted sum computation has had a long history in cognitive modeling. For example, early neural network models of the 1950s used it. Sun (1994) explored its logical and other interpretations and implications, which led directly to its present use.

9. Within this formula, one may define:

$$benefit_j = \frac{c_7 + PM(j)}{c_8 + PM(j) + NM(j)}$$

(as discussed before) or from the rule utility values. In the latter case, we have:

$$p(j \mid x) = \frac{e^{U_j/\tau}}{\sum_i e^{U_i/\tau}}$$

where $p(j|x)$ is the probability of selecting rule j given input state x, U_i is the utility for rule i, τ is the "temperature," and i ranges over all applicable rules.

Yet another numerical measure is *base-level activation* (BLA; Anderson, 1993), for capturing certain priming effects (Tulving, 1985). For example, for an action rule, its BLA is determined by:

$$b_j^r = ib_j^r + c\sum_{l=1}^{n} t_l^{-d}$$

where b_j^r is the base-level activation of rule j, ib_j^r is the initial base-level activation of rule j (by default, $ib_j^r = 0$), c is the amplitude (by default, $c = 2$), d is the decay rate (by default, $d = 0.5$), and t_l is the time (e.g., in *ms*) since the lth use (or creation) of the rule. Superscript r above indicates that the measures are related to rules.

This measure has an exponential decay and corresponds to the odds of needing rule j based on past experiences (Anderson, 1993). When the base-level activation of a rule falls below a "density" parameter (d_r), the rule is no longer available for use, thus capturing forgetting. Likewise, each chunk node has a similar base-level activation (b_j^c, defined by a similar equation) and a corresponding "density" parameter (d_c).

Finally, analogous to the strength (activation) of a condition chunk node, each conclusion (action) chunk node has its strength (activation) too. The *strength* of an action chunk node is determined from taking the maximum of multiple measures of rule support pointing to the same

where j indicates a particular rule, $PM(j)$ = number of positive matches for j, $NM(j)$ = number of negative matches for j, both defined in Section 3, and by default $c_7 = 1$, $c_8 = 2$;

$$cost_j = \frac{execution\_time\_of\_rule\_j}{average\_execution\_time\_of\_rules}$$

where the two execution times need to be specified (either as fixed constants or as functions that are computed on the fly).

action chunk node (i.e., using the function *max*), if there are multiple action rules reaching the same conclusion.[10]

In addition to external actions (e.g., actions for physical movements), there are also internal actions available to the ACS, which may be used, for example, for setting or resetting goals, for managing working memory (storing or removing contents), for directing the operation of the NACS, and so on. In particular, actions from the ACS can query or command the NACS (Section 3.2). In this case, the NACS returns, via working memory, one or more chunks resulting from reasoning within the NACS, which are then used by the ACS in determining action recommendations.

3.1.3. Learning

3.1.3.1. Learning in the Bottom Level

First, there is the learning of implicit action-centered knowledge at the bottom level of the ACS, that is, the learning of implicit reactive routines. Such learning may be accomplished through trial-and-error interaction with the world, in keeping with the ecological-functional considerations and the desiderata. Cleeremans (1997) argued that implicit learning could not be captured by symbolic models but neural networks. Sun (1999) made arguments regarding distributed representation and incremental nature of implicit learning.

As mentioned before, one way of implementing a mapping going from input states to action choices in order to capture implicit action-centered knowledge is to use a multilayer neural network (e.g., a Backpropagation or MLP network). To improve the mapping, that is, to learn implicit action-centered knowledge, adjustment of parameters (e.g., weights) may be carried out incrementally, in ways consistent with the nature of distributed representation.

Reinforcement learning algorithms, as developed by machine learning and operations research (e.g., Bertsekas & Tsitsiklis, 1996; Sutton & Barto, 1998), are suitable for tuning neural networks, that is, for enabling learning of implicit action-centered knowledge. With reinforcement learning algorithms, learning at the bottom level is incremental, gradual,

10. Strengths of action chunks are needed, if a weighted sum is used for cross-level integration, in which case values of actions from the two levels are combined (see the subsection on level integration for more details).

and iterative. The sequential nature of implicit reactive routines is captured through reinforcement learning algorithms with temporal credit assignment mechanisms (such as the Q-learning algorithm; Watkins, 1989). Sequential behavior may be accomplished on the basis of action decision making using moment-to-moment information. Implicit reactive routines therefore exhibit sequential behavior without explicit planning (Sun 2002). Such a learning approach has been justified from the ecological-functional perspective in Chapter 2.

As described before, in the bottom level of the ACS, a Q value is an evaluation of an action in a given state. To acquire the Q values, the Q-learning algorithm, for example, may be applied. Within this algorithm, *Q(x, a)* estimates the maximum (discounted) cumulative reinforcement that one can receive from the current state x onward after action a is performed. The updating of *Q(x, a)* is based on the temporal difference in evaluating the state and the action, which enables the learning of Q values that approximate the (discounted) cumulative reinforcement (Bertsekas and Tsitsiklis, 1996). Through successive updates of the Q values, one can learn to take into account future steps in longer and longer sequences (using only moment-to-moment information; Watkins, 1989).

The basic form of Q-learning is as follows:

$$\Delta Q(x,a) = \alpha(r + \gamma e(y) - Q(x,a))$$

where $\Delta Q(x,a)$ is the adjustment to $Q(x,a)$, x is the current state, a is the current action, α is the learning rate, r is the immediate reinforcement received right after action a in state x, *e(y)* is the maximum Q value in the next state y, γ is a discount factor, and $r + \gamma e(y)$ estimates the (discounted) total reinforcement that can be received from the current state and action onward. That is, the adjustment to Q values is based on the difference between two successive estimates of Q values (before and after an action is performed).

Note that reinforcement (feedback) received by the ACS (e.g., r above) may be determined physically, biologically, and/or socioculturally in terms of what constitutes reinforcement and when it is received (through the MS and the MCS, to be discussed in Chapter 4). Therefore, implicit reactive routines acquired through reinforcement learning may be sociocultural to some extent, in addition to being oriented towards the physical world. Details of reinforcement will be discussed in Chapter 4.

To implement Q-learning, a four-layered Backpropagation (MLP) network may be used. The network is internally subsymbolic with distributed representation in the hidden layer. The outputs of the third layer indicate the Q values of all actions (with each action represented by an individual node), and the node in the fourth layer determines the action to be performed based on a distribution of Q values (e.g., a Boltzmann distribution).

The error measure on which learning in the Backpropagation (MLP) network is based is:

$$err_i = \begin{cases} r + \gamma e(y) - Q(x, a_i) & \text{if } a_i = a \\ 0 & \text{otherwise} \end{cases}$$

where err_i is the error of node i on the third layer, a_i is the action corresponding to node i, *and* a is the action just performed. In order to gradually minimize errors, weight updating in this network is done iteratively using the Backpropagation learning algorithm based on the error measure above (see the appendix for more details; see also Rumelhart et al., 1986; Levine, 2000).

Supervised learning using the Backpropagation learning algorithm only (that is, with an error measure that is the difference between the actual output and the target output, without using an error measure derived from reinforcement learning algorithms such as the one above) is also possible when appropriate. For example, supervised learning is appropriate when there is a direct indication of the correct output (target), in which case an error is calculated directly as the difference between the actual output and the target. This may capture instructed learning, in which case learning is directly sociocultural in nature (Sun, 2001). In a similar way, in the error measure implementing Q-learning above, $\gamma e(y)$ might be omitted (when future Q values are irrelevant), which leads to what was termed "simplified Q-learning" (Sun, 2003), basically the same as supervised learning.

3.1.3.2. Learning in the Top Level

At the top level of the ACS, explicit knowledge is captured in the form of action rules, coded with condition and action chunk nodes. Explicit knowledge at the top level can be learned in a variety of ways, in accordance with symbolic-localist representation used there. Because of its representational and other characteristics, one-shot learning is appropriate.

With such learning, one dynamically acquires rules and modifies them as needed, in keeping with the ecological-functional perspective (Sun, 2002).

As mentioned before, implicit knowledge existing at the bottom level can be utilized in learning explicit knowledge at the top level, through bottom-up learning (e.g., the *Rule Extraction and Refinement* or RER algorithm; Sun et al., 2001). That is, implicit knowledge accumulated in neural networks can be used for establishing and then refining explicit rules. This is a kind of "rational reconstruction" of implicit knowledge.

Other forms of learning explicit knowledge are also possible. For example, explicit knowledge may be established using explicit hypothesis testing without the help of the bottom level initially. However, subsequently, explicit hypotheses may be tested with the help of information from the bottom level. In that case, learning is also bottom-up to a certain extent (e.g., the *Independent Rule Learning* or IRL algorithm; Sun et al., 2005).

Furthermore, explicit knowledge can be established at the top level of the ACS, for example, through externally provided information. Knowledge acquired from external sources may be coded at the top level using chunk nodes and action rules. It may also be coded using more complex "fixed rules" (FRs), when more complex forms are required (Sun, 2003). Such learning may be sociocultural in nature.

Once explicit knowledge is established at the top level (e.g., through externally provided information), it can be assimilated into the bottom level. This often occurs during the novice-to-expert transition in instructed learning settings (Dreyfus & Dreyfus, 1987; Anderson & Lebiere, 1998). The assimilation process, termed top-down learning (as opposed to bottom-up learning), can be carried out using the same implicit learning mechanisms sketched earlier.

So, there are a variety of ways in which explicit action-centered knowledge is learned in Clarion, as in humans. Explicit action rules, and the chunk nodes involved in encoding these rules, can be learned in the following ways:

1. in a bottom-up way (using the *Rule Extraction and Refinement* algorithm, or RER, to be detailed in Section 3.3)

2. by hypothesis testing more independently (using the *Independent Rule Learning* algorithm, or IRL, to be detailed in Section 3.3), or
3. from external sources (which might lead to "fixed rules" or FRs, as mentioned before; Sun, 2003)

In all these cases, rules and chunk nodes are learned in a "one-shot" fashion, even though information (e.g., statistical information) used in learning may accumulate gradually. I will defer the full discussion of these learning methods to Section 3.3, where various forms of knowledge extraction, assimilation, and transfer will be discussed together.

Learning serves the needs of an individual in interacting with the world, and particularly when coping with everyday activities. Learning within the ACS tunes the decision-making mechanisms for better action decision making in such activities to better meet the needs of an individual. Learning as described above should be seen in this light, in accordance with the ecological-functional perspective (Sun, 2002).

3.1.4. Level Integration

For level integration, that is, for integrating the action recommendations from the two levels of the ACS, several methods exist, including *stochastic selection* and *combination*.

First, look into the method of *stochastic selection*. Suppose that there are the following components within the ACS: the RER rule set, the IRL rule set, the FR rule set, and the networks at the bottom level (assuming they all finished processing within a time limit if a time limit exists; see Appendix A.1). Using this method, at each step, the probability of using any given rule set is determined: P_{RER} (probability of using the RER rule set), P_{IRL} (probability of using the IRL rule set), or P_{FR} (probability of using the FR rule set). The probability of using the bottom level is $P_{BL} = 1 - P_{RER} - P_{IRL} - P_{FR}$. These selection probabilities can be either fixed (pre-set by the metacognitive subsystem) or variable (calculated on the fly by the metacognitive subsystem). Then, it is just a matter of selecting a component stochastically using the probabilities above. Note that deterministic selection of the bottom level (or any other component) is a special case of stochastic selection (where probabilities are 1 or 0).

Variable selection probabilities are calculated using the notion of "probability matching" (e.g., Lopez and Shanks, 2008), as follows:

$$P_{BL} = \frac{\beta_{BL} \times sr_{BL}}{\Phi}$$

$$P_{RER} = \frac{\beta_{RER} \times sr_{RER}}{\Phi}$$

$$P_{IRL} = \frac{\beta_{IRL} \times sr_{IRL}}{\Phi}$$

$$P_{FR} = \frac{\beta_{FR} \times sr_{FR}}{\Phi}$$

where *sr* stands for success rate (roughly percentage of positive matches, where a positive match for a component occurs if the state and the actual action performed match the decision by the component and the result is positive, as discussed in Section 3.3), β is a weighting parameter, and $\Phi = \beta_{BL} \times sr_{BL} + \beta_{RER} \times sr_{RER} + \beta_{IRL} \times sr_{IRL} + \beta_{FR} \times sr_{FR}$. That is, the probability of selecting a component is determined based on the relative success rate of that component (calculated by the metacognitive subsystem; see Chapter 4).

Second, turn to the method of *combination*. With this method, *bottom-up verification* can be done whereby outcomes from the bottom level are sent to the top level, which then rectifies the outcomes using its explicit knowledge. This is likely to happen in reasoned action decision making where final outcomes are explicit. Alternatively, *top-down guidance* can occur whereby outcomes of the top level (from rule sets) are sent down to the bottom level, which then takes them into consideration along with its own implicit knowledge in making action decisions. This is likely to happen in fluid skill performance.

In both cases, the simplest implementation is a weighted sum of the corresponding values for actions across the two levels (assuming both levels finish processing within a time limit if a time limit exists; see Appendix A.1) and then stochastic selection based on a Boltzmann distribution of the combined values (using the same equation as before). The weights of different components must be specified, which can also be either fixed or variable (same as discussed earlier).

3.1.5. An Example

Below is a simple example that briefly illustrates the working of the ACS.

A serial reaction time (SRT) task is used as an example here (Curran & Keele, 1993). In this task, subjects were presented a repeating sequence of X marks, each in one of four possible positions. The subjects were asked to press, as quickly as possible, the button that corresponded to the position in which an X mark appeared as soon as an X mark appeared. Subjects might learn to predict new positions on the basis of preceding positions, although usually not consciously. That is, they might learn the sequential dependency relations embedded in the sequence, which might lead to faster responding.

Let us see how this might work in Clarion. Learning at the bottom level of the ACS proceeds as described before: it amounts to iterative weight updating of a neural network in the bottom level. Such learning promotes implicit knowledge formation, which is embedded in the weights of the neural network. The resulting weights in fact specify a function relating previous positions (the input) to the prediction of the current position (the output). The prediction can then be used to preposition the hand for pressing a button. If the prediction is correct, it leads to faster responses. Over time, responses become faster as a result.

Implicit knowledge acquired at the bottom level of the ACS can also lead to the extraction of explicit knowledge at the top level of the ACS. As will be detailed in Section 3.3, an initial extraction step may create an explicit rule that corresponds to the input and the output determined by the bottom level. The rule may be used to direct actions (in conjunction with the bottom level, through level integration as described earlier). Later on, generalization may make the rule more generic, having more chances of matching inputs, while specialization may make the rule narrower in scope. As will be detailed in Section 3.3, these operations on explicit rules are guided by statistical information from the bottom level (following the idea of bottom-up learning). Other ways of learning explicit knowledge are also possible.

3.2. The Non-Action-Centered Subsystem

3.2.1. Background

The Non-Action-Centered Subsystem (the NACS) captures declarative processes, involving declarative knowledge in declarative memory (in semantic or episodic memory). The NACS captures various forms of

reasoning (e.g., Helie & Sun, 2010; Sun & Zhang, 2006). The inputs and outputs of this subsystem usually come from or go to other subsystems, in particular, the ACS.

To justify the existence of this subsystem, recall that that the distinction between procedural and declarative processes and its orthogonality with the implicit-explicit distinction have been argued for in Chapter 2 (Sun, 2012). It is therefore reasonable to posit the separate existence of the NACS, for the sake of capturing declarative processes, separate from procedural processes. It is also reasonable to posit the division between the implicit and the explicit level within the NACS, for the sake of capturing implicit and explicit declarative processes, respectively.

To justify the NACS, we also need to address the distinction between episodic and semantic memory within declarative processes, that is, within the NACS, as well as its orthogonality with the implicit-explicit distinction.

First, look into the distinction between episodic and semantic memory. Quillian (1968) originally proposed the idea of semantic memory for the sake of organizing information for semantic processing. However, this notion has been generalized to include all general knowledge that is not directly related to specific past experiences (i.e., not episodic in nature) and not action-centered (i.e., not procedural). Tulving (1972, 1983), for example, expounded on the difference between semantic and episodic memory. Roger (2008) and Norman et al. (2008) discussed computational models of semantic and episodic memory respectively.

In line with the ecological-functional perspective, Klein et al. (2002) pointed out that episodic and semantic memory evolved to solve different problems. However, while some tasks may require information from episodic or semantic memory alone, other tasks may require information from both. Dissociations may not be absolute; one may find independence for some tasks and dependence for others. The extent of functional independence may reflect the informational requirements of an individual (Klein et al., 2002). The division of episodic and semantic memory should thus be functional.

Next, toward establishing the orthogonality of the implicit-explicit distinction and the semantic-episodic distinction, look into the distinction between implicit and explicit semantic memory. On the one hand, explicit semantic memory is well established. Ever since Quillian (1968), semantic memory has been largely portrayed as explicit and conceptual, consisting of nodes representing explicit concepts and links representing

explicit conceptual relations among them. Collins and Loftus (1975), for instance, advocated such a view. On the other hand, what distinguishes implicit semantic memory from explicit semantic memory is that the former involves implicit connections among memory contents whereby these connections are outside of conscious awareness.

One notion important for implicit semantic memory is priming. Here we consider priming in the form of a mechanism that facilitates the identification of the same (or related) objects seen before on a later occasion, in the sense that the identification requires less information or occurs more quickly (Tulving, 1985; Nelson et al., 1998). Such priming often occurs in the absence of conscious awareness, as shown by empirical data from implicit memory research (see, e.g., Toth et al., 1994; Roediger, 1990). Moreover, Tulving and Schacter (1990) suggested that conceptual priming (including conceptual priming without conscious awareness) involved semantic memory. Implicit semantic memory is thus justified.

It is reasonable to hypothesize that implicit semantic memory is separate in some way from its explicit counterpart, on the basis of many kinds of dissociations, which suggested the possibility of separate memory stores (Dunn and Kirsner, 1988). Other arguments presented in Chapter 2 for the separation of implicit and explicit memory are also applicable here; the reader is referred to them as well. Moreover, in Schacter's (1987) memory model, there were separate implicit memory stores, some of which were semantic. As added support for this view, Sun and Zhang (2006) showed how the division of implicit and explicit semantic memory accounted for categorical inferences where similarity-based processes played a significant role; Helie and Sun (2010) showed how the division of implicit and explicit semantic memory also accounted for creative problem solving.

To establish the orthogonality of the explicit-implicit distinction and the episodic-semantic distinction, the distinction between implicit and explicit episodic memory also needs to be addressed. Explicit episodic memory is well established (Tulving, 1983). It stores information concerning actual prior experiences in an explicit and individuated form, including spatial and temporal information about events and activities. It constitutes an explicit personal memory ("self-referential memory"; Tulving, 1983). On the other hand, implicit episodic memory is a derived memory in the sense defined by Klein et al. (2002), which is formed through transforming available information in a way that enables speedy supply of information by reducing further processing, so that

prompt actions can be taken in relevant circumstances. It keeps track of statistics abstracted from actual experiences stored in explicit episodic memory (Hasher & Zacks, 1979).[11] Some may argue that implicit episodic memory should be categorized as a semantic memory, but such a debate would not be very useful because it would merely be about a label. For example, what was termed "semantic trait memory" by Klein et al. (2002) is an implicit episodic memory according to the Clarion framework. In general, implicit episodic memory can be understood in this light.[12]

Therefore, the overall structure of the NACS is as depicted in Figure 3.1.

3.2.2. Representation

The NACS captures declarative memory (both semantic and episodic). On that basis, the NACS also captures various forms of reasoning. Below, a brief description of the essential mechanisms and processes of the NACS is given (see also Sun, 2002, 2003).

3.2.2.1. Overall Algorithm

Look into semantic memory of the NACS. Assume inputs in the form of chunk nodes at the top level of the NACS being activated.[13] First, top-down activation occurs to activate corresponding (micro)feature nodes at the bottom level of the NACS. Then, within-level processing occurs in accordance with the structure and content of each level. Finally, bottom-up activation leads to integrating the outcomes of the two levels. That is, the NACS performs the following steps:

1. Observe the input information (from outside sources or from the previous iteration).

11. It includes what I called "abstract episodic memory," which stores frequencies of transitions, for example, from any state and action to any new state. Abstract episodic memory is implicit (Hasher & Zacks, 1979) and thus resides in the bottom level.

12. In line with the ecological-functional perspective, according to Klein et al (2002), derived memory is formed on the basis of predictability, importance, urgency, and economy. Precomputed summaries in implicit episodic memory reduce online computation and speed up retrieval, but they require additional representation. Human memory may have evolved to address such trade-offs. The division of implicit and explicit episodic memory is functional in this sense.

13. Alternatively, inputs may be transformed into such a form.

2. Perform top-down activation: activated chunk nodes at the top level activate their corresponding (micro)feature nodes at the bottom level.
3. Simultaneously process the information at both levels.
4. Perform bottom-up activation, and calculate the integrated activations of chunk nodes.
5. Stochastically choose a response chunk node (from a Boltzmann distribution of the integrated activations). If its internal confidence level (ICL)[14] is higher than a preset threshold (ψ), then output the response. Otherwise, treat the current integrated activations as the new input and go back to step 2.[15] (Alternatively, all activated chunk nodes are output as long as their ICLs are above a threshold.)

3.2.2.2. Representation in the Top Level

In the top level of the NACS, explicit knowledge is represented by chunk nodes, same as in the top level of the ACS. However, unlike in the ACS, chunk nodes in the NACS are not divided into condition and action chunk nodes. Each chunk node represents a concept that can be used either as a condition or a conclusion in an associative rule.

Chunk nodes in the NACS are linked to form *associative rules*. Unlike action rules in the ACS, the condition of an associative rule can contain multiple chunk nodes. See Figure 3.4. In the simplest case, representing associative rules using connection weights, a weighted sum is used to calculate the activation (strength) of a conclusion chunk node from rule application:

$$s_j^r = \sum_{i=1}^{n} s_i \times w_{ij}^r$$

where s_j^r is the activation (strength) of conclusion chunk node j from the application of an associative rule, s_i is the activation of condition chunk node i, i ranges from 1 to n (where n is the number of chunk nodes in the condition of the associative rule), and w_{ij}^r is the weight from condition chunk

14. The ICL may be the integrated activation of the chosen chunk node or a normalized version based on a Boltzmann distribution.

15. This happens only when there is time remaining (which is determined by comparing a time limit and the response time computed on the fly; see Appendix A.1),

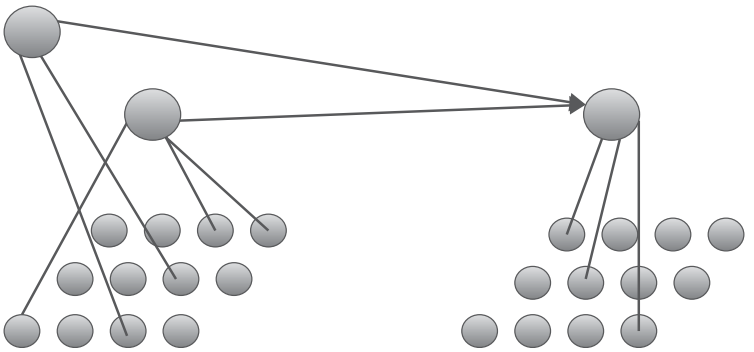

Figure 3.4. An associative rule is formed, at the top level of the NACS, by connecting the condition chunk nodes to the conclusion chunk node. These chunk nodes at the top level are also connected to their corresponding (micro)feature nodes at the bottom level.

node i to conclusion chunk node j (by default, $w_{ij}^r = 1/n$; these weights should sum to 1 or less). Superscript r above indicates that these measures are related to rules.[16] The application of this equation is referred to as *rule-based reasoning* within the NACS. The top level of the NACS thus amounts to a linear connectionist network (Sun, 1994).[17]

Chunks in the NACS are related by similarity. The activation of a chunk node caused by the similarity of other chunks to this chunk (represented by the chunk node) is termed *similarity-based reasoning* (SBR). Specifically,

$$s_j^s = s_{c_i \sim c_j} \times s_i$$

where s_j^s is the activation (strength) of chunk node j due to similarity (where superscript s indicates that the measure is similarity-related), $s_{c_i \sim c_j}$ is the similarity from chunk i to chunk j, and s_i is the activation of chunk node i.[18]

In the equation above, as the default, $s_{c_i \sim c_j}$ is calculated simply as follows (cf. Tversky, 1977; Sun, 1994, 1995):

16. All rules fire in parallel in the NACS. As such, a chunk node can receive activations from more than one associative rule. In that case, the *maximum* of these rule-based activations is used (by default).

17. The weighted sum has been used in neural network models since their earliest days. Sun (1994) investigated symbolic processing in neural networks and provided a logical interpretation of this computation. Its use here is thereby justified.

18. A chunk node can simultaneously receive activations from more than one similarity matching. The *maximum* of them is used (by default).

$$s_{c_i \sim c_j} = \frac{n_{c_i \cap c_j}}{f\left(n_{c_j}\right)}$$

where, by default, $n_{c_i \cap c_j}$ is the number of overlapping (micro)features between chunk i and chunk j, n_{c_j} is the number of (micro)features that chunk j has, and $f(x) = x^{1.1}$ (to make it superlinear; Sun, 1994). It is essentially the percentage of overlapping (micro)features with respect to the target concept.

More generally, however, we may define them as follows:

$$n_{c_i \cap c_j} = \sum_k v_k^{c_j} h_k\left(c_i, c_j\right)$$

and

$$n_{c_j} = \sum_k v_k^{c_j}$$

Therefore, $s_{c_i \sim c_j}$ becomes the following:

$$\begin{aligned} s_{c_i \sim c_j} &= \frac{n_{c_i \cap c_j}}{f\left(n_{c_j}\right)} \\ &= \frac{\sum_k v_k^{c_j} h_k\left(c_i, c_j\right)}{f\left(\sum_k v_k^{c_j}\right)} \end{aligned}$$

where $v_k^{c_j}$ is the intensity of connection of chunk j to (micro)feature k (i.e., top-down weights; by default, $v_k^{c_j} = 1$ for all k's), $h_k(c_i, c_j) = 1$ if chunk i and chunk j share (micro)feature k and $h_k(c_i, c_j) = 0$ otherwise. The function f is a slightly superlinear, positive, monotonically increasing function (Sun, 1994, 2003).

In the general definition above, with the default weights (i.e., 1), $n_{c_i \cap c_j}$ counts the number of (micro)features shared by chunks i and j, and n_{c_j} counts the number of (micro)features in chunk j. The similarity measure basically amounts to the percentage of overlapping (micro)features with respect to the target concept, as in the simpler version. However, more generally, weights can vary from their default values to account for prior knowledge or the context.

Similarity-based reasoning and similarity measure $s_{c_i \sim c_j}$ are accomplished through the interaction of the two levels of the NACS, involving both top-down and bottom-up activation flows, without any additional mechanisms (as will be detailed later).

A chunk node in the NACS may also be activated by a command or query from the ACS. When a chunk node in the NACS is activated by the ACS, its activation is set to full activation (by default): that is, $s_j^{ACS} = 1$, where j indicates a chunk node in the NACS. However, the other two sources of activation can have smaller positive values.

Therefore, summarizing the discussion thus far, a chunk node at the top level of the NACS can be activated by

- its association with another chunk node via an associative rule
- its similarity relation with another chunk via similarity-based reasoning
- an ACS query/command

Overall, the activation (strength) of a chunk node in the top level of the NACS is equal to the maximum activation that it receives from the three aforementioned sources, modulated by their respective weights:

$$s_j = max(\beta_0 \times s_j^{ACS},\ \beta_1 \times s_j^r,\ \beta_2 \times s_j^s)$$

where s_j is the overall activation (strength) of chunk node j, and β_0, β_1, and β_2 are scaling (balancing) parameters. (By default, $\beta_0 = \beta_1 = \beta_2 = 1$, although in this way rule-based reasoning may overwhelm similarity-based reasoning because SBR usually results in lower activation.)

Chunks that are inferred (activated) in the NACS can be sent to the ACS for its consideration in its action decision making (via chunk nodes through working memory). If only one chunk is to be returned to the ACS, a chunk is selected and returned. A chunk may be selected stochastically by transforming chunk node activations into chunk retrieval probabilities through a Boltzmann distribution:

$$P(j) = \frac{e^{s_j/\tau}}{\sum_i e^{s_i/\tau}}$$

where $P(j)$ is the probability that chunk j is chosen to be returned to the ACS, s_i is the activation (strength) of chunk node i, and τ is the temperature (degree of randomness). The chunk that is sent back to the ACS is accompanied by an internal confidence level (ICL), which may be the activation of the chunk node or its normalized activation (the Boltzmann probability described above). If all activated chunks are to be returned to the ACS, no stochastic selection is necessary, but the internal confidence level is calculated the same way as above.

In addition to the aforementioned activation (strength) of a chunk node, each chunk node has a base-level activation (Anderson, 1993), for capturing certain priming effects (the same as in the ACS):

$$b_j^c = ib_j^c + c\sum_{l=1}^{n} t_l^{-d}$$

where b_j^c is the base-level activation of chunk node j, ib_j^c is the initial base-level activation of chunk node j (by default, $ib_j^c = 0$), c is the amplitude (by default, $c = 2$), d is the decay rate (by default, $d = 0.5$), and t_l is the time since the lth use (or creation) of the chunk node. Superscript c above indicates that the measures are related to chunks. When the base-level activation of a chunk node falls below a "density" parameter (d_c), the chunk is no longer available for retrieval or for use in reasoning (rule-based or similarity-based), capturing forgetting. Each associative rule in the NACS has a similar base-level activation (b_j^r) and a corresponding density parameter (d_r).

3.2.2.3. Representation in the Bottom Level

The bottom level of the NACS involves distributed representation with (micro)features that encode chunks (which are also encoded by chunk nodes at the top level; Sun, 2003), the same as in the bottom level of the ACS. All the (micro)feature nodes of a chunk at the bottom level are connected to the corresponding chunk node at the top level so that, when a chunk node is activated, its corresponding (micro)feature nodes are also activated, and vice versa. Here I will focus on the bottom level of semantic memory within the NACS.

The top-down activation works this way: An activated chunk node at the top level of the NACS activates all its (micro)feature nodes at the bottom level. By default, top-down weights from the chunk node to its (micro)feature nodes are 1. So those (micro)feature nodes are activated to the same extent as the chunk node (with the same strength level). More generally, however, top-down weights can vary (e.g., between 0 and 1) to put different emphase on different (micro)features. It is denoted as $v_k^{c_j}$—the top-down weight from chunk j to its (micro)feature k.[19]

19. Multiple allowable values in a dimension need to be addressed. Top-down activation within the NACS first determines the activation for each dimension of the chunk at the bottom level (using top-down weights mentioned above) and then, within each dimension,

On the other hand, the bottom-up activation uses the following equation:

$$S_j^s = \sum_k \frac{v_k^{c_j}}{f\left(\sum_l v_l^{c_j}\right)} \times A_k^{c_j}$$

where S_j^s is the activation of chunk node j resulting from the bottom-up activation by its (micro)features (i.e., from similarity-based reasoning as mentioned before; hence superscript s), $A_k^{c_j}$ is the activation of its k^{th} (micro)feature node, $v_k^{c_j}$ is the top-down weight from chunk node j to its (micro)feature node k, and $\frac{v_k^{c_j}}{f\left(\sum_l v_l^{c_j}\right)}$ is the bottom-up weight from (micro)feature node k to chunk node j. The bottom-up weights are thus essentially a normalized version of the corresponding top-down weights, for the sake of accomplishing similarity computation defined earlier.[20]

Activation flows between chunk nodes and their corresponding (micro)feature nodes allow for a natural computation of similarity. The similarity measure defined previously is naturally accomplished using top-down and bottom-up activation flows, without a need for any dedicated similarity-based reasoning mechanism. Similarity-based reasoning is accomplished using

1. top-down activation by chunk nodes of their corresponding (micro)feature nodes
2. (micro)feature overlapping between any two chunks
3. bottom-up activation of all related chunk nodes

One can easily verify that a top-down and bottom-up activation cycle with the equations above can implement exactly the similarity measure defined previously.

the activation of the dimension is equally divided among all the values of the dimension that are allowable within the chunk. So in effect, the top-down weight for a dimension is divided equally among all the allowable values.

20. Bottom-up activation also needs to take into consideration multiple allowable values in a dimension of a chunk. First, the activations of the multiple values within a dimension that are allowed within the chunk are combined by taking the maximum (i.e., by using the function *max*); then, across dimensions, the weighted sum specified above is applied.

In addition to enabling similarity-based reasoning through cross-level interaction, the bottom level of the NACS also captures implicit non-action-centered processes within itself. Implicit processing within the bottom level of the NACS is accomplished using a number of different types of neural networks connecting distributed (micro)feature nodes in various ways (Sun, 2003).

Some networks at the bottom level of the NACS are hetero-associative, such as Backpropagation (MLP) networks (see, e.g., Sun, Zhang, & Mathews, 2009). As discussed before, in such a network, input-to-output mappings are established through learning so that, given a situation, proper inferences can be made. See the description in Section 3.1 (because the ACS uses the same type of network).

Another form of hetero-associative network is DFT (i.e., decision field theory; see Busemeyer & Johnson, 2008). DFT is more complex but can account for many psychological phenomena of decision making. See Sun and Helie (2013) for details (see also Chapter 5 for some discussion).

Some other networks in the bottom level of the NACS are auto-associative, in which, once trained, a pattern tends to correct or complete itself from an incomplete or "faulty" version. This allows the retrieval of learned chunks using partial match of (micro)features. Auto-associative mapping may be accomplished simply by using a feed-forward Backpropagation (MLP) network in which each input pattern is mapped to itself (i.e., inputs and outputs are always the same). Auto-associative networks also include Hopfield networks (Hopfield, 1982; Grossberg, 1988), in which nodes are fully connected to each other (i.e., each node in a network is connected to all the other nodes of the network).

A more complex Hopfield-type (fully connected) network has been used in the bottom level of the NACS (e.g., Helie & Sun, 2010). This network, known as NDRAM (Chartier & Proulx, 2005), allows the learning of continuous-valued patterns as attractors. The activation within the network is determined as follows (in a synchronous fashion):

$$x_{i[t+1]} = g\left(\sum_{j=1}^{N} w_{ij} x_{j[t]}\right)$$

where $x_{i[t+1]}$ is the activation of node i in the network at time $t + 1$, w_{ij} is the weight from node j to node i, N is the total number of nodes in the network, and

$$g(x) = \begin{cases} +1 & , \quad \text{if} \quad x > 1 \\ (\delta + 1)x - \delta x^3, & \text{if} \quad -1 \le x \le 1 \\ -1 & , \quad \text{if} \quad x < -1 \end{cases}$$

where $\delta > 0$ is a parameter representing the slope of the transmission function (by default, $\delta = 0.4$). See Figure 3.5 for a graphic representation of this function.

A rough explanation of the network above, without going into technical details, is as follows. A network like this may be viewed as a dynamic system. The activation patterns of the network may be viewed as a phase space in a dynamic system. Within the dynamic system, learned patterns are captured by attractors within the phase space, and

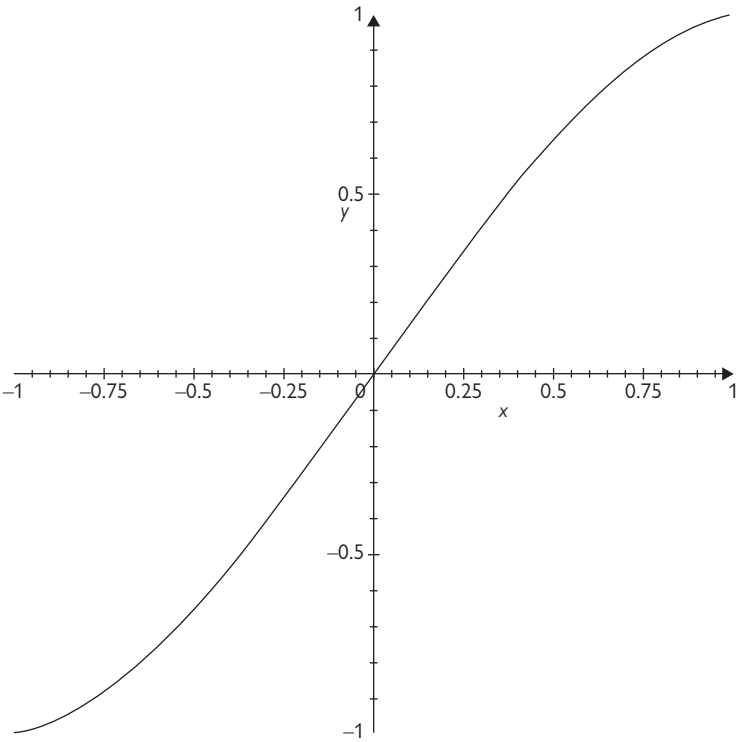

Figure 3.5. The activation function *g(x)* (with the default parameter value). The horizontal axis indicates *x*, while the vertical axis indicates *g(x)*.

trajectories within the phase space from input patterns often converge to attractors. The phase space in the dynamic system (resulting from the network) actually represents a psychological space: concepts/categories in a psychological space are captured by attractors in the phase space; new concepts/categories can be learned and added to the psychological space through creating new attractors in the phase space; existing concepts/categories can be retrieved from the psychological space through trajectories converging to existing attractors in the phase space (Helie & Sun, 2010).

3.2.2.4. Representation of Conceptual Hierarchies

Continuing the discussion of representation, a pertinent issue is the representation of conceptual hierarchies, which are important to human reasoning. In Clarion, conceptual hierarchies can be captured using similarity-based reasoning within the NACS, through the interaction of the two levels, without explicit representation of hierarchies. Hierarchical relations may be viewed as special cases of similarity, and therefore similarity-based reasoning can carry out reasoning based on hierarchies. However, in Clarion, alternatively or simultaneously, conceptual hierarchies can also be explicitly represented at the top level of the NACS. Details of the representation will not be covered here, because they are not essential for subsequent discussions; but they are explained more in the appendix at the end of the chapter for any reader interested in the issue (see also Sun, 2003).

3.2.3. Learning

3.2.3.1. Learning in the Bottom Level

Within the semantic memory of the NACS, for implicit learning at the bottom level, many hetero-associative or auto-associative learning algorithms can be applied to hetero-associative or auto-associative networks there respectively.

For hetero-associative learning using the Backpropagation learning algorithm in an MLP (Backpropagation) network, inputs are mapped to outputs through adjusting weights associated with each layer of nodes within the network (Rumelhart et al., 1986; Levine, 2000). Weight adjustment is usually based on gradually minimizing an error function

that measures deviation from the desired input-output mappings, which leads to approaching the desired mappings through many iterations of weight adjustment. With this learning algorithm, learning may be online (e.g., learning occurs each time a stimulus is presented) or in a batch mode. Parameters for such a network include: number of layers (if not the default number of three layers), number of nodes in each layer, weights and thresholds associated with each node (which can be learned or externally set), learning rate, momentum, and so on. Because these details are standard and not unique to Clarion, I will not get into them here. They are sketched in the appendix.

For auto-associative learning in a Hopfield-type (fully connected) network, a Hebbian learning algorithm may be applied. In such a learning algorithm, weight updating is based on co-activations of nodes (instead of minimizing an error function as in the case of MLP networks): If two nodes are activated at the same time, the weight between them is strengthened (so that they are more likely to be activated together in the future). This can lead to creating an attractor in the network for remembering coactivation patterns. For instance, a learning algorithm used in the NACS is the NDRAM learning algorithm (Chartier & Proulx, 2005). With this algorithm, learning is online (i.e., learning occurs each time a stimulus is presented). Parameters for this network include: number of nodes, learning rate, memory efficiency, vigilance, and so on. The algorithm is not specific to Clarion; therefore, details are only sketched in the appendix.

3.2.3.2. Learning in the Top Level

At the top level of the semantic memory of the NACS, like at the top level of the ACS, chunk nodes and rules can be learned. There are a number of possibilities in this regard.

First, chunk nodes and associative rules in the top level of the NACS can be learned by being given from an external source, or from another component of Clarion. The given information can then be encoded in the forms of chunk nodes or associative rules, as appropriate. (Or the information may be coded by using more complex "fixed rules" as mentioned before; see Sun, 2003.)

Second, chunk nodes and associative rules in the top level can also be learned by acquiring explicit knowledge from the bottom levels of the NACS or even the bottom level of the ACS. This aspect will be covered in

Section 3.3; all issues related to knowledge extraction, assimilation, and transfer will be covered there.

In addition, at every step, each item experienced has a certain probability of being encoded in the NACS as a chunk (an episodic and/or a semantic chunk) with a chunk node at the top level. This aspect will also be covered in Section 3.3.

The NACS serves the need of an individual to cope with the world, and in particular to deal with everyday activities, just as the ACS. Like the ACS, the NACS actively "participates" in the activities of an individual through meeting the individual's informational needs in such activities. Learning serves to improve the capability of the NACS. Different possibilities of learning as sketched above should be viewed in this light, in accordance with the ecological-functional perspective (Sun, 2002, 2012; Section 3.3).

3.2.4. Memory Retrieval

It might be useful to see how memory retrieval is accomplished in the NACS. Let us look into some typical forms of memory retrieval as often tested in psychological experiments, such as free recall, cued recall, recognition, and so on. They may be accomplished differently in the NACS. (I will focus on semantic memory; see, for example, Helie and Sun, 2014b.)

According to Clarion, cued recall consists of presenting a cue to the NACS (by the ACS) and then performing reasoning within the NACS based on the cues. There are a number of possibilities:

- When there are explicit rules at the top level of the NACS concerning the given cue, these rules may lead to the activation of some chunk nodes representing items to be recalled. Thus these items are recalled (as directed by the ACS).
- Recall may also be done through the interaction between the top and the bottom level of the NACS (as determined by the ACS). It may be accomplished as a special case of similarity-based reasoning (discussed earlier); that is, recall may be done based on similarity of the cue to various existing chunks in the NACS. Some chunk nodes at the top level may be activated in the end through such similarity-based reasoning (via top-down and bottom-up activation flows that calculate similarity). Chunk

nodes (representing chunks that are sufficiently similar to the cue) are thus recalled (as determined by the ACS).[21]

- Going further, associative memory networks at the bottom level (not just microfeature representation) may also be involved in cued recall (as determined by the ACS). In such a case, an associative memory network maps the given cue to another pattern (e.g., through a Hopfield-type attractor network or a Backpropagation network). Corresponding chunk nodes at the top level, if exist, are activated by bottom-up flows from that pattern. Such chunk nodes are thus recalled (as directed by the ACS).

For free recall, retrieval may be initiated (by the ACS) with a random activation pattern in the bottom level of the NACS in an auto-associative network (a Hopfield-type network in particular; Helie & Sun, 2010), which then leads to a stable activation pattern (which may in turn activate corresponding chunk nodes at the top level, if exist, through bottom-up activation flows).[22] In the case of a Hopfield-type attractor network, settling may take multiple cycles. Attractors with larger attractor fields are more likely to be settled into (thus corresponding chunk nodes at the top level, if exist, are more likely to be activated through bottom-up activation flows). The activated chunk nodes are thus recalled (as directed by the ACS). Alternatively, if there are relevant explicit rules at the top level of the NACS, they can be used also.

In either of these two cases above, if one item (one chunk) needs to be selected and returned to the ACS (as determined by the ACS), all activated chunk nodes at the top level compete through a Boltzmann distribution of activations. The winner is sent back to the ACS (along with its internal confidence level as described before).

According to Clarion, recognition may be accomplished through presenting an item to be recognized to the NACS (by the ACS). It may then proceed in a number of ways:

- If there are explicit rules at the top level indicating whether the item should be recognized or not, then the rules may be applied (as determined by the ACS).

21. Each of these chunks may represent an individual item. But it may also be a "prototypical" representation of a group of such items (more later). The same goes for attractors in an attractor network.

22. However, even "free recall" may not be completely free.

- Alternatively, the interaction between the two levels of the NACS may be involved. Some items may be coded as chunks (with chunk nodes at the top level and microfeature nodes at the bottom level). The similarity between the item to be recognized and the chunk nodes is computed through top-down and bottom-up activation flows. Judgment is rendered on the basis of similarity (as directed by the ACS).
- Alternatively, an auto-associative network (a Hopfield-type attractor network in particular) at the bottom level of the NACS determines whether an item should be recognized or not (as directed by the ACS), based on the activation pattern resulting from the auto-associative mapping—whether it is sufficiently similar to the original item.

In each of these types of memory retrieval, if relevant implicit and explicit knowledge both exist, either or both may be used, as determined by the ACS (and/or the MCS). Exactly which possibility or which combination of possibilities materializes in a given situation depends on a number of factors such as how much knowledge that one has, what kind of knowledge that one has, nature of the instructions that one receives, individual cognitive style, prior experience, and so on.

Clearly, during memory retrieval, various forms of reasoning happen within the NACS. For example, rule application (at the top level), similarity matching (involving both levels), settling within an attractor network (at the bottom level), and so on are all forms of reasoning within the NACS. Human memory is rarely a literal process; it is often a constructive process involving reasoning of various kinds. In human memory, there is often more than just a single location where an item is found. Even at the time of memory encoding, errors, distortions, or omissions may occur. Human memory may also decay. Reasoning and other processes may be necessary to recover needed information. In Clarion, by utilizing these mechanisms discussed above, information may be reinterpreted, reconstructed, and combined.

3.2.5. An Example

Below, an example about representation, learning, and memory retrieval within the NACS is explained. (For detailed simulations, see chapters 5, 6, and 7.)

In an experiment on artificial grammar learning, there is a training phase first, followed by a test phase. During the training phase, subjects are asked to memorize a set of strings (which, unbeknownst to the subjects, was generated according to a finite-state grammar). After the training phase, a test phase ensues, during which subjects are asked, among other things, to recall or recognize strings, or to complete partial strings.

According to Clarion, strings are memorized within the NACS in various ways; each time a string is presented, memory is strengthened. I focus only on semantic memory here.

At the bottom level of the NACS, implicit memory is kept. When a Hopfield-type attractor network is used, attractors are gradually created and strengthened within the network as a result of seeing these strings (see learning details in the appendix). After sufficient experiences, those attractors representing given strings are established.

At the top level, specific chunk nodes for representing these strings may be established, with each chunk node indicating a particular string. But they are subject to forgetting, failure to encode, and so on (see the relevant parameters discussed earlier). Associative rules for representing strings may also be established (see rule learning in Section 3.3). They are also subject to forgetting, failure to encode, and so on.

During the test phase of the experiment, subjects may be asked to recall, recognize, or complete strings. For instance, at the test phase, strings are presented one at a time and the subjects are asked to judge whether these strings have been seen before or not (i.e., undergo recognition tests). The NACS may respond using a number of mechanisms. For instance, some of the seen strings may have been coded as chunks (with chunk nodes at the top level). In that case, similarity-based reasoning may be invoked comparing the chunks with a given string through top-down and bottom-up activation flows. Recognition happens when there is sufficient similarity. For another instance, a Hopfield-type attractor network at the bottom level may be used, in which settling into an attractor that closely resembles the given string (or failure to do so) may be used as the basis for making the judgment.

During the test phase, subjects may also be asked to recall as many strings as possible. In that case, retrieval occurs mainly in the bottom level of the NACS (because likely they have learned few explicit rules at the top level). For example, within a Hopfield-type attractor network in the bottom level of the NACS, a random initial activation pattern leads to a

settling process, which may lead to settling into one of the attractors of the network, and thus a string corresponding to the attractor is retrieved. This process may be repeated multiple times to retrieve multiple strings.

Furthermore, during the test phase, partial strings may be presented to the subjects and they are asked to complete these strings. In this case, completion occurs mainly in the bottom level of the NACS (because likely there are few explicit rules at the top level). One possibility is that a partial string is presented to a Hopfield-type attractor network at the bottom level of the NACS, and then the settling process may lead to an attractor that represents a possible completion of the partial string. Another possibility is to use a Backpropagation (MLP) network at the bottom level of the NACS to complete these strings (mapping partial strings to full strings).

3.3. Knowledge Extraction, Assimilation, and Transfer

3.3.1. Background

In this section, I describe learning methods that transfer knowledge from one component (module, level, or subsystem) of Clarion to another, which lead to more complex or more effective knowledge representation in many circumstances (e.g., synergy as mentioned before).

As discussed before, implicit knowledge can be acquired through trial and error, and on top of that explicit knowledge can be acquired through the mediation of implicit knowledge: hence the notion of bottom-up learning. The basic process of bottom-up learning of procedural knowledge in Clarion (i.e., the RER algorithm as mentioned before) is as follows (Sun et al., 2011): if an action implicitly decided by the bottom level is successful, then one extracts an explicit rule that corresponds to the action selected by the bottom level and adds the rule to the top level. Then, in subsequent interaction with the world, one verifies the extracted rule by considering the outcome of applying the rule: if the outcome is not successful, then the rule should be revised and made more specific; if the outcome is successful, the rule may be generalized to make it more universally applicable. Details of bottom-up learning will be addressed in this section.

However, although one can learn without externally provided knowledge, one can make use of such knowledge when it is available (Dreyfus

and Dreyfus, 1987; Anderson and Lebiere, 1998). In Clarion, to deal with such learning, externally provided knowledge, in the form of explicit symbolic conceptual representation can (1) be combined with existing explicit representation at the top level (i.e., internalization), and (2) be assimilated into implicit processes at the bottom level (i.e., assimilation). This process is known as top-down learning, which is more naturally accomplished in Clarion than in other models. Top-down learning will also be discussed in this section.

In addition, as yet another way of converting knowledge forms to facilitate its use (as people often do; Klein et al., 2002), in Clarion, knowledge acquired in one subsystem may be transferred to another subsystem. In particular, knowledge acquired within the ACS in interacting with the world (through action decision making) may be transferred to the NACS for purposes of reasoning. Such transfer will also be described in this section.

As a result of these learning processes, explicit, symbolic representation is grounded in lower-level processes from which it obtains its meaning and for which it often provides focus and clarity (Sun, 2012). This groundedness is guaranteed by the way in which higher-level representation is produced—it is, in the main, extracted out of lower-level processes and contents (in particular, implicit reactive routines). Even external, culturally transmitted symbols and other explicit representations have to be linked up, within the mind of an individual, with lower-level processes in order to be effective. Clarion captures such groundedness.

It is worth noting that culture also structures (constrains) the interaction of an individual with the world through mediating tools, signs, and other cultural artifacts. Thus culture affects lower-level processes too (in particular, implicit reactive routines and their learning and explication), although maybe to a lesser extent.

3.3.2. Bottom-Up Learning in the ACS

3.3.2.1. Rule Extraction and Refinement

In the ACS, while implicit reactive routines are being learned at the bottom level, explicit rules at the top level can also be learned using information already acquired at the bottom level, that is, bottom-up learning. Bottom-up learning has been implemented in the *Rule Extraction and*

Refinement (RER) algorithm (Sun et al., 2001). The basic idea of the RER algorithm has been explained above. This process can, in a sense, be viewed as the autonomous generation of symbolic representation from subsymbolic representation (Sun, 2013b).

For example, an initial rule resulting from RER may be: "If the size of an object is large and its speed is fast, then stay away." Or to put it in another way: "If the value of input dimension 1 is 3 and the value of input dimension 3 is 5, then do action 2." Generalization may lead to adding one more allowable value in one of the input dimensions. For example, generalization of the rule above may lead to: "If the size of an object is medium or large and its speed is fast, then stay away." Specialization may lead to removing one allowable value in one of the input dimensions.

To carry out RER, the following is done within each action cycle in the ACS:

1. Update rule statistics used for rule extraction, generalization, and specialization.
2. Check the current criterion for rule extraction, generalization, and specialization:
    2.1. If the result is successful according to the current rule extraction criterion, and there is no rule matching the current state and action, then extract a new rule. Add the extracted rule to the top level of the ACS.
    2.2. If the result is unsuccessful according to the current specialization criterion, then revise all the rules matching the current state and action through specialization:
        2.2.1. Remove these rules from the top level.
        2.2.2. Add the specialized versions of these rules to the top level.
    2.3. If the result is successful according to the current generalization criterion, then generalize the rules matching the current state and action:
        2.3.1. Remove these rules from the top level.
        2.3.2. Add the generalized versions of these rules to the top level.

One can find psychological arguments in favor of this kind of algorithm in, for example, Bruner, Goodnow, and Austin (1956), Dominowski (1972), Sun et al. (2001), and Sun (2013b).

Let us examine the operations within the algorithm above. At each action step, the following information is examined: *(x, y, r, a)*, where *x* is the state before action *a* is performed, *y* is the new state after action *a* is performed, and *r* is the immediate reinforcement received right after action *a*. Based on the information, the positive and negative match counts, $PM_a(C)$ and $NM_a(C)$, are updated (in step 1 of the algorithm), for the condition of each matching rule and each of its minor variations (e.g., the rule condition plus or minus one possible value in one of the input dimensions), in relation to the action just performed. The Positive Match count, $PM_a(C)$, equals the number of times that an input state matches condition C, action *a* is performed, and the result is positive; the Negative Match count, $NM_a(C)$, equals the number of times that an input state matches condition C, action *a* is performed, and the result is negative.[23]

Positivity (or its opposite, negativity) is determined based on a positivity criterion, which depends on task circumstances. It may be based on information from the bottom level or based on other information (e.g., immediate reinforcement).[24]

Based on *PM*s and *NM*s, an information gain measure, *IG(A, B)*, is calculated. Essentially, the measure compares the percentages of positive matches (i.e., the success rates) under two different rule conditions *A* and *B*. If *A* can improve the percentage (the success rate) to a certain degree over *B*, then *A* is considered better than *B*. This has been justified computationally (see, e.g., Lavrac & Dzeroski, 1994). That is,

$$IG(A,B) = \log_2 \frac{PM_a(A) + c_1}{PM_a(A) + NM_a(A) + c_2} - \log_2 \frac{PM_a(B) + c_1}{PM_a(B) + NM_a(B) + c_2}$$

23. At each step, all relevant *PM*s are incremented when the positivity criterion is met; all relevant *NM*s are incremented when the positivity criterion is not met. At the end of each "episode" (a sequence of steps that together constitute an isolatable event, defined in a domain-specific way), all *NM*s and *PM*s are discounted by a multiplicative factor no greater than 1 (the default is *0.90*). The results are time-weighted statistics (useful in non-stationary situations).

24. For example, when immediate feedback is given, positivity may be determined by:

$$r > threshold_{RER}$$

Another example used in Sun et al. (2001) is:

$$max_b Q(y, b) + r - Q(x, a) > threshold_{RER}$$

which indicates whether or not action *a* is reasonably good (Sun and Peterson, 1998).

where A and B are two alternative rule conditions that lead to action a, and c_1 and c_2 are two constants representing the prior (the default values are $c_1 = 1$ and $c_2 = 2$).[25]

In the algorithm above, whether or not to extract a rule is decided based on the positivity criterion, which measures whether the current step is successful or not (as determined by the current step: (x, y, r, a)):

- *Extraction*: If the current step is positive according to the current positivity criterion and if there is no rule that matches this step at the top level (matching both the state and the action), then set up a rule corresponding to the current step, that is, "if C, then a", where C is a chunk node that specifies the values of all the dimensions exactly as in the current input state x[26] and a denotes the action performed at the current step.[27]

In the algorithm, whether generalization and specialization should be performed is decided based on the information gain measure. If, in terms of the IG measure, a rule is better (to some extent) than its corresponding "match-all rule" (i.e., the rule with the same action but with a condition that matches all possible input states), then the rule is considered successful and eligible for generalization. Otherwise, specialization should be considered.

As the example earlier illustrated, generalization amounts to adding an additional value (disjunctively) to one input dimension in the condition of a rule, so that the rule will have more opportunities of matching inputs, and specialization amounts to removing one value from one input dimension in the condition of a rule, so that it will have fewer opportunities of matching inputs. That is,

25. This measure compares the percentages of positive matches under different conditions A and B, with the Laplace estimator.

26. One may use some form of attention to focus on fewer input dimensions or values. See the description of the MCS in Chapter 4 for details.

27. The issue of the proliferation of rules needs to be addressed. A probability parameter (p_{re}) determines how likely a rule will be extracted, given that the criterion for rule extraction is met. Similar probabilities determine how likely a rule will be generalized or specialized. The default values for these probabilities are 1. Also, as mentioned before, a density parameter (d_r) determines the minimum frequency of matching inputs in order not to forget an action rule.

- *Generalization*: If the current rule is successful and there is a slightly generalized condition that is potentially better, then use a slightly generalized condition for the rule.[28]
  Technical Description of Generalization: Technically, If $IG(C, all) > threshold_1$ and $max_{C'} IG(C', C) \geq 0$, where C is the current condition of a rule matching the current state and action, *all* refers to the corresponding match-all rule (with the same action as specified by the original rule but with a condition that matches any input state), and C′ is a modified condition such that C′ = C plus one value (i.e., C′ has one more value in one of the input dimensions), then set $argmax_{C'} IG(C', C)$ as the new (generalized) condition of the rule.
- *Specialization*: If the current rule is unsuccessful, but there is a slightly specialized condition that is better, then use a slightly specialized condition for the rule.
  Technical Description of Specialization: Technically, If $IG(C, all) < threshold_2$ and $max_{C'} IG(C', C) > 0$, where C is the current condition of a rule matching the current state and action, *all* refers to the corresponding match-all rule (with the same action as specified by the original rule but with a condition that matches any input state), and C′ is a modified condition such that C′ = C minus one value (i.e., C′ has one fewer value in one of the input dimensions), then set $argmax_{C'} IG(C', C)$ as the new (specialized) condition of the rule.[29]
- *Deletion*: In the operation of specialization described above, removing the last value from any dimension of a rule, if performed, makes it impossible for the rule to match any input state. So in that case the rule is deleted.

RER may generate explicit knowledge that supplements implicit knowledge at the bottom level, and together they may lead to better overall performance. Although explicit rules were extracted from the bottom level in the first place, the resulting explicit representation, different from that of the bottom level in characteristics, makes these rules useful. Sun et al. (1998) identified the following factors that contributed to

28. For example, Dominowski (1972) showed that even when their hypotheses were correct, human subjects tended to shift their hypotheses anyway.

29. Clearly one should have: $threshold_2 \leq threshold_1$, to avoid oscillation.

the synergy between the top and the bottom level: (1) the complementary representational forms of the two levels (discrete versus continuous); (2) the complementary learning processes of the two levels (one-shot rule learning versus gradual weight tuning); and (3) the bottom-up rule learning criterion as described above.[30]

3.3.2.2. Independent Rule Learning

The Independent Rule Learning (IRL) algorithm does not involve the initial extraction step used by the RER algorithm. That is, it does not use the information from the bottom level for the initial extraction of a rule; hence the term "Independent Rule Learning." But IRL may be considered a kind of bottom-up learning also, because in the other steps of the algorithm, information from the bottom level is used for refining rules. IRL is somewhat similar to but less bottom-up than RER.

An IRL rule specifies some constraints on inputs as its condition and some constraints on outputs as its action, thus different from RER rules. For example, a IRL rule, different from RER rules, may be: "if the value of input dimension 1 is greater than the value of input dimension 2 and the value of input dimension 3 is 15, then output the value of input dimension 4." So the condition of an IRL rule may be a "chunk template" rather than a chunk. As a result, IRL rules may be more complex than RER rules.

With IRL, rules of various forms are generated at the top level. Then, these rules are tested through experience using IG measures. In general, multiple sets of rules are generated in a domain-specific order (or randomly), one set at a time, and rules within the current set are tested through experience.

In IRL, the positivity criterion in calculating IG may be based on information from the bottom level or based on other information, the same as discussed before with regard to RER. Based on a defined positivity criterion, an application of the IG measure (as defined before) to IRL rules concerns rule deletion:

If $IG(C, random) < threshold_3$, delete the rule C.

30. A more technical explanation can also be provided, for example, based on considerations of function approximation. See Sun and Peterson (1998) for details.

where *random* refers to completely random actions. The match-all rule used in RER is not used here, because IRL rules are more complex—an IRL rule may recommend different actions in different input states, based on the constraints specified within the rule. Thus the match-all rule turns into random actions in the formula above. If the IG measure of a rule falls below a threshold, then the rule is deleted (cf. Dienes and Fahey, 1995).

Within the condition of an IRL rule, beside prespecified constraints, there may be a second part, which specifies allowable values in various input dimensions as in RER rules.[31] Subsequently, generalization and specialization can be performed on this second part in the same way as in RER. Generalization and specialization can be performed based on the IG measure as defined earlier for RER (because this part resembles RER).[32]

3.3.2.3. Implications of Bottom-Up Learning

First, in bottom-up learning in the ACS, although the accumulation of statistics is gradual at the bottom level, the acquisition and refinement of rules at the top level are one-shot and all-or-nothing. Therefore, bottom-up learning is different from gradual weight tuning at the bottom level (e.g., by using reinforcement learning in neural networks).

Second, in the ACS, an explicit concept is learned and an explicit representation of the concept is established as a result of extracting an action rule, in the process of, and for the sake of, accomplishing a particular task at hand. Specifically, when the condition of an action rule is established, a localist encoding of the condition is also established at the top level and a new symbol is thus formed (in the form of a chunk node), if it was not there already.

More specifically, when a rule is extracted, a unitary entity (a chunk node) is set up in the top level of the ACS to represent the condition of the rule as a whole, which connects to the (micro)feature nodes

31. If an initial rule does not specify any allowable values of any input dimensions in its condition (besides some constraints about inputs and outputs), the second part of the rule condition is considered as consisting of all the values in all the input dimensions initially. Of course, no generalization can be performed on such a rule initially. If allowable values of only some input dimensions are specified initially, then other, unspecified input dimensions are considered to allow all possible values initially.

32. Probability, density, and other parameters similar to those used for RER are also present for IRL.

(values of various dimensions) represented in the bottom level of the ACS. Together they form a chunk. The chunk is subsequently refined in the process of refining the rule. For example, a rule may be extracted in the top level of the ACS as follows: "if (temp, warm) (rainfall, heavy), then (action, stay-inside-building)". A chunk node is set up in the top level of the ACS to represent the following chunk: "*((temp, warm) (rainfall, heavy)*)". The chunk node is linked to the (micro)feature nodes representing *(temp, warm)* and *(rainfall, heavy*) respectively at the bottom level. This chunk (with its chunk node at the top level) is acquired through extracting the action rule, which happens in the process of accomplishing some particular task.

Third, as a result of bottom-up learning, explicit (symbolic) knowledge acquired is concerned with existentially significant aspects of the world in relation to the individual involved, and in particular to the individual's needs and goals (Sun, 2012). In other words, explicit concepts and rules learned (at the top level of the ACS) are concerned with those aspects of the world that have significant bearings on an individual's survival and functioning in the world. They are not strictly "objective" classifications of objects, persons, and events in the world, but the direct result of the interaction of the individual with the world, manifesting the regularities encountered in such interactions (Merleau-Ponty, 1963) and reflecting the intrinsic needs and the goals of the individual (on the basis of the motivational and metacognitive subsystems, discussed in Chapter 4). In addition, they are also action-oriented, concerned specifically with helping to decide on what to do in frequently encountered situations. Thus, explicit knowledge in the ACS is not strictly objective, but action-oriented, goal/need-oriented, and oriented towards the activities of an individual interacting with the world (Heidegger, 1927; Merleau-Ponty, 1963).

In this way, an individual projects his or her own perspectives and needs onto the world and brings forth the meanings of situations encountered and the meanings of symbolic representation acquired (Johnson, 1987). Thus, explicit symbolic representation acquired is grounded in implicit, subsymbolic representation and in interaction between the individual and the world (Sun, 2013b).

Sun (2002) provided some computational analyses of explicit concepts formed within Clarion in the context of specific tasks. It was found that the concepts formed were indeed concerned with those aspects of the environment that were important to the tasks at hand, serving the purpose of facilitating action decision making in accomplishing the tasks.

Specifically, in the contexts that were analyzed, those concepts formed help to identify certain combinations of environmental features (dimensional values represented at the bottom level) that are significant for action decision making, thus helping to make proper action decisions, thereby facilitating the accomplishment of the goals with regard to the tasks.

3.3.3. Top-Down Learning in the ACS

As defined before, top-down learning is the process by which explicit knowledge (often externally given and sociocultural in nature) is assimilated into implicit representation. In the ACS, first, externally given explicit action-centered knowledge is expressed as chunk nodes and action rules at the top level of the ACS (in formats as discussed before). Assimilation into the bottom level of the ACS is then accomplished, either by using supervised learning in which the top level serves as the "teacher" (e.g., using the Backpropagation learning algorithm at the bottom level), or through gradual practice guided by explicit knowledge (e.g., using the Q-learning algorithm at the bottom level).

Among these two methods, assimilation through gradual practice guided by the explicit knowledge is always performed in the ACS. That is, with explicit knowledge (in the form of action rules) in place at the top level, the bottom level learns under the "guidance" of the rules automatically when explicit knowledge is used in deciding on an action.

Given explicit rules at the top level, initially, one may rely mostly on them for action decision making. Meanwhile one learns implicit knowledge at the bottom level through "observing" the actions directed (mostly) by the action rules at the top level, using the same reinforcement learning algorithm at the bottom level as described before. After each action is selected and performed, reinforcement learning occurs at the bottom level. Such reinforcement learning is always carried out at the bottom level, regardless of whether actions are decided by the top level or the bottom level. Gradually, when more and more implicit knowledge is acquired by the bottom level in this way, one relies more and more on the bottom level (when the level integration mechanism, as mentioned before, is adaptable). Hence, top-down learning takes place (Sun, 2002, 2003).

Supervised learning can also be performed for assimilation of explicit knowledge into implicit knowledge, in which the top level serves as the "teacher" for the bottom level. At each step, an explicit rule is selected

from the top level (as the input and the target output) and used to train the bottom level using a supervised learning algorithm (e.g., the Backpropagation learning algorithm).[33]

Through top-down learning, explicit procedural knowledge becomes implicit, embodied skills in the ACS and thus becomes more efficient (Dreyfus & Dreyfus, 1987). This process is another possibility for the grounding of (previously generated, externally given, possibly sociocultural) explicit symbolic representation. In this way, externally given explicit representation is grounded in low-level implicit representation and in activities interacting with the world. That is, it is assimilated into, as well as linked up to, low-level implicit representation, and it becomes enmeshed in the ongoing interaction between the individual and the world, which is the context within which assimilation takes place. Therefore, such knowledge can be used in ways that enhance the functioning of the individual in the world.

Note that externally given explicit knowledge, if not useful in enhancing the functioning of the individual in the world, may simply be forgotten and removed (e.g., as dictated by the "density" parameters described earlier).

3.3.4. Transfer of Knowledge from the ACS to the NACS

Within the NACS, chunk nodes at the top level (for representing explicit knowledge) are acquired in a variety of ways from a variety of sources. These sources include the ACS: each input state experienced by the ACS as a whole is coded by a chunk node in the NACS; so is each action chosen by the ACS. Even each perception-action step experienced by the ACS as a whole (which includes the input state, the action in that state, the next input state following the action, and the immediate reinforcement following the action) can be coded by a chunk node.[34]

Within the NACS, these aforementioned types of chunks (with their chunk nodes at the top level) are part of semantic memory. They are experience based, because they are created due to experiences and they

33. A frequency parameter indicates the relative frequency of supervised learning, relative to regular action steps (e.g., performing *m* steps of supervised learning for every *n* steps of action decision making).

34. If a chunk (with a chunk node) is transferred to the NACS, a threshold determines the minimum activation of a dimensional value to be included in the chunk.

reflect such experiences (although they are different from episodic memory; more below). These chunks in the NACS are created as they are experienced by the ACS. So, in a sense, they are transferred from the ACS to the NACS; in other words, they are transferred from procedural memory to declarative (semantic) memory. As a result of such transfer, they are task oriented and concerned with the person-world interaction.

In semantic memory of the NACS, chunks (with chunk nodes at the top level) are also formed when they are created within the ACS as a result of rule learning. As mentioned before, in the ACS, a condition chunk is created as a result of learning an action rule. When the condition of an action rule is established, a localist encoding of that condition (a chunk node) is established at the top level of the ACS, which connects to its (micro)features (dimensional values) represented at the bottom level of the ACS. At the same time as the rule learning occurring in the ACS, a corresponding chunk is set up in semantic memory of the NACS and its chunk node at the top level is linked to (micro)features (dimensional values) at the bottom level of the NACS.[35] This is another instance of transfer of procedural knowledge to declarative (semantic) knowledge (from the ACS to the NACS).

Although semantic knowledge can be generated as a result of specific past experiences (as these types of chunks above demonstrate), semantic knowledge is not tied to specific past experiences. In contrast, episodic knowledge in episodic memory is directly tied to specific past experiences (with specific time and other episodic information included as part of the encoding). As discussed before, the distinction between semantic and episodic knowledge has been well argued for.

In Clarion, for instance, each action cycle as a whole is coded as an episodic chunk (with a chunk node), within episodic memory of the NACS. In addition, the following types of items can also be created within episodic memory of the NACS: each input state as observed by the ACS, each action chosen by the ACS, and so on.

35. To avoid proliferation, explicit knowledge in the NACS is subject to some parameters, similar to the ACS. An encoding probability parameter (p_c) determines how likely an encoding of a chunk node will be successful. Likewise, an encoding probability parameter (p_a) determines how likely an encoding of an associative rule will be successful. A density parameter (d_a) determines the minimum frequency of invocation (encoding, reencoding, extraction, reextraction, or application) of an associative rule in order to keep it. Similarly, a density parameter (d_c) determines the minimum frequency of invocation (encoding, reencoding, extraction, reextraction, or activation) of a chunk node in order to keep it.

Furthermore, other types of entities within the ACS and the NACS (e.g., each action rule in the ACS, each chunk in the ACS, each associative rule in the NACS, each chunk in the NACS, each association inferred from the bottom level of the NACS, and so on), once invoked, can spawn a corresponding item within episodic memory of the NACS. Evidently, many of the aforementioned episodic items may be considered transferred from the ACS to the NACS. Moreover, episodic knowledge is also task oriented, resulting from the person-world interaction.

Another way to learn declarative knowledge is taking externally given knowledge and encoding it in the NACS. Explicit declarative knowledge can be given from external sources, received as inputs by the ACS.[36] Then, directed by the ACS, it can be encoded in semantic memory of the NACS, using associative rules and chunk nodes (which are created if not previously existing).

Note that symbols (in the form of chunk nodes) that have been formed, despite the fact that they reside in the NACS, are task oriented and context dependent because they are formed in relation to the tasks and goals at hand and for the purpose of exploiting environmental regularities. For instance, a symbol (a concept) is formed as part of an action rule in the ACS, which is learned to accomplish a goal within a task to fulfill a need in a particular environment. Such contexts help to determine which set of (micro)features in the environment needs to be attended to together. As a result, acquired symbols (concepts) are functional, even when they are transferred to the NACS. Knowledge acquired in this way is concerned with existentially and ecologically significant aspects of the world: that is, concerned with those aspects of the world that have significant bearings on an individual in interaction with the world and ultimately in survival in the world. They are not strictly "objective" classifications of the world but rather the result of the interaction with the world and the projection of one's needs and goals (Sun, 2013b). Thus, even in the NACS, Clarion emphasizes the functional role of symbols/concepts and the importance of function, need, and goal in forming symbols/concepts.

Note also that in semantic memory, the two-level representation with both chunk nodes and (micro)feature nodes constitutes, to some

36. Externally given knowledge may be presented in forms that can be transformed into rules and chunks (details of the transformation are not dealt with here).

extent, a "prototype" model of concepts (e.g., Smith and Medin, 1981). A localist chunk node at the top level serves as the identification of a set of correlated (micro)features at the bottom level, when activated in a bottom-up direction. A chunk node at the top level also serves to trigger (micro)features at the bottom level, in a top-down direction, once the corresponding concept is brought into attention (i.e., activated).

3.3.5. Bottom-Up and Top-Down Learning in the NACS

In semantic memory of the NACS, chunk nodes at the top level, acquired from external sources, through action rule extraction in the ACS, or through other means, are used to encode explicit knowledge extracted from the bottom level of the NACS, in the form of explicit associative rules between these chunk nodes.

Specifically, in semantic memory of the NACS, an explicit associative rule is extracted at the top level, when an implicit associative mapping is performed in the bottom level. Associative rules are established at the top level of the NACS between the chunk node(s) denoting the cue for the mapping and each of the chunk nodes denoting outcomes from the mapping.

More specifically, a set of (micro)features is activated as the cue for performing an implicit associative mapping, in which case a chunk node is set up in the top level to represent the cue (if there is no chunk node already there linked to the same set of nodes at the bottom level). Alternatively, a number of existing chunk nodes at the top level are activated as the cue, in which case their corresponding (micro)features are then activated at the bottom level to perform an implicit associative mapping. Corresponding to the associative mapping at the bottom level, explicit associative rules are set up at the top level (if not there already). That is, an associative rule is established that connects the chunk node(s) representing the cue with each chunk node "compatible" with the result of the associative mapping at the bottom level. Among "compatible" chunk nodes are those existing ones sufficiently activated by bottom-up activation from the result of the associative mapping at the bottom level. Another "compatible" chunk node, which is set up if not already there, corresponds to the result from the bottom level as a whole.[37]

37. The extraction threshold for chunks ($threshold_{ce}$) specifies the minimum activation level of the resulting chunk node for chunk extraction to be considered. The probability

Similar to explicit representation in the ACS discussed before, explicit chunk nodes and explicit associative rules formed in ways specified above in the NACS are also task oriented to some extent, because they are formed in relation to specific tasks at hand when the NACS reasoning capacities are invoked by the ACS for dealing with the tasks.

In a different direction, explicit associative rules at the top level of the NACS can be used to train the bottom level of the NACS, similar to top-down learning within the ACS described earlier.

3.3.6. Transfer of Knowledge from the NACS to the ACS

Transfer of knowledge in the other direction, from the NACS to the ACS, is also possible. For instance, results of reasoning from the NACS can be sent to the ACS through working memory, which can then be used for action decision making within the ACS as well as for other purposes.

More interestingly, information stored within the NACS can also be used for off-line learning within the ACS. For instance, items in episodic memory can be used to train the ACS, as if they were real experiences, which amounts to memory consolidation (from a more individuated form to a more aggregate form, helping to distill statistical regularities; cf. McClelland et al., 1995).[38] Such transfer can be expected to enhance learning within the ACS and to reduce the need for larger amounts of actual experiences.

3.3.7. An Example

Below, I describe an example involving these forms of learning. In particular, it involves the interaction of these different forms of learning in an individual's interaction with the world (sociocultural or physical).

parameter (p_{cc}) determines how likely the extraction of a chunk will be successful (provided that $threshold_{ce}$ is reached). If a chunk is extracted, another threshold determines the minimum activation of a dimensional value to be included in the chunk. Likewise, the extraction threshold for associative rules ($threshold_{ae}$) specifies the minimum activation level for associative rule extraction to be considered (where the minimum activation level concerns a conclusion chunk node resulting from associative mapping). The probability parameter (p_{ae}) determines how likely the extraction of an associative rule will be successful (provided that $threshold_{ae}$ is reached). As before, density parameters determine the minimum frequencies of invocation not to forget an associative rule or a chunk.

38. Episodic memory can also be consolidated into "abstract episodic memory," which can then be used to train the ACS (Sun, 2003).

3.3.7.1. Learning about "Knife"

Imagine the case of a child learning the concept of "knife" (and its related knowledge). Unaware of the danger of a knife, a child approaches the sharp edge of a knife in a way that causes pain. Recoiling from the object, the child quickly registers a rule: This thing is to be avoided. Soon enough, he forgets that rule (having so many other things to learn and remember). So the experience re-occurs under similar or different circumstances. The experiences lead the child to develop a reactive routine: stay away from sharp edges.

On the other hand, parental inputs provide the child with a verbal label for the object: "knife" (while pointing to the object). The label initially is closely associated with the visual image of a particular knife and the pain that it once caused. But gradually, it is generalized in accordance with experiences: "knife" could be in various shapes (although always having a sharp edge), could be in various sizes, has a handle, and so on (i.e., it becomes associated with various visual, tactile, proprioceptive, and other information). In the meantime, the implicit reactive routines associated with knifes lead to more solid establishment, through bottom-up explication, of explicit concepts and explicit rules concerning what a "knife" is and how one should act in relation to it.

The establishment of the explicit concept of "knife" leads to the possibilities of various further knowledge, beliefs, and memory associated with it, and consequently various kinds of reasoning that can be performed in relation to the concept. For instance, the child may associate the concept "knife" with previous experiences related to knifes (i.e., with episodic memory), or with tales that others told him about knifes.

Furthermore, based on the knowledge already acquired, it may occur to the child that if he needs to slice a tomato, knives may be useful. Going further, it may occur to him that if he needs to kill an animal, knives may also be used. From that point on, the child may find many uses for knives and develop much knowledge about them.

A similar description of gradual learning can be applied to a wide range of other circumstances. For example, it may be applied to the learning of concepts of stone tools (such as handaxes) in the prehistoric ages. At the other end of the spectrum, a similar description may also apply to the learning of complex modern technical concepts, such as automobiles, airplanes, computers, and so on. For example, with regard to moving cars, there can be similar descriptions of the instinctual act of getting out of its

way, the verbal label denoting cars, the refinement of the concept behind the label (e.g., with regard to number of wheels, movement, shape, size, color, and so on), episodic memory associated with the concept, further knowledge, further inferences, and so on.

3.3.7.2. Learning about "Knife" within Clarion

I now sketch the details within Clarion that carry out the previously described learning. (For other developmental models, see, e.g., Shultz & Sirois, 2008.)

Extraction within the ACS

First of all, as described before, in the ACS, implicit reactive routines are developed in the bottom level through reinforcement learning from trial-and-error experiences, which form the basis for future action decisions, for example, in the presence of knifes ("stay away from sharp edges").

Based on such implicit reactive routines, explicit concepts and explicit action rules arise within the ACS, for example, through the RER algorithm as described before, namely, through extracting and refining explicit action rules at the top level of the ACS from the information at the bottom level of the ACS.

Using the RER algorithm, an explicit action rule may be created: "if sharp edge, shining metal surface, handle, then do not touch". Clearly, such a rule is concerned with existentially significant aspects of the world. It is extracted through various operations involving extraction, generalization, and specialization (as part of the RER algorithm described before). For example, initially, the following rule was extracted: "if long sharp edge, shining metal surface, wooden handle, then do not touch." Then, through generalization, it became: "if long sharp edge, shining metal surface, handle, then do not touch." Further generalization and specialization led to: "if sharp edge, shining metal surface, handle, then do not touch."

As a result of rule extraction and refinement, concepts (with symbols in the form of chunk nodes) were formed (created and refined); for example, the concept of "knife" captures features such as sharp edge, shining metal surface, and handle. As discussed before, such symbols (concepts) are meaningful, because (1) they are linked to implicit representation at the bottom level and (2) they were created in the process of accomplishing an existentially relevant task and therefore they are existentially relevant to an individual.

In addition, within the ACS, verbal labels may be received from external sources, and used to denote some concepts learned in a bottom-up way or given externally.

Transfer into the NACS

Those concepts with symbolic representation (i.e., chunk nodes) extracted within the ACS are useful outside of the ACS. For one thing, they enable an individual to reason within the NACS about relevant situations. Due to the existential relevance and groundedness of these representations within the context of the ACS, their presence within the NACS provides same relevance to reasoning and other functionalities performed within the NACS. Therefore, the NACS is also grounded in implicit processes and the person-world interaction (as the ACS).

For example, a concept (a chunk with its chunk node) transferred from the ACS represents "knife." It is then used in the NACS for constructing declarative knowledge (e.g., associative rules). For instance, an associative rule within the NACS may be as follows: "if knife, hostile person, then potentially violent situation," or "if knife, hostile person, then dangerous situation" (provided that the other chunks involved, such as "hostile person," "potentially violent situation," and "dangerous situation," have been established within the NACS). Such rules and concepts facilitate reasoning about a task or a situation (in particular in an explicit, deliberative manner at the top level of the NACS, along with reasoning in an implicit and intuitive manner at the bottom level of the NACS). For instance, the associative rules above may be used, along with other possible rules and chunks (concepts), for explicit reasoning about various options in a dangerous standoff.

Extraction within the NACS

Concepts (chunks with chunk nodes) can be extracted within the NACS itself. For instance, in the previously described domain, a concept may be formed as a result of implicit associative mapping performed at the bottom level of the NACS, capturing the outcome of the mapping: "potentially violent situation", which connects to (micro)features at the bottom level. At the same time, associative rules, for example, "if knife, hostile person, then potentially violent situation," may also be extracted from implicit associative mapping at the bottom level of the NACS.

Concepts (symbols) extracted from the bottom level of the NACS may interact with externally provided concepts (symbols), just as concepts

(symbols) extracted and transferred from the bottom level of the ACS may interact with externally provided concepts (symbols). Below is the discussion of this aspect within Clarion.

Interaction of External and Internal Concepts

Externally provided information enables top-down learning (as opposed to bottom-up learning) in the ACS. For example, a parent may instruct the child: "If you see a knife lying around, tell an adult immediately." This instruction is thus set up at the top level of the ACS as an action rule, with its condition and action set up as two separate chunks (with two corresponding chunk nodes at the top level). The chunk nodes at the top level are linked up with the (micro)features at the bottom level. Furthermore, the action rule may be assimilated into the bottom level of the ACS (into its implicit reactive routines). This is a common, ever-present part of learning for humans in a sociocultural environment, and it is often intertwined with autonomous learning.

Assimilation of externally provided explicit knowledge also occurs within the NACS. For instance, externally provided information (e.g., from a parent) may be: "knives are made of metal." This information is then set up at the top level of the NACS as an associative rule: "if knife, then metal-object." Two chunks involved, one representing the condition and the other the conclusion, are also established and connected to (micro)features at the bottom level of the NACS (which give the concepts/symbols their meanings). In this case, while the chunk for "knife" may have already been established in the ACS and thereafter transferred into the NACS, the concept for "metal-object" may be new and therefore established in the form of a chunk (with a chunk node at the top level). The associative rule may then be used to train the bottom level for top-down learning.

In the process, externally provided symbols, extracted symbols (from the bottom level of the NACS), and transferred symbols (from the ACS) interact within the NACS. For instance, external symbols may be subsumed by extracted symbols, or vice versa, and thereby the two kinds of symbols are related to each other and as a result enhance each other. As an example, an externally provided symbol may denote a concept "gang rivalry," and this concept (symbol) may be subsumed by (i.e., considered a subcategory of) the internally extracted concept (symbol) "potentially violent situation".

For another example, imagine that the child was told that knives were made of metal, while he previously extracted the explicit associative rule

within the NACS that knives were hard objects. So he infers that metal objects are hard objects (or vice versa) within the NACS.

For yet another example, suppose an externally provided symbol and a transferred symbol overlap to some extent but are not identical (hence a conflict) within the NACS. It is wise to reconcile the difference in some way, for example, by creating a new symbol (concept) that subsumes both, or by modifying one of the two to make the two symbols more different from each other (or more similar). There are many other possibilities as well.

As a result of such interactions, the conceptual system of the individual (with symbolic and subsymbolic representation, in the ACS and the NACS) becomes richer and more complex. There is a chance that some of the enriched representation may spread culturally and thereby enrich culture-wide representation and conceptual systems.

The upshot is that not all concepts and symbolic representations that one has are acquired externally (culturally). Likewise, not all concepts and symbolic representations that one has are acquired individually (autonomously). What is learned individually (autonomously) interacts with culturally prevalent symbols, concepts, and representations, as well as being closely related to social interaction. The danger of downplaying the role of autonomous learning and autonomous generation of symbolic representations is that one may end up mistakenly viewing individuals as robots being programmed entirely by the culture in which they found themselves, neglecting other possibilities. The danger of downplaying the role of sociocultural processes in the generation and adoption of symbols and representations is that one may miss an extremely potent force that shapes the individual mind (D'Andrade & Strauss, 1992; Zerubavel, 1997).

3.3.7.3. Learning More Complex Concepts within Clarion

Clearly, the mechanisms and processes of Clarion described above can be applied to the learning of many other types of concepts. Primitive technical concepts, for example, stone tools in the prehistoric ages (e.g., handaxes), can be learned in this way (Sun, 2012).

But what about more complex or more technically sophisticated concepts? I believe that a similar learning process applies also. For example, the Clarion mechanisms and processes may apply to the learning of complex modern technical concepts, including those involved in car driving, airplane piloting, computer programming, emergency response management, and so on.

For instance, with regard to the concept of cars, there can be similar descriptions based on the Clarion mechanisms and processes: embodied implicit reactive routines within the ACS leading to the instinctual action of getting out of the way of a moving car, learning the verbal label denoting cars, refinement of the concept behind the label (e.g., with regard to its features including number of wheels, shape, size, color, and so on, using generalization and specialization), receiving basic driving instructions from external sources, practicing to drive a car through trial and error within the ACS, extraction of explicit knowledge at the top level from implicit knowledge at the bottom level of the ACS, as well as declarative representations including semantic knowledge concerning cars, episodic memory associated with cars, further inferences made, and so on.

However, what about truly abstract concepts? For instance, how can one come up with a concept like "all" (the abstract notion as in mathematical logic) in Clarion? It is a concept that is context-dependent and therefore highly variable in its denotation. It is seemingly difficult to ground out such a concept in simple person-world interaction.

Indeed, there is no easy story to tell about such a concept. In fact, the same goes for many other mathematical-logical notions and other types of abstract concepts, for example, "logical derivation," "syntactic proof," or "topological transformation," among many others. Such a concept would have to piggyback on many layers of abstraction and grounding as a result of long learning processes. For example, a child may first learn the concept of all the people in a room, all the animals in a yard, and so on. Eventually, one extracts the concept of the abstract "all" or is taught by others the concept, which is nevertheless grounded in these more concrete "all" concepts, which in turn are grounded in low-level (micro)features, eventually all the way down to perceptual characteristics. Thus, many layers of abstraction and grounding may be involved in intuitively grasping and interpreting abstract concepts. This is one reason why mathematical logic (and other abstract topics) is normally taught in colleges, not in kindergartens.

Note that a complex or an abstract concept may be grounded in low-level, even perceptual, (micro)features, but by no means is it always defined entirely by these. Its precise definition may have to be explicitly specified through complex symbolic structures at the top level, independent of or in conjunction with implicit representation at the bottom level.

3.4. General Discussion

To recapitulate and to expand on the discussions thus far, below I further address the issues of the two levels and the two learning directions. In addition, I also address a few theoretical controversies related to implicit versus explicit processes.

3.4.1. More on the Two Levels

Why are there two "levels" in Clarion? First, to summarize our discussions earlier, we need the top level in Clarion—psychologically speaking, we need to capture explicit knowledge that humans exhibit (e.g., what they express when they verbalize). The existence of such knowledge is beyond doubt, and the distinction between implicit and explicit knowledge has been amply demonstrated empirically (see, e.g., Reber, 1989; Stadler & Frensch,1998; Seger, 1994; Sun, 2002). Therefore it needs to be captured in some form. Hence there is the top level in Clarion.

The existence of the top level, besides the bottom level, also leads to "synergy" (Sun, Slusarz, & Terry, 2005): that is, better performance under various circumstances due to the interaction of the two levels.

Likewise, we need the bottom level in Clarion. The evidence for the existence of implicit knowledge, as distinct from explicit knowledge that can be easily expressed verbally, is mounting. Although the issue is not uncontroversial, there are nevertheless sufficient reasons to believe in the existence and the significance of implicit knowledge in many cognitive-psychological processes (as variously argued by Reber, 1989; Seger, 1994; Stadler & Frensch, 1998; Sun, 2002; Evans & Frankish, 2009). I argued that implicit knowledge was best captured by neural networks with distributed representation (Sun, 2002; Cleeremans, 1997). Hence there is the bottom level in Clarion. In addition, the bottom level is also important for capturing bottom-up learning.

Furthermore, there are several different kinds of significant differences between the two levels:

- Phenomenological difference: the distinction between the conscious and the unconscious in a phenomenological (first-person, subjective) sense.

- Psychological difference: the distinction between the implicit and the explicit as revealed by experimental work in psychology (e.g., implicit versus explicit learning, implicit versus explicit memory, unconscious versus conscious perception, and so on).
- Implementation difference: for example, the representational difference (symbolic-localist versus distributed representation) between the two levels of Clarion.

In Clarion, the implementation difference leads to accounting for the phenomenological and the psychological difference. So, in this sense, the implementation difference is fundamental to the cognitive architecture in its ability to explain various differences between the two levels.

The idea that both implicit and explicit processes contribute to the mind, as embodied by Clarion, is not a new idea. There have been a few (but rather lonely) voices arguing that point early on. Reber (1989), Mathews et al. (1989), and so on were cited regarding their idea that both implicit and explicit processes contribute to learning and performance. However, the novelty of Clarion in this regard is also evident. The main novel point of Clarion is its focus on the interaction of implicit and explicit processes. The interaction was highlighted in Clarion, (1) in terms of adjustable amounts of contributions from these two types of processes, (2) in terms of their synergy effects (depending on contextual factors), and (3) in terms of their mutual influences during learning (i.e., bottom-up and top-down learning).

Clarion provides some evidence that the interaction between the two types of processes is important. It accounts for a wide variety of empirical data in a coherent, unified way, both quantitatively and qualitatively, based on the interaction between the two types, as will be discussed in subsequent chapters (see also Sun et al., 2001; Sun, Slusarz, & Terry, 2005; Helie & Sun, 2010). In this way, Clarion succeeded in interpreting many empirical findings that had not been adequately explained before and/or captured in computational models before (such as bottom-up learning and synergy effects), and pointed to a way of incorporating such findings into a coherent, unified model (both conceptually and computationally).

Another novel point of Clarion concerns computational modeling of learning: While most models involving implicit/explicit processes did not have detailed (implemented, demonstrated) computational processes

capturing learning of both types, Clarion has, and learning processes within Clarion account for relevant human data.

3.4.2. More on the Two Learning Directions

Let us look into the two directions of learning emphasized in Clarion.

Bottom-up learning is useful. The main advantage of bottom-up learning is that it enables learning in complex domains where there is little or no a priori explicit domain-specific knowledge to begin with. This is because implicit learning is capable of dealing with more complex situations and does not compete much for limited attentional resources. The bottom-up approach makes learning explicit knowledge easier by learning implicit knowledge first, which does not require much attentional resources, and then learning explicit knowledge by utilizing implicit information available to guide (i.e., to narrow down) the search.

Furthermore, there have been human data in the literature that indicate that humans do engage in bottom-up learning (e.g., Stanley et al., 1989; Karmiloff-Smith, 1986; Sun et al., 2001; Sun, 2002; see also Helie et al., 2010). So, bottom-up learning is considered cognitively-psychologically realistic.

Of course, an individual can learn explicit knowledge directly. One does so on many occasions. But there are certain advantages that come with bottom-up learning as opposed to directly learning explicit knowledge. For one thing, as mentioned above, employing this two-step approach may be a more efficient way of learning explicit knowledge, because implicit learning is more suitable for dealing with complex situations and then, guided by implicit knowledge, the search space for explicit knowledge may be narrowed down. This might be one reason why evolution has led to this approach.

But is top-down learning more useful? Practically speaking, maybe this is the case, given culturally created systems of schooling, apprenticeship, and other forms of guided (instructed) learning. Top-down learning is quite prevalent in contemporary society. However, I argue that bottom-up learning is more fundamental. It is more fundamental in two senses: the ontological sense and the ontogenetic sense.[39]

39. In addition, exploration of bottom-up learning may lead to advances relevant to artificial intelligence.

Ontologically, explicit knowledge needs to be obtained by someone in the first place before it can be imparted to others to enable top-down learning. Therefore, bottom-up learning, which creates new explicit knowledge, is more fundamental. Only after bottom-up learning (or other types of learning) has created explicit knowledge, can top-down learning be possible.

Ontogenetically, there seem to be some indications that children learn sensory-motor skills (as well as some other types of knowledge, such as certain types of concepts) implicitly first, and then acquire explicit knowledge on that basis. See, for example, Karmiloff-Smith (1986), Mandler (1992), Keil (1989), and so on for relevant discussions. Therefore, bottom-up learning is also important ontogenetically (developmentally) in many ways.

Given its fundamental nature, bottom-up learning has been emphasized in Clarion (see, e.g., Sun, 2002). An additional reason that bottom-up learning has been emphasized in Clarion is because it was not emphasized sufficiently in the literature (if not neglected altogether). Recently, however, there have been some emerging models of bottom-up learning, besides Clarion, such as Helie et al. (2011).

Top-down learning, on the other hand, has been extensively explored, both empirically and theoretically. For instance, Dreyfus and Dreyfus (1987) explored and analyzed this type of learning in complex skill-learning situations (e.g., in learning to play chess). Relatedly, the ACT-R cognitive architecture has focused mostly on this form of learning (Anderson & Lebiere, 1998). Consequently in empirical work related to ACT-R, top-down learning has been emphasized, and elaborate computational models of top-down learning have been developed.

Top-down learning in Clarion is accomplished naturally, quite different from other computational models of top-down learning. In symbolic cognitive architectures and other symbolic cognitive models, complex symbol manipulations are needed to accomplish top-down learning. In contrast, top-down learning in Clarion is accomplished using the same learning mechanisms as used for implicit learning at the bottom level (without addition or modification). Therefore top-down learning in Clarion is simple and straightforward, without being bogged down by cumbersome mechanisms. It corresponds well to theoretical analysis of such learning (such as Dreyfus and Dreyfus, 1987), and accounts for relevant psychological data (as will be discussed in Chapter 5).

3.4.3. Controversies

Implicit learning and implicit memory are somewhat controversial topics. To base a theoretical framework on implicit learning and memory, it appears that some justifications are needed.

Although implicit learning and memory are somewhat controversial, the existence of implicit processes is generally not in question—what is in question is their extent and importance (Seger, 1994; Stadler & Frensch, 1998; Cleeremans et al., 1998; Sun, 2002; Evans & Frankish, 2009). Clarion allows for the possibility that both types of processes and both types of knowledge coexist and interact with each other to shape learning and performance, so it goes beyond the controversies that focused mostly on details of implicit learning and memory.

For example, some criticisms of implicit learning focused on the alleged inability to isolate implicit processes experimentally. Such methodological problems are not relevant to Clarion, because in Clarion, it is well recognized that both implicit and explicit processes are present in the majority of situations and that they are likely to influence each other in a variety of ways.

Another strand of criticism centered on the fact that implicit learning was not completely autonomous and was susceptible to the influence of explicit cues, attention, and intention. These findings are in fact consistent with the Clarion framework of two interacting levels.

Yet another strand of criticism was about the supposed continuum from the completely explicit to the completely implicit. Judging from empirical data concerning implicit learning and implicit memory, there appears, on the surface at least, indeed a continuum from the completely explicit to the completely implicit, with many shades of gray in between. However, the framework of Clarion, despite its two-level dichotomy, can account for such a continuum.

For example, to account for completely inaccessible (i.e., completely implicit) processes (such as visceral processes), a module within the ACS may be posited that has a bottom level but no corresponding top level. This module may have well-developed implicit processes, but it will never have any corresponding explicit processes.

For another example, at the other end of the spectrum, to account for completely explicit processes, a module may be posited in which there is a top level but no corresponding bottom level. Thus, the module will have explicit processes, but not implicit processes.

In between the two extreme cases, there are modules with both a top level and a bottom level, thus involving both explicit and implicit processes. Some such modules may have a better-developed top level (with rich representational structures and contents) and thus are more explicit, while some other modules may have fewer structures and contents available within their top level and thus are less explicit.

Different degrees of explicitness among different modules may also be determined in part by availability and applicability of algorithms for acquiring (learning) explicit knowledge (such as the RER algorithm) and for applying explicit knowledge. When such algorithms are more available or more applicable, a module may become more explicit.

In addition, the level integration parameters (that regulate the integration of outcomes from the two levels) can be adjusted (e.g., by the metacognitive subsystem) to involve different proportions of explicit and implicit processes during any specific task, which change on the fly the explicitness of a module during task performance.

Generally speaking, controversies surrounding implicit and explicit processes are not as relevant to Clarion as one might believe.

3.4.4. Summary

In summary, the structuring of the ACS and the NACS, each involving both implicit and explicit processes, is cognitively-psychologically justified. Different representations are involved in these processes (implicit versus explicit, and procedural versus declarative). Different types of learning (implicit or explicit) occur. Moreover, bottom-up learning and top-down learning allow implicit and explicit processes to influence each other in learning. Furthermore, learning within the ACS and within the NACS also interact, for example, in the form of transferring knowledge from one subsystem to the other. Clarion captures all of these mechanisms and processes and their interactions.

Appendix: Additional Details of the ACS and the NACS

A.1. Response Time

A.1.1. Response Time of the ACS

The response time (RT) of the ACS is a function of the respective response times of the bottom level and the top level. That is,

$RT = f(RT_{BL}, RT_{TL})$. When stochastic selection of levels is used for determining the final output, RT is determined by the level used to generate the final output.

The response time of the bottom level is determined by: $RT_{BL} = f_{BL}(PT_{BL}, DT_{BL}, AT_{BL})$, where PT_{BL} is the bottom-level perceptual time, DT_{BL} is the bottom-level decision time, and AT_{BL} is the bottom-level actuation (action) time. A simple instance of this function is: $RT_{BL} = PT_{BL} + DT_{BL} + AT_{BL}$.

The response time of the top level is determined by: $RT_{TL} = f_{TL}(PT_{TL}, DT_{TL}, AT_{TL})$, where PT_{TL} is the top-level perceptual time, DT_{TL} is the top-level decision time, and AT_{TL} is the top-level actuation (action) time. For example, $RT_{TL} = PT_{TL} + DT_{TL} + AT_{TL}$.

Often, the response time of the bottom level is faster than that of the top level: specifically, $PT_{TL} \geq PT_{BL}$; $DT_{TL} \geq DT_{BL}$, and $AT_{TL} \geq AT_{BL}$.

For values of these parameters, the following is assumed. First, by default, $PT_{BL} = 200\ ms$, and $PT_{TL} = PT_{BL} + 100ms$.[40] Second, by default, $DT_{BL} = 350ms$.[41] Third, taking into consideration priming of both action rules and action chunks, $DT_{TL} = operation\text{-}time + t_1/rule\text{-}BLA + t_2 / chunk\text{-}BLA$, where *rule-BLA* and *chunk-BLA* are base-level activations discussed before (for the action rule involved and for the action chunk involved, respectively), and *operation-time*, t_1, and t_2 are determined by the mental operation carried out by an action rule (e.g., number of counting steps).[42] Fourth, AT_{TL} and AT_{BL} are variables depending on a host of factors (e.g., response modality and speed). For example, for a verbal response or a mouse click, we may have: $AT_{TL} = AT_{BL} = 500ms$ (Anderson & Lebiere, 1998).

Note that within the ACS, the bottom level is in general faster than the top level. Empirical evidence indicates that implicit procedural processes are often faster (e.g., unconscious perception, reflex

40. Before conscious awareness of perceptual information, a great deal of unconscious preprocessing goes on (Marcel, 1983; Merikle & Daneman, 1998). Therefore, it takes more time to consciously access perceptual information. Relatedly, it was found that for an unconscious idea to become conscious, it takes several hundred *ms* (Libet, 1985). Therefore the default values were set as above.

41. This kind of unconscious decision making is rather direct (from states to actions directly) and therefore fast. This value is roughly based on the data from Libet (1985).

42. This formula includes two kinds of priming: priming of the action rule applied, and priming of the action chunk selected. It is assumed that the response time involving an action rule or an action chunk is proportional to the odds of that rule or chunk being needed based on past uses (Anderson, 1993). Therefore the BLAs are used.

response, and so on; Sun, 2002). But this may not be true for declarative processes.[43]

A.1.2. Response Time of the NACS

At the top level of the NACS, all the applicable associative rules are applied in parallel. So the total associative rule application time is a function of application times of individual associative rules. Application of associative rules may involve the retrieval of result chunks. Therefore, chunk retrieval time may need to be added to associative rule retrieval time. The total time for associative rule application is equal to associative rule retrieval time plus result chunk retrieval time:

$$t_{TL} = h\,(t_a + t_c)$$

where h ranges over all applicable rules, and t_a and t_c are the associative rule retrieval time and the chunk retrieval time, respectively, both resulting from the same rule.

Chunk retrieval time is inversely proportional to the base-level activation of the chunk node in question. That is,

$$t_c = t_3 + t_4 \,/\, \textit{chunk-BLA}$$

where t_c is the chunk retrieval time, and t_3 and t_4 are two constants.

Associative rule retrieval time is inversely proportional to the base-level activation of the associative rule in question. That is,

$$t_a = t_5 + t_6 \,/\, \textit{associative-rule-BLA}$$

where t_a is the associative rule retrieval time, and t_5 and t_6 are two constants.

At the bottom level, the time spent on one iteration of associative mapping is a constant, because implicit processes are viewed as direct mappings. We denote that constant as t_{BL}, the default value of which is 350ms (as previously discussed).

So, if both levels of the NACS are involved, the total time is:

43. The response time difference between the two levels is not intrinsic to their implementation in simulation. Both levels are simulated in Java or C#, and as a result there is no inherent speed difference. For simulation, an internal clock is used, and relevant timing parameters as discussed above can be specified.

$$t_{NACS} = max\ (t_{TL}, t_{BL} \times n)$$

where n is the number of iterations performed by the bottom level of the NACS.

So, if the ACS initiates chunk retrieval, then the time spent by the NACS is determined by t_c. If the ACS initiates associative rules at the top level, the time spent by the NACS is determined by t_{TL}. If the ACS initiates both levels of the NACS, then the time is determined by $t_{NACS} = max\ (t_{TL}, t_{BL} \times n)$.

When the ACS initiates chunk retrieval or associative rule application within the NACS, chunk retrieval or associative rule application is part of the execution of a relevant ACS action. Therefore, in such cases, the time spent by the NACS may be counted as part of the actuation time of the ACS, that is, considered as part of AT_{TL} or AT_{BL}.

A.2. Learning in MLP (Backpropagation) Networks

Learning in MLP (Backpropagation) networks is as usual (see, e.g., Rumelhart et al., 1986 or Levine, 2000). Regardless of whether supervised learning or Q-learning is used, there is always an error measure, although it may be calculated differently. Based on the error measure, assuming a three-layer network is used (consisting of the input, hidden, and output layers), the usual Backpropagation learning rules are as follows.

For adjusting output weights (weights from hidden nodes to output nodes):

$$\Delta w_{ji} = \alpha x_{ji} \delta_j$$

where w_{ji} is the weight associated with output node j from hidden node i, x_{ji} is the input to output node j from hidden node i, α is the learning rate, and

$$\delta_j = err_j o_j (1 - o_j)$$

where err_j is the error measure for output node j, and o_j is the output from output node j (e.g, $o_j = Q\ (x, a_j)$, in case Q-learning is involved).

For adjusting hidden weights (weights from input nodes to hidden nodes):

$$\Delta w_{ji} = \alpha \delta_j x_{ji}$$

where w_{ji} is the weight associated with hidden node j from input node i, x_{ji} is the input to hidden node j from input node i, α is the learning rate, and

$$\delta_j = o_j(1 - o_j)\sum_k \delta_k w_{kj}$$

where o_j is the output from hidden node j, k denotes the nodes downstream in the output layer, w_{kj} is the weight associated with output node k from hidden node j, and $\delta_k = err_k\, o_k\, (1 - o_k)$.

For further technical details, see the companion technical book. See also Rumelhart et al. (1986) or Levine (2000).

A.3. Learning in Auto-Associative Networks

In the Hopfield-type auto-associative attractor network specified earlier (NDRAM; Chartier & Proulx, 2005), upon the presentation of an activation pattern and after p iterations of settling within the network, weights are adjusted as follows:

$$w_{ij[k+1]} = \zeta w_{ij[k]} + \eta\left(\overline{x}_i \overline{x}_j - x_{i[p]} x_{j[p]}\right)$$

where $w_{ij[k]}$ is the weight between nodes i and j at time k ($w_{ij[0]} = 0$), $x_{i[p]}$ is the activation of node i after p iterations (by default, $p = 1$), η is the learning rate (by default, $\eta = 0.001$), and ζ is a memory efficiency parameter (by default, $\zeta = 0.9999$).

In the equation above, $\overline{x}_i$ is the output of the vigilance module:

$$\overline{x}_i = z\frac{(\mu x_{i[0]} + x_{i[p]})}{1 + \mu} + (1 - z)x_{i[0]}$$

where $x_{i[0]}$ is the initial activation of node i (before the p iterations), μ is a free parameter that quantifies the effect of the initial activation (by default, $\mu = 0.01$), and z is defined by:

$$z = \begin{cases} 1, \text{ if } \sum_{i=1}^{N} x_{i[0]} x_{i[p]} \left(\sum_{j=1}^{N} x_{i[0]}^2 \sum_{j=1}^{N} x_{i[p]}^2 \right)^{-1/2} > \rho \\ 0, \text{ Otherwise} \end{cases}$$

where $0 \leq \rho \leq 1$ is the vigilance parameter (Grossberg, 1976). In words, $z = 1$ if the correlation between the initial activation and the final activation is higher than ρ and zero otherwise.

Thus, (1) the initial activation is learned, if the correlation between the initial and the final activation is low (which suggests a new and different activation pattern), and (2) a weighed average between the initial and the final activation is learned, if the correlation is high (which suggests a variation of an already learned activation pattern).

Note that the learning algorithm above is online: that is, learning occurs each time a stimulus is presented to the model. See also Rumelhart et al. (1986) and Grossberg (1988).

A.4. Representation of Conceptual Hierarchies

Here is a quick discussion of representation of conceptual hierarchies within the NACS. First, conceptual hierarchies can be captured through similarity-based reasoning within the NACS, as explained earlier. According to the *reverse containment principle* (Sun, 1994), in the ideal case, if chunk i represents a category that is a superset of the category represented by chunk j, all the (micro)features of chunk i are included in the (micro)features of chunk j. For example, chunk i represents the category "bird" while chunk j represents the category "sparrow." The feature-based description of "sparrow" would naturally include the feature-based description of "bird," plus additional features unique to sparrows.

This kind of "flattened" representation can fully capture conceptual hierarchies; in other words, it can accomplish whatever explicit hierarchical representation can accomplish. For example, flattened representation can capture so-called inheritance hierarchies and inheritance-based reasoning (Sun, 1993, 1994; Sun & Helie, 2013). In fact, the reverse containment principle allows for a natural explanation of inheritance-based inferences. For example, once the chunk node representing "sparrow" is activated at the top level, due to similarity (through top-down and bottom activation flows), the chunk node representing "bird" will also be activated at the top level. Then from explicit rule-based reasoning, the chunk nodes at the top level representing the characteristics of birds will be activated, indicating that they are applicable to "sparrow". In this way,

"sparrow" inherits the properties of "bird." Cancellation of inheritance is also possible (for details, see Sun, 1993, 1994; Sun & Helie, 2013; Helie & Sun, 2014b).

However, in addition to, or instead of, flattened representation based on the reverse containment principle as sketched above, explicit hierarchies, when necessary, may be represented as well (Licato et al., 2014b). Two basic elements are needed for this representation: abstract relations and instantiated relations. For example, abstract relations may be: *ISA(x, y)*; instantiated relations may be: *ISA(sparrow, bird)*. Each is represented by a chunk node at the top level, and by a set of (micro)feature nodes at the bottom level. The (micro)feature set of *ISA(sparrow, bird)* includes the (micro)feature set of *ISA(x, y)*, based on the reverse containment principle. In addition, the (micro)feature set of *ISA(sparrow, bird)* includes the (micro)feature sets of *sparrow* and *bird*, plus some other (micro)features (e.g., those that help to indicate the ordering of *sparrow* and *bird* in this relation).

So, with such explicit representation using chunk nodes, there can be explicit associative rules connecting related chunk nodes at the top level: for example, "if sparrow ISA(x, y), then bird", "if robin ISA(x, y), then bird", "if sparrow bird, then ISA(sparrow, bird)", "if bird REVERSE ISA(x, y), then robin", "if bird REVERSE ISA(x, y), then sparrow", "if ISA(sparrow, bird) SECOND, then bird", and so on. Such rules together constitute explicit conceptual hierarchies. Inheritance reasoning, for example, may be carried out using these rules.

The two kinds of conceptual representations above, flattened (implicit) and explicit, can work with each other. For instance, when the two chunk nodes for *ISA(x, y)* and for *sparrow* are both activated, their (micro)feature nodes are activated at the bottom level. These activated (micro)features nodes will in turn, via the bottom-up activation flow, activate related chunk nodes (applying similarity-based reasoning as a result of feature overlapping). In particular, *ISA(sparrow, bird)*, *ISA(sparrow, animal)*, and the like are activated strongly (but not fully), more strongly than, say, *ISA(robin, bird)*, *ISA(bird, animal)*, *ISA(table, furniture)*, and so on.

Many kinds of inferences can be carried out on the basis of such representation (with chunk nodes, rules, and microfeatures). Exact logical reasoning can be carried out, including propositional logics and many forms of first-order logics (Sun, 1994). Inexact reasoning can also be

performed, which includes: reasoning based on surface similarity (as discussed earlier; Sun & Zhang, 2006), reasoning based on structural similarity (i.e., analogy; Licato et al., 2014), metaphor (Sun, 1995b), inheritance reasoning (Sun, 1993), fuzzy rule-based reasoning (Sun, 1994), and so on. I will not get into further details of these here; the interested reader should refer to these publications cited above for further details.

4

The Motivational and Metacognitive Subsystems

4.1. Introduction

There may arguably be two kinds of control present within the mind: the primary control of actions, and the secondary control of the action decision making per se. To accomplish the latter, motivational and metacognitive mechanisms and processes, among others, are needed (Sun, 2007b; Simon, 1967; Wright & Sloman, 1997).

To counteract the tendency of overspecialization and fragmentation of fields, important elements of the mind such as motivations and metacognition should be incorporated into cognitive science, rather than being excluded. Translating into architectural and mechanistic terms theoretical hypotheses regarding the relationships among cognitive, metacognitive, motivational, and other aspects of the mind can be useful. In so doing, a more complete picture of the mind may emerge.

For an individual, to survive and to function well in the world, behavior must have certain necessary characteristics (Sun, 2003; Sun, 2007b).

For example, among others, the following considerations need to be addressed:

- Sustainability: One must attend to essential needs, such as hunger and thirst, and also know to avoid dangers and other negative situations (Murray, 1938).
- Purposefulness: One must be able to choose actions to enhance sustainability (instead of, e.g., randomly; Tolman, 1932; Hull, 1951; Toates, 1986).
- Focus: One must be able to focus activities with respect to specific purposes. That is, actions need to be somehow consistent, persistent, and contiguous, with respect to purposes (Toates, 1986; Tyrell, 1993). However, one also needs to be able to give up some activities when necessary (temporally or permanently, e.g., when a much more urgent need arises; Simon, 1967).
- Adaptivity: One must be able to adapt (i.e., to learn) for the sake of ensuring and improving sustainability, purposefulness, and focus (Hull, 1951; Timberlake & Lucas, 1989).

To address these considerations, motivational and metacognitive mechanisms and processes are needed. For instance, motivational mechanisms are needed to address sustainability and purpose. Motivational dynamics is essential for human (or animal) behavior, and it is ever-present—"Man is a perceptually wanting animal", as Maslow (1943) put it. Maslow (1943) also argues that "the situation or the field in which the organism reacts must be taken into account but the field alone can rarely serve an exclusive explanation for behavior. . . . Field theory cannot be a substitute for motivation theory." Motivation, as has been conceived thus far (by Maslow and others), needs to be operationalized—to be expressed in mechanistic and process terms.

On the other hand, metacognition refers to "one's knowledge concerning one's own cognitive processes and products, or anything related to them" (Flavell, 1976). Metacognition also includes "the active monitoring and consequent regulation and orchestration of these processes in relation to the cognitive objects or data on which they bear, usually in the service of some concrete goal or objective." Metacognition is needed for the sake of focus and adaptivity as mentioned above (and ultimately for the sake of sustainability and purpose). The notion also needs to be operationalized.

In the remainder of this chapter, the motivational and the metacognitive subsystem of Clarion are presented. Section 4.2 addresses the motivational

subsystem. In Section 4.2, the essential considerations for motivational representations, mechanisms, and processes are discussed first; then, details of motivational representations, mechanisms, and processes are described, followed by a description of the structure of the motivational subsystem. Section 4.3 addresses the metacognitive subsystem. Some essential considerations in this regard are discussed, followed by details of various metacognitive mechanisms and processes. The appendix at the end of this chapter contains some further technical details (including learning and adaptation).

4.2. The Motivational Subsystem

4.2.1. Essential Considerations

The motivational subsystem (the MS) is concerned with why an individual does what he or she does in any situation at any point in time. Simply saying that an individual chooses actions (within the ACS) to maximize reinforcement or rewards leaves open the question of what determines reinforcement or rewards. The MS provides the context in which the goal and the reinforcement (of the ACS) are determined. The relevance of the MS to the main part of the cognitive architecture, the ACS, lies exactly in the fact that it provides that necessary context. It thereby influences the working of the ACS (and by extension, the working of the NACS).

A bipartite motivational representation is in place in the MS, similar to dual-representational structures found elsewhere in Clarion. The explicit goals, such as "*find food*" (which is essential to the working of the ACS as explained before), may be generated based on the internal activations of drives such as "*hunger*." The explicit representation of goals derives from, and hinges upon, implicitly generated drive activations. See Figure 4.1 for a sketch of the MS.

The issue of explicit versus implicit motivational representations needs some examination. On the one hand, empirical and theoretical arguments point to the relevance of explicit representation of goals (see, e.g., Kanfer & Ackerman, 1989; Newell, 1990; Anderson & Lebiere, 1998). On the other hand, the internal processes of drives, needs, or desires are often not explicit and not readily accessible consciously (Hull, 1943; Murray; 1938; Maslow, 1943). It seems reasonable to assume that (1) the idea of dual representation is applicable here (Sun, 2002) and (2) relatedly, implicit motivational processes are more essential than explicit ones (Sun, 2009). Let us look into some details.

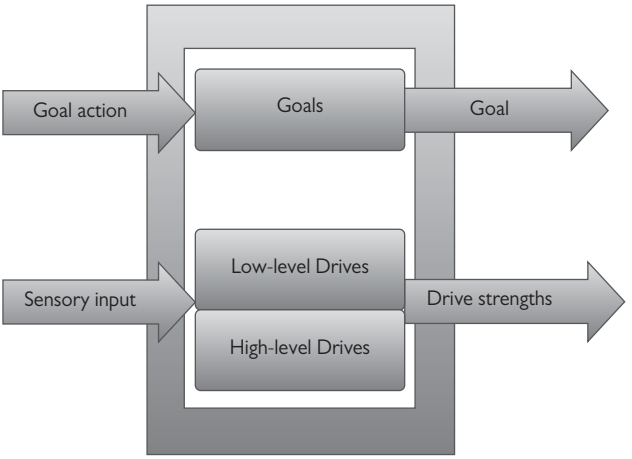

Figure 4.1. The basic structure of the motivational subsystem.

Explicit goals provide specific, definite motivations for actions. While implicit motivational states may change from moment to moment, explicit goals are more persistent and longer lasting. In many circumstances, persistence is needed (discussed later). Furthermore, it is sometimes necessary to compute a match of a state of the world to a goal, so as to discern the progress in achieving the goal and to generate reinforcement signals accordingly. This match is facilitated by using an explicit representation of goals. In addition, explicit goals facilitate explicit cognitive processes to work on these goals and their attainment (in addition to involving implicit processes). Explicit goals allow more behavioral flexibility and formation of expectancies (Epstein, 1982).

However, the more fundamental part of motivation is implicit, consisting of basic drives, basic needs, basic desires, intrinsic motives, and so on (Hull, 1951; Murray, 1938; Maslow, 1943). Human motivational processes cannot be captured by explicit goal representation alone (cf. Anderson & Lebiere, 1998 or Rosenbloom, Laird, & Newell 1993), because, for one thing, they are known to be highly complex and varied (see, e.g., Weiner, 1992). For instance, the interactions of motives, especially their combinations, often require a more complex representation (McFarland, 1989; Tyrell, 1993). Their changes over time, which are often gradual and dynamic, also require a more quantitative, graded representation. Moreover, Maslow (1943) and Murray (1938) specifically discussed the unconscious characteristics of "needs". Given all of

the above, it is natural to hypothesize that implicit motivational processes are necessary.

Furthermore, implicit motivational processes are also fundamental (Sun, 2009). Only on the basis of implicit motivational processes, explicit goal representations arise, which clarify implicit motivational dynamics. Castelfranchi (2001), for example, discussed such implicit-to-explicit motivational "emergence," in ways analogous to general implicit-to-explicit cognitive "emergence" (as more broadly discussed in preceding chapters; see Sun et al., 2001). Maslow (1943), Tolman (1932), and Deci (1980) also emphasized the fundamental role of implicit motives.

Empirical evidence from social psychology also points to the duality of human motivation. For example, Wood and Quinn (2005) explored the relationship between implicit and explicit motivation, in ways analogous to the analysis of implicit and explicit cognitive processes in Sun et al. (2005). Strack and Deutsch (2005) expressed similar views and described what I have termed top-down and bottom-up influences (implicit motivation affecting explicit motivation and vise versa). Woike (1995) showed how implicit and explicit motives might have different effects.

Implicit motives have been variously referred to as basic drives, basic needs, basic desires, intrinsic motives, and so on. I have been referring to them all as "drives" (Sun, 2003, 2009). In the past, Hull (1951) developed the most detailed conception of "drives"—an implicit, preconceptual representation of motives. In his view, drives arose from need states, behaviors were driven so as to eliminate need states, and drive reduction was the basis of reinforcement. Although Hull's conception of drive had significant explanatory merits, his theory failed to capture many motivational phenomena—the variety of different motivations proved too difficult to be encompassed by his theory. A more general notion is therefore needed.

Here I adopt a generalized notion of "drive," different from the stricter interpretations (e.g., as physiological deficits that require to be reduced by corresponding behaviors). With this notion, drives denote internally felt needs of all kinds that likely lead to corresponding behaviors, regardless of whether the needs are physiological or not, whether the needs are reduced by corresponding behaviors or not, or whether the needs are for end states or for processes. Therefore, it is a generalized notion that transcends controversies surrounding the stricter notions of drive. This notion is adopted to account for implicit, context-dependent, and complex drivers of behavior, as well as other properties mentioned early on.

For even more theoretical or empirical background, see Sun (2009). For related views and models, see Hull (1943, 1951), Murray (1938), Maslow (1943), Wright and Sloman (1997), Doerner (2003), Reiss (2004), and Bach (2009).

4.2.2. Drives

Based on these considerations, a set of essential drives is posited within Clarion, termed "primary drives", which includes both low-level and high-level ones, to be detailed below.

4.2.2.1. Primary Drives

Primary drives are those drives that are essential to an individual and are most likely evolutionarily acquired (genetically hard-wired) to a significant extent to begin with (Murray, 1938; Reiss, 2004; Ryan and Deci, 2000; Sheldon, 2011).

Low-level primary drives include

- *food*
- *water*
- *reproduction*
- *sleep*
- *avoiding danger*
- *avoiding unpleasant stimuli*

and so on. (For more on these, see Murray, 1938; McDougall, 1936; and so on.)[1]

Beyond such low-level primary drives, concerning mostly physiological needs, there are also high-level primary drives, which are more socially oriented, concerned mostly with social interaction.[2] Based on the literature on human motivation (e.g., James, 1890; McDougall, 1936;

1. Some of these drives (e.g., "*food*") may have intrinsic positive rewards associated with them, and thus they are mostly aimed at obtaining positive rewards. Some other drives (such as "*avoiding danger*") are mostly aimed at avoiding negative rewards or punishments. See the discussion of approach-oriented versus avoidance-oriented drives.

2. Note again that a generalized notion of "drive" is adopted, different from the stricter interpretations of drives (e.g., as physiological deficits that require to be reduced by corresponding behaviors).

Murray, 1938; Maslow, 1987; Reiss, 2004, 2008; Sun, 2009), the following high-level primary drives are posited:

- *Affiliation and Belongingness*. According to Murray (1938), it denotes the need to "form friendships and associations. To greet, join, and live with others. To co-operate and converse sociably with others. . . . To join groups." It is essentially the same as the need for social contact as termed by Reiss (2004). It is also similar to the notion of belongingness as proposed by Maslow (1987). As Maslow (1943) put it, it denotes "our deep animal tendencies to herd, to flock, to join, to belong". This drive apparently varies across species—not all species have an equally strong need for social belongingness. See also Ryan and Deci (2000).
- *Recognition and Achievement*. It is the need to "excite praise and commendation. To demand respect. To boast and exhibit one's accomplishments. To seek distinction, social prestige, honours or high office", and to "overcome obstacles, . . . to strive to do something difficult as well and as quickly as possible" (Murray, 1938). Murray referred to these tendencies as the need for superiority. Maslow (1943) claimed that "all people . . . have a need or desire for a stable, firmly based, usually high evaluation of themselves, for self respect or self esteem, and for the esteem of others". It is the desire for competence, adequacy, and being recognized for such. See also Ryan and Deci (2000).
- *Dominance and Power*. According to Murray (1938), it denotes the need to "influence or control others. To persuade, prohibit, dictate. To lead and direct. To restrain. To organize the behaviour of a group." It encompasses the notion of dominance proposed by Murray (1938), as well as the notion of power proposed by Reiss (2004).
- *Deference*. "To admire and willingly follow a superior. . . . To co-operate with a leader. To serve gladly" (Murray, 1938).
- *Autonomy*. According to Murray (1938), it is the need to "resist influence or coercion. To defy an authority or seek freedom in a new place. To strive for independence." It was also emphasized by Ryan and Deci (2000). Like many other drives, it varies across species and individuals—not everyone has an equally strong need for autonomy (or deference).

- *Similance*. "To empathize. To imitate or emulate. To identify oneself with others. To agree and believe" (Murray, 1938).
- *Fairness*. One seeks fairness in social interaction, for example, as documented by evolutionary psychologists (Barkow et al., 1992). The notions of reciprocal fairness and inequity aversion have been explored (e.g., Fehr and Gintis, 2007). Fairness is also related to the notion of vengeance by Reiss (2004). Vengeance, and some other related tendencies (such as gratitude), may be derived from the drive for fairness.[3]
- *Honor*. It denotes the desire to obey a moral or cultural code (Reiss, 2004). See also the need for blame-avoidance in Murray (1938).
- *Nurturance*. It is the need to "mother" a child as well as the need to help the helpless (Murray, 1938). It is related to the "need for family" proposed by Reiss (2004).
- *Conservation*. "To arrange, organize, put away objects. To be tidy and clean," and to "collect, repair, clean and preserve things" (Murray, 1938). It is related to the notion of order and the notion of saving in Reiss (2004).
- *Curiosity*. It is the need to "explore. . . . To ask questions. To satisfy curiosity. To look, listen, inspect" (Murray, 1938). It is the desire for knowledge (Reiss, 2004).

On the basis of these ideas from Murray, Maslow, Reiss, and others, the primary drives in the motivational subsystem of Clarion, both low-level and high-level, are summarized in Table 4.1.

This set of primary drives has been explored in a series of prior writings (e.g., Sun, 2003, 2009), and justified based on existing work in social psychology as well as in ethology.[4] For further details, see Sun (2009), as well as

3. The fairness drive is often for getting intrinsic positive rewards (e.g., feeling good), instead of for avoiding punishments by others (e.g., when the lack of fairness is noticed). Likewise, vengeance (as part of the fairness drive) is also often for getting intrinsic positive rewards (e.g., feeling good), not necessarily directly for preventing future unfairness by others. For example, it was found that the reward system was activated if one was given the opportunity to punish those who cheated (de Quervain et al., 2004).

4. Briefly, this set of primary drives is highly similar to Murray's (1938), with only a few differences (e.g., the drive for conservation covers both the need for "conservance" and the need for order proposed by Murray). Likewise, compared with Reiss (2004), one can see that they are similar but with some differences. In addition, Schwartz's (1994) 10 universal values bear some resemblance to these drives; each of his values may be

Table 4.1. A list of primary drives and their brief specifications.

Drives	Specifications
• *Food*	The drive to consume nourishment.
• *Water*	The drive to consume liquid.
• *Sleep*	The drive to rest.
• *Reproduction*	The drive to mate.
• *Avoiding danger*	The drive to avoid situations that have the potential to be harmful.
• *Avoiding unpleasant stimuli*	The drive to avoid situations that are physically (or emotionally) uncomfortable or negative in nature.
• *Affiliation and belongingness*	The drive to associate with other individuals and to be part of social groups.
• *Dominance and power*	The drive to have power over other individuals.
• *Recognition and achievement*	The drive to excel and be viewed as competent.
• *Autonomy*	The drive to resist control or influence by others.
• *Deference*	The drive to willingly follow or serve a person of a higher status.
• *Similance*	The drive to identify with other individuals, to imitate others, and to go along with their actions.
• *Fairness*	The drive to ensure that one treats others fairly and is treated fairly by others.
• *Honor*	The drive to follow social norms and codes and to avoid blames.
• *Nurturance*	The drive to care for, or attend to the needs of, others who are in need.
• *Conservation*	The drive to conserve, to preserve, to organize, or to structure (e.g., one's environment).
• *Curiosity*	The drive to explore, to discover, and to gain new knowledge.

Murray (1938), Maslow (1987), and Reiss (2004). Research has shown their relevance in many domains. For example, Sun and Wilson (2014b) showed their relevance to personality and personality disorders. Sun (2013) showed their relevance to moral judgment. See also Reiss (2008).

4.2.2.2. Secondary Drives

While primary drives are more or less evolutionarily hard-wired (i.e., innate) and relatively unalterable, there may also be "derived" drives.

derived from some primary drives. So, prior work for justifying these frameworks may be applied, to a significant extent, to justifying this set of drives (McDougall, 1936; Murray, 1938; Maslow, 1987; Reiss, 2004; Sun, 2009).

They are secondary, more changeable, and acquired mostly in the process of pursuing the satisfaction of primary drives.

Derived drives include: (1) gradually acquired drives, through some kind of "conditioning" (Hull, 1951), or (2) externally set drives, through externally provided instructions or reinforcement (from individuals or institutions). For example, due to the transfer of the desire to please a superior into a specific desire to conform to his or her instructions, following a certain instruction may become a derived drive. Ryan and Deci (2000) and Weinstein, Przybylski, and Ryan (2013), for example, investigated the internalization of motives.

4.2.2.3. Approach Versus Avoidance Drives

Drives are also roughly divided up into approach-oriented drives and avoidance-oriented drives, as indicated in Table 4.2.

Some researchers (e.g., Gray, 1987) have argued that underlying extroversion is a behavioral approach system (BAS) and underlying neuroticism is a behavioral inhibition system (BIS). Others have similarly argued for approach and avoidance systems (e.g., Clark & Watson, 1999; Cacioppo, Gardner, & Berntson, 1999; Smillie, Pickering, & Jackson, 2006). The BAS is sensitive to cues signaling rewards, and results in active approach. The BIS is sensitive to cues of punishment, and results in avoidance, characterized by anxiety or fear. The division between approach-oriented and avoidance-oriented drives provides an underlying structure for the division between the BAS and the BIS. The reader is referred to the appendix for more detailed justifications for the division between approach-oriented and avoidance-oriented drives.

Table 4.2. Approach-oriented versus avoidance-oriented drives.

Approach Drives	Avoidance Drives	Both
Food	*Sleep*	*Affiliation and belongingness*
Water	*Avoiding danger*	*Similance*
Reproduction	*Avoiding unpleasant stimuli*	*Deference*
Nurturance	*Honor*	*Autonomy*
Curiosity	*Conservation*	*Fairness*
Dominance and power		
Recognition and achievement		

4.2.2.4. Drive Strengths

The activation of drives, as well as the resulting drive strengths, needs to be examined. Drive activation should be orchestrated to ensure the survival and functioning of an individual in a complex world, by meeting a variety of crucial needs of the individual. For one thing, it should reflect internal as well as external conditions that an individual faces.

A set of essential considerations concerning drive strengths has been identified (Tyrell, 1993; Sun, 2003, 2009):

- Proportional activation. The activation (i.e., the strength) of a drive should be proportional to the corresponding perceived deficit in the aspect relevant to the drive (such as "*food*" or "*water*").
- Opportunism. Opportunities need to be taken into consideration. For example, the availability of water may lead to preferring drinking water over eating food (provided that the food deficit is not too much greater than the water deficit).
- Contiguity of actions. There should be a tendency to continue the current action sequence, rather than switching to a different one (e.g., to avoid the overhead of switching). In particular, actions to satisfy a drive should persist beyond minimum satisfaction (i.e., beyond a level of satisfaction barely enough to reduce the strength of the most urgent drive to be slightly below those of the other drives). For example, one should not run to a water source and drink only a minimum amount, and then run to a food source and eat a minimum amount, going back and forth.
- Interruption when necessary. However, when a much more urgent drive arises (such as "*avoiding danger*"), actions for a lower-priority drive (such as "*sleep*") can be interrupted.
- Combination of preferences. The preferences resulting from different drives should be combined to generate an overall preference for a certain course of action (i.e., a certain action goal). In this way, a compromise candidate might be generated that is not the best for any single drive but the best in terms of the combined preference.

Let us see how these considerations can be fulfilled. The first two considerations together point to the use of products, such as "*FoodDeficit* ×

FoodStimulus", in determining the strengths of drives, which take into consideration both deficit and availability (Tyrell, 1993).

The third consideration necessitates a persistent goal structure, as has been argued earlier, which can be set and then persist unless interrupted by a much more urgent drive (such as "*avoiding danger*" when a severe danger is close by). In this way, we may avoid "thrashing": switching back and forth among two or more alternative tasks that are demanded by drives with roughly comparable strengths, while preserving the possibility of interruption when a much more urgent need arises, as dictated by the forth consideration.

Combination of preferences, when deciding on a goal, is an issue that deserves some thoughts. It is believed that combination should be carried out by the resemblance of a multivote system whereby a goal emerges from tallying the votes by different drives (Tyrell, 1993). The problem with the single-vote approach is that only the top-priority goal of each drive is taken into consideration, but lesser goals may be ignored, which may nevertheless make excellent compromise candidates (Sun, 2009).

4.2.3. Goals

On the basis of drives, a goal may be set, which is more explicit and more specific when compared with drives. Drive activations provide the context within which explicit goals are created, set, and carried out. Goals can be set by the metacognitive subsystem (the MCS) based on drive activations (Simon, 1967).

Briefly, a goal structure resides at the top level of the MS, which consists of a number of goal slots, each of which can hold a goal along with its parameters. These goals compete to be the current (active) goal (e.g., based on a Boltzmann distribution of the BLAs of the goals; Sun, 2003). The current goal chosen from those is then considered in action decision making by the ACS.

Some goals may be exactly focused while others may be broader (relatively speaking). In fact, there can be hierarchies of goals in terms of their specificity (Carver & Scheier, 1998). Goals can also be categorized in various other ways. For example, some goals may be approach-oriented while others avoidance-oriented (as discussed earlier; see also Sun & Wilson, 2014b). Goals may also be inward, outward, or a combination thereof. And so on.

Although the most essential way of goal setting is accomplished by the MCS based on drive activations, goals, especially those highly specific action goals, may also be created and set on the fly by the ACS during its action decision making (especially for dealing with sequential actions). For example, goals may be set to emulate a stack-like recursive action structure: last in first out (i.e., the last goal set is addressed first, and the rest of the goals are dealt with in a similar fashion, recursively). Internal actions are available in the ACS for setting and removing goals for such purposes.

Goals are different from drives in many respects. For instance, there may be multiple drives being activated at the same time (e.g., being hungry and being thirsty at the same time). However, there is usually only one goal being pursued at a time (Newell & Simon, 1972; Rosenbloom et al., 1993), although a goal can include multiple parameter dimensions (Sun, 2003). Drives are often more diffused in terms of focus, while goals are often more specific (McFarland, 1989). Drives are more implicit, while goals are more explicit (Murray, 1938; Maslow, 1943; Hull, 1951). Drives are often hard-wired, while goals are often more flexibly created, set, and carried out (Hull, 1951; Sun, 2009).

4.2.4. Modules and Their Functions

Internal processing of drives within the MS involves the following closely interacting modules.

4.2.4.1. Initialization Module

The initialization module carries out the following two mappings:

(a) type of a person → $deficit_d$ (for each d)

that is, the mapping from a person type (a description of personal characteristics, e.g., personality traits; Sun and Wilson, 2014b) to the initial deficit of each drive. Drive deficits determine the inclination of activating corresponding drives. (The initial deficits as determined above can later be changed.)

(b) type of a person → $baseline_d$ (for each d)

that is, the mapping from a person type to the baseline strength of each drive.

For obvious reasons, these two mappings are usually performed only once for an entire simulation. More details of these two mappings can be found in Sun and Wilson (2011), Sun, Wilson, and Mathews (2011), and Sun and Wilson (2014, 2014b), in relation to capturing and explaining human personality types. See Chapter 6 regarding personality modeling.

4.2.4.2. Preprocessing Module

For each drive d, there is a "preprocessor" that picks out relevant information for determining the drive-specific stimulus level, which is an evaluation of the current situation with regard to a particular drive (to be used in calculating the drive strength). That is, this module performs the following mapping:

state $x \rightarrow stimulus_d$ (for each d).

This mapping represents a kind of built-in detector for relevant information in relation to a particular drive.[5]

4.2.4.3. Drive Core Module

This module generates drive strengths (drive activations) based on the following (Sun, 2009):

$$ds_d = gain_d \times stimulus_d \times deficit_d + baseline_d$$

where ds_d is the strength of drive d, $gain_d$ is the gain parameter for drive d,[6] $stimulus_d$ is a value representing the relevance of the current situation to drive d, $deficit_d$ indicates the perceived deficit in relation to drive d (which represents an individual's inclination toward activating drive d), and $baseline_d$ is the baseline strength of drive d. So, the

5. This mapping may include generalizations from some familiar scenarios to other scenarios, accomplished, for example, by using neural networks or similarity-based reasoning.

6. The term "gain" is borrowed from electrical engineering where it indicates the factor by which voltage may be increased in an amplifier or other devices. Note that $gain_d$ can be decomposed (Read et al., 2010): $gain_d = g_d \times g_s \times g_u$, where g_d is the individual gain for drive d, g_u is the universal gain affecting all the drives, g_s is the gain affecting all the drives of one type (e.g., approach-oriented or avoidance-oriented). Averaging of the two g_s parameters can be used to handle those drives that are both approach and avoidance oriented.

drive strength is determined jointly by situational factors and internal inclinations.[7]

The justifications for the mappings performed within the preprocessing module and the core module may be found in a variety of literatures, ranging from ethological research and modeling (Toates 1986; McFarland, 1989; Tyrell, 1993) to cognitive-psychological research and modeling (e.g., Sun, 2009). In particular, the multiplicative combination of $stimulus_d$ and $deficit_d$ has been discussed earlier (see also Sun, 2003, 2009; Tyrell, 1993).[8]

4.2.4.4. Deficit Change Module

This module determines how $deficit_d$ changes, in relation to input states encountered (including goals adopted), actions performed, and so on. That is, it performs the following mapping:

state x, action $a \rightarrow$ change of $deficit_d$ (for each d)

See the appendix for further discussions of this mapping.

Each of these four modules above is implemented as a neural network, for example, a Backpropagation network (or simply implemented as a lookup table; Sun, 2003).

4.3. The Metacognitive Subsystem

The existence of a variety of drives and many possible goals resulting from them leads to the need for metacognitive control and regulation. Metacognition refers to one's knowledge (implicit or explicit) concerning one's own cognitive processes and their outcomes, and monitoring and regulation of these processes (through goal setting and other means; Flavell, 1976). It has been studied extensively in cognitive psychology (e.g., Mazzoni & Nelson, 1998; Reder, 1996) as well as in some other fields.

7. Note that drive strengths could be a function of the equation above. In the simplest case, an identity function is assumed, as shown above.

8. Note that $gain_d$, $deficit_d$, and $baseline_d$ are treated as inputs to the core module because they may be set and tuned by processes outside the core module. In this regard, $stimulus_d$ may also be tunable (i.e., learnable). Learning is dealt with in the appendix.

In Clarion, the metacognitive subsystem (the MCS) is closely tied to the motivational subsystem (the MS). The MCS monitors, controls, and regulates for the sake of goal setting and goal achievement (Simon, 1967; Wright & Sloman, 1997). Control and regulation include setting goals (which are then used by the ACS) on the basis of drives, setting essential parameters of the ACS and the NACS (on the basis of drives and goals), interrupting and changing ongoing processes in the ACS and the NACS, and so on. Control and regulation are also carried out through determining reinforcement for the ACS (on the basis of drives and goals).

4.3.1. Essential Considerations

Some essential characteristics of the MCS should be discussed first. In the literature, metacognition has often been conceived as separate, specialized mechanisms for the sole purpose of monitoring, regulating, and controlling regular cognition. Moreover, metacognition has often been portrayed as explicit processes that involve (exclusively or mostly) deliberative reasoning.

However, metacognition may not be entirely explicit. Reder (1996) argued that it was often implicit, for avoiding using up limited cognitive resources (such as attention) and interfering with regular processes. But is metacognition separate and standalone? Although some of the metacognitive functions can conceivably be separate and standalone, it may not be necessary to posit a strict separation with regard to all of the metacognitive functions (including monitoring, controlling, verbal reporting, and goal setting). Occam's razor seems to be pointing in the opposite direction: Metacognition may not be completely separate and may share resources with regular processes (Sun & Mathews, 2012).

There is indeed some evidence that supports this conception. For example, Reder and Schunn (1996) interpreted their experimental results as indicating that feeling-of-knowing (FOK) judgments were implicit, because both experimental and simulation results indicated that FOK was an error-prone associative process, which did not involve detailed analysis of terms involved. Likewise, Metcalfe (1986) discovered the failure of the "warmth" judgment (feeling of closeness to solutions) in predicting how close a subject was to solving a problem. In a related vein, Glenberg, Wilkinson, and Epstein (1982) argued against the view that subjects were engaged in explicit monitoring of their own performance.

It might be hypothesized that it was difficult to get an accurate metacognitive assessment because metacognition was largely implicit (Sun & Mathews, 2012).

In terms of the relationship between metacognitive and regular processes, data from Glenberg et al. (1982) argued against completely separate metacognitive processes. Rather, metacognitive judgments often fell out as a result of processing stimuli. There was also evidence suggesting that there were different types of metacognitive functionalities and they might be intertwined with regular processes in different ways (Sun, 2003).

Thus, it may be hypothesized that metacognitive processes are neither necessarily explicit, nor necessarily implicit. They may be a combination of both, the same as regular processes. Furthermore, (explicit or implicit) metacognitive processes may not be completely separate from regular processes (either explicit or implicit).

4.3.2. Modules and Their Functions

Structurally, the metacognitive subsystem is divided into a number of relatively independent functional modules. These modules include

- the goal module
- the reinforcement module
- the processing mode module
- the learning selection module
- the reasoning selection module
- the input filtering module
- the output filtering module
- the parameter setting modules

and so on. See Figure 4.2 for a sketch. Below, I will look into a few key functional modules of the MCS.

4.3.2.1. Goal Module

Goal selection (on the basis of drives) is accomplished by this module. The module determines the new goal based on the strengths of the drives, along with the current input state and other information. The new goal is then placed in the goal structure (see Chapter 3). The bottom level of this module may rely more on implicit information (e.g., information

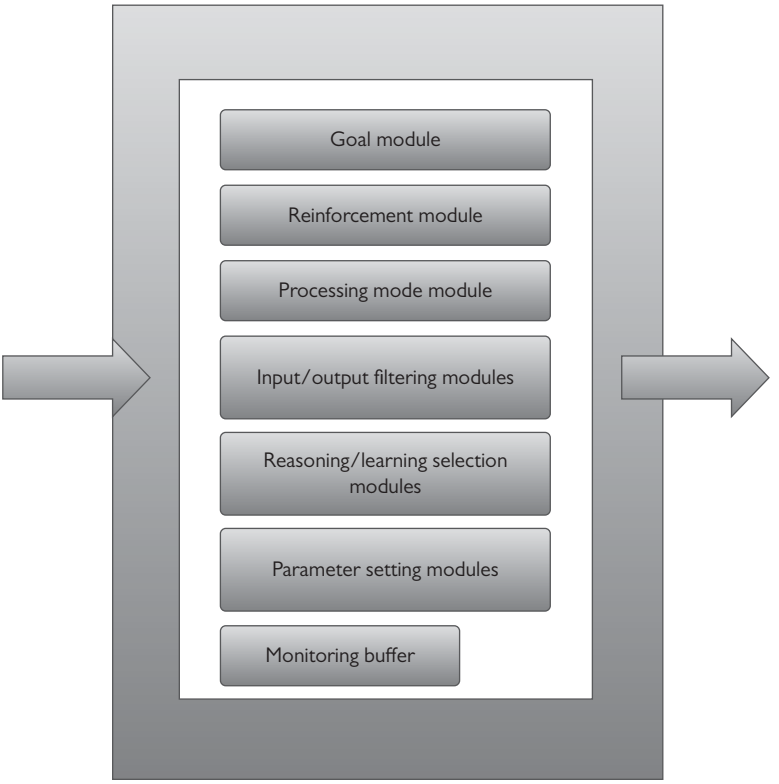

Figure 4.2. The major modules within the metacognitive subsystem.

about drives) in recommending goals, while the top level may rely more on explicit information.

The notion of goal setting on the basis of implicit motives (i.e., drives in Clarion) is supported by the arguments from, for example, Tolman (1932) and Deci (1980). There are also related empirical findings such as Over and Carpenter (2009) and Elliot and Thrash (2002).

Note that two possible ways in which goal selection might be carried out (Tyrell 1993; Sun, 2003) are:

- *Balance-of-interests*: Each drive votes for multiple goals, with different numerical values (representing different degrees of preference). The votes from different drives are tallied, and the goal that receives the highest total votes becomes the new goal.
- *Winner-take-all*: In this case, drives compete and one drive wins the competition. The new goal is chosen for the sole

purpose of addressing the winning drive (representing only its preference).

Among the two, the multivote approach is preferred, because it allows multiple preferences and different degrees of desirability to be taken into consideration. The approach satisfies the consideration regarding "combination of preferences" as discussed earlier.

For instance, for selecting a new goal, the goal module first determines goal strengths for some or all of the goals, based on information from the MS (the drive strengths and the current goal) as well as the current sensory inputs. The following calculation is performed:

$$gs_g = \sum_{d=1}^{n} relevance_{s,d\to g} \times ds_d,$$

where gs_g is the strength (activation) of goal g, $relevance_{s,d\to g}$ is a measure of how relevant drive d is to goal g, which represents the support that drive d provides to goal g (possibly taking into account the current input state s), and ds_d is the strength of drive d as determined by the MS. Once calculated, the goal strengths are turned into a Boltzmann distribution and the new goal is chosen stochastically from that distribution.

An issue here is when to select a new goal. A number of possibilities exist, taking into consideration the requirements of persistence and contiguity and of prompt interruption when necessary (as discussed in 4.2). For instance, as the default, a persistence factor is used:

$$gs_g^{all}(t) = \rho \times gs_g^{all}(t-1) + (1-\rho) \times gs_g(t)$$

where $gs_g^{all}(t)$ is the overall strength of goal g (which encourages persistence by favoring the current goal) to be used for selecting the new goal, $gs_g(t)$ is the current one-step goal strength calculated from the formula specified earlier, $gs_g^{all}(t-1)$ is set to 1 (if goal g was chosen at time $t-1$) or 0 (if goal g was not chosen at time $t-1$), and ρ is the persistence factor (between 0 and 1). The overall goal strength takes into account not only the currently calculated strength but also the previous goal choice. It satisfies (to some extent) the requirements of persistence, contiguity, and prompt interruption when necessary.

In Clarion, the role of goal setting is not limited to the MCS. Goals can also be set by the ACS for the sake of coordinating its actions (see Chapter 3).

4.3.2.2. Reinforcement Module

The reinforcement module produces an evaluation (feedback). For instance, it produces an evaluation of the current input state (and/or the current action) in relation to the current goal and the current activations of drives—whether the result satisfies the goal and the activated drives (a binary output), or alternatively, how much it satisfies the goal and the activated drives (a graded output). The (environmental and internal) sensory information, the activated drives, the goal, and so on are inputs to the module. This module maps them to a value that is used as reinforcement (e.g., used in the ACS for reinforcement learning; see Chapter 3).[9]

Within Clarion, reinforcement is determined from measuring the degree of satisfaction of the activated drives and the current goal, with regard to input state information (and/or actions). That is, reinforcement is an evaluation of the input state (and/or the action), in terms of whether the result helps to address the currently activated drives or not, and whether it satisfies the currently active goal or not (either a binary or a graded output). One possibility is as follows (Sun & Fleischer, 2012):

$$r = DOS_g(s,a) + \sum_d ds_d \times DOS_d(s,a)$$

where $DOS_g(s,a)$ measures the degree of satisfaction of the current goal g by the result of input state s and action a (performed within the input state), ds_d is the strength (activation) of drive d, $DOS_d(s, a)$ measures the degree of satisfaction of drive d, and d ranges over all drives.[10]

Approach-oriented drives may contribute to positive rewards; avoidance-oriented drives may contribute to punishments (Higgins, 1997). Therefore, DOS for an approach-oriented drive may produce a non-negative value (i.e., ≥ 0), while DOS for an avoidance-oriented drive may produce a non-positive value (≤ 0).

9. One problem facing machine learning is coming up with an appropriate reinforcement signal. In general, the world in which an individual lives does not readily provide a simple, scalar reinforcement signal, as often assumed in the literature on machine learning (Sutton and Barto, 1998). Instead, it simply "changes" into a new "state," after an action is performed, which may or may not be fully observable to the individual. Thus an appropriate reinforcement signal has to be determined internally, through synthesizing various sources of information, as sketched above.

10. The result from the equation may need to be scaled to a proper range when neural networks are involved (because outputs from neural networks may be limited to, e.g., [0, 1]). Moreover, DOS_g and DOS_d may need to be scaled separately so as to balance the two measurements.

It is not necessary for the reinforcement module to produce an evaluation for every step. It is only necessary to produce an evaluation for an input state (and/or action) that satisfies (fully or partially) the current goal and/or the currently activated drives in some way. This is because many learning mechanisms (such as Q-learning used in the ACS) can deal with temporal credit assignment. Of course, intermediate evaluations, when available, can be beneficial and help to speed up learning. Intermediate reinforcement may be based on the progress toward an end (Sun, 2002).

The reinforcement module can be implemented either implicitly (at the bottom level of the module) or explicitly (at the top level), or both implicitly and explicitly (at both the top and the bottom level), the same as other modules of the MCS. It is likely that the bottom level relies more on implicit information (e.g., information about drives) in deciding on reinforcement, while the top level may rely more on explicit information.

In biological organisms, there can conceivably be hard-wired "automatic" evaluations and resulting "automatic" reinforcement, such as those concerning satisfying hunger and thirst. On the other hand, for humans and other sufficiently complex organisms, reinforcement signals may (in part) be learned as well, on the basis of interacting with the world (including sociocultural aspects). For example, for humans, complex high-level reinforcement may be learned from sociocultural sources, for the sake of evaluating complex sociocultural situations. One may even learn to adjust (to some extent) the evaluation of simple, direct, bodily states.[11]

4.3.2.3. Processing Mode Module

The processing mode module determines how much each level of the ACS should be used for action decision making: that is, how explicit (or implicit) the ACS processing should be, or what the level of "cognitive control" (as termed by some) should be.

For instance, this module may directly specify the weights or the probabilities to be used within the ACS for cross-level integration. For another instance, probability matching, which was discussed in Chapter 3, is also

11. In Clarion, pretraining or online learning of reinforcement is possible. See the appendix. Imitative learning may also be relevant in this regard (Sun, 2003).

carried out by this module (see Section 3.1.4 for details). Beyond these, there are also a number of other mechanisms within this module related to determining processing modes. I will describe that of the inverted U curve below.

Explicitness of processing within the ACS (degree of "cognitive control") can be determined by avoidance-oriented drives (Wilson, Sun, and Mathews, 2009, 2010). As discussed in Wilson et al. (2009, 2010), avoidance-oriented drive strengths, which capture anxiety levels in a sense, are used to determine the likelihood of performing a task in a more explicit or a more implicit way in the ACS. The hypothesis in this regard is that when anxiety is at an elevated but relatively low level, it helps to increase explicitness in action decision making (which is more effortful but more accurate, and thus appropriate for this situation). However, when anxiety reaches an even higher level, it begins to impair explicit processing. In the latter situation, there is an evolutionary advantage in favoring faster and more automatic (more implicit) processes (e.g., to facilitate a quick escape). To represent this phenomenon, an inverted U curve is used (cf. Yerkes & Dodson, 1908; Hardy & Parfitt, 1991). Therefore, to determine the proportion of explicit versus implicit processing in the ACS, this module maps avoidance-oriented drive strengths to degree of explicitness of processing (e.g., the probability that the top level of the ACS will be used when performing a task), based on an inverted U curve.[12]

For instance, the following parabolic equation leads to an inverted U curve: $y = -0.38x^2 + 0.20x + 0.58$, where x is the maximum avoidance-oriented drive strength (the maximum of all avoidance-oriented drive strengths) and y is the degree of explicit processing of the ACS (Wilson et al., 2009). Figure 4.3 shows this equation graphically. With this equation, the curve begins at $x=0$ below the peak point of the curve, which represents the degree of explicitness when the maximum avoidance-oriented drive strength is very low. As the drive strength increases (i.e., as anxiety increases), the degree of explicitness goes up but then goes down, following roughly an inverted U curve.[13]

The results from the inverted U curve alter the parameters used by the cross-level integration method of the ACS (as discussed in Chapter 3,

12. Note that in terms of the relationship between anxiety and avoidance motivation, this approach disagrees with, for example, Humphreys and Revelle (1984).

13. This equation serves as an example only. The curve may vary in accordance with situational or individual differences.

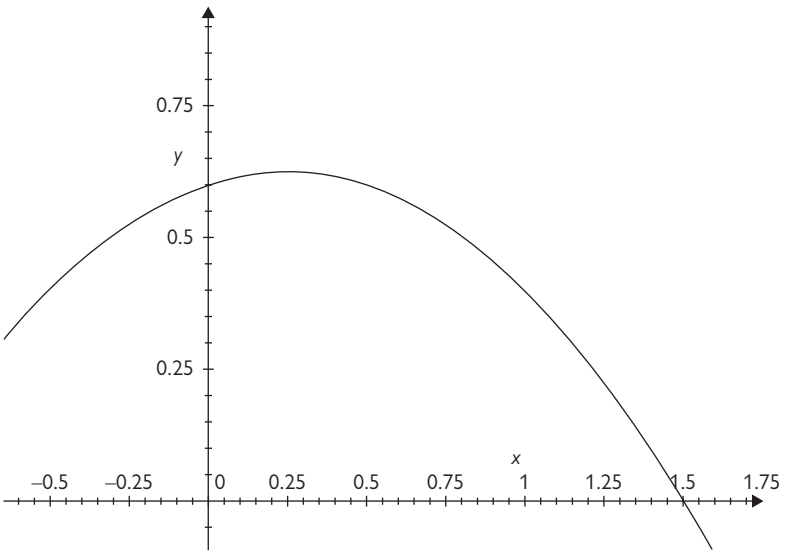

Figure 4.3. The *x*-axis represents the maximum avoidance-oriented drive strength from the MS ($0 \leq x \leq 1$), while the *y*-axis represents the degree of explicit processing determined for the ACS ($0 \leq y \leq 1$).

decided by other mechanisms within this module, e.g., by probability matching). For instance, integration parameters used within the ACS (be it the probability or the weight of the top level) may be modulated multiplicatively by the output from this curve.[14] (Alternatively, the output from the curve may override the values of these parameters.)

Note that this module may designate the top level or the bottom level of the ACS to be used (e.g., by specifying its probability to be 1). For instance, in a routine (highly automatized) situation, one relies almost completely on the bottom level of the ACS.

4.3.2.4. Input/Output Filtering Modules

Attention focusing, either on inputs or on outputs, is carried out by the input/output filtering modules of the MCS. It is accomplished through removing either some specific input/output dimensions or some specific

14. Modulation of integration probabilities of the ACS is accomplished as follows: first, multiply the probability of each explicit component by the value determined from the inverted U curve; then re-normalize all the probabilities (in order for them to sum to 1) by dividing each probability (modified or not) by the sum of all.

values within some input/output dimensions (i.e., regulation of either dimensions or values), with metacognitive actions.

The input-filtering module performs input focusing. Such a metacognitive function is empirically justified. For example, Maner et al. (2005) showed that motives could bias a person's perception.[15] Montgomery et al. (2009) and Epstein (1982) also provided relevant data and arguments. See also the discussion of reconfiguration by Huang and Bargh (2014).

The output-filtering module, as its name suggests, performs output focusing. Mirroring input filtering, this module can likewise be justified.

The information based on which attention is focused lies mainly in the activated drives, the current goal, the current sensory inputs, the working memory, and the ongoing performance of the ACS and the NACS (e.g., as registered in the monitoring buffer of the MCS). External instructions or hints can also impact attention focusing, which are captured through goals or working memory. Based on these sources of information, certain dimensions or certain values are suppressed (accomplished through a zero-activation signal that prohibits the transmission of information through multiplicative connections).

Attention focusing may be carried out differently for different subsystems and/or for different components within a subsystem.

4.3.2.5. Reasoning/Learning Selection Modules

Selection of reasoning methods within the NACS or the ACS is done by the corresponding modules of the MCS, with metacognitive actions that enable certain methods and disable others. The basis for this type of decision, again, lies in the activated drives, the current goal, the sensory information, the working memory, and the ongoing performance of the ACS or the NACS. The selection may be made separately for different components of the ACS and the NACS.

Likewise, selection of learning methods within the ACS or the NACS is carried out by the corresponding modules of the MCS, which enable certain methods and disable others. The basis for the decision consists of the sensory input information, the activated drives, the current goal, the working memory, the monitoring of ongoing performance (e.g., performance improvement or the lack of it, as registered in the monitoring buffer of the

15. For example, activating a self-protection motive led to perceiving more anger in the faces of other people. Activating a mate-search motive led to perception of more sexual arousal in opposite-sex targets (Maner et al., 2005).

MCS), and so on. Selection can be made separately for different components of the ACS and the NACS. For example, possible learning methods for a module in the bottom level of the ACS include: Q-learning, simplified Q-learning, supervised learning, and so on (see Chapter 3).

As in the other modules of the MCS, in these selection modules, it is likely that the bottom level relies mainly on implicit information (e.g., information about drives), while the top level relies more on explicit information (such as goals).

4.3.2.6. Monitoring Buffer

The monitoring buffer is responsible for keeping track of the operations of different components (e.g., different subsystems, and levels and modules within them). It is subdivided into sections: the ACS performance section, the NACS performance section, the ACS learning section, the NACS learning section, and other sections. Each section contains information about both the bottom level and the top level. See, for example, Carver and Sheier (1990) regarding separate monitoring.

For instance, in an ACS or NACS performance section, the information about each component includes the relative strength of the top few conclusions, which indicates how distinguished or certain the top conclusions are in relation to other competing ones. The relative strength is defined as follows:

$$RS = \frac{\sum_{i \in top} S_i}{\sum_j S_j}$$

where the size of the set *top* is determined by a parameter for each component (that is, the number of top conclusions tracked is determined by a parameter for each component), and j ranges over all conclusions from that component.

4.3.2.7. Other MCS Modules

Setting of other major parameters involved in the ACS and the NACS can also be carried out by modules within the MCS. These parameters include, for example, learning rates in the bottom level of the ACS or

the NACS, temperatures in stochastic selection, rule learning thresholds for RER or for IRL, and others. They are set based on the same kinds of factors as enumerated earlier. They may be set separately for different modules within the ACS or the NACS. Different modules within the MCS are responsible for setting different parameters.

4.4. General Discussion

Below, I discuss a number of issues concerning the MS and the MCS. These issues include: reconciling reactive and motivational perspectives (without versus with the MS), scope of the MCS, need for the MCS, and so on.

4.4.1. Reactivity versus Motivational Control

On the one hand, there are reactive accounts of behavior (e.g., Brooks, 1991), which I have been addressing, especially in chapters 1 and 2 (Sun, 2002). On the other hand, there are also motivational accounts of behavior (Toates, 1986; Weiner, 1992; Sun, 2009), which have been covered in this chapter and also need to be taken into account. However, these two perspectives seem polar opposites to each other. Can they be reconciled?

Clarion synthesizes reactive and motivational accounts of behavior. At its core, Clarion is reactive, without the necessity of relying on motivational and metacognitive mechanisms, as argued early on in Chapter 2 (see also Sun, 2002). Reactivity in Clarion is enacted through the ACS, which can run reactively by itself (assuming that inputs from the MS and the MCS are constant, not available, or not used). But, on the other hand, Clarion can also fully incorporate motivational and metacognitive mechanisms and processes, including a bipartite representation of goals and drives as their basis, through the inclusion of the MS and the MCS. Thus, Clarion synthesizes the two worldviews. According to Clarion, which of these two modes, reactive or motivational, dominates is determined by contextual factors in specific circumstances.

4.4.2. Scope of the MCS

One might naturally assume that the metacognitive subsystem, given its name, covers all metacognitive functionalities. In actuality, the MCS

covers only some most essential metacognitive processes. Other "metacognitive" functions (as termed in the literature) may be carried out by other subsystems (such as the ACS). The general use of the term "metacognition" is broader than what is intended here.

For instance, not all selections of methods, strategies, and parameters are carried out by the MCS. Some selections may be carried out by processes within the ACS. For example, when faced with a numerical problem, the decision of "retrieve versus compute" (Reder and Schunn, 1996) may be carried out by a module in the ACS (either explicitly or implicitly), before computation or retrieval is carried out by modules in the NACS. As explained before, the ACS is divided into multiple modules. Some of these modules may be responsible for selecting strategies. There may even be a progressive organization of modules that facilitate increasingly detailed decisions. It may be hypothesized that only selection and regulation at the highest level must be carried out by the MCS. That is, the MCS may be at the top of hierarchical decision making.

In the same vein, some metacognitive information invoked in metacognitive experiments, such as the "feeling of knowing" judgment (e.g., Reder & Schunn, 1996), can conceivably be found in the ACS or the NACS. For example, the feeling of knowing may be assessed in the NACS through similarity-based processes (see Chapter 3). Some other types of metacognitive information, such as the "warmth" judgment (regarding how close one is to a solution), may be registered in the monitoring buffer of the MCS (see Section 4.3.2.6). Metacognitive control and regulation, on the basis of such information, can be carried out by the MCS, or by the ACS (when decisions to be made are at a relatively low level). See Chapter 6 for examples of such scenarios.

Furthermore, it has been claimed that some explicit metacognitive recognition of apparently conscious control ("free will") might in fact be illusory, in that feeling of control might be merely the result of self-interpretation of automatic or even external causes (e.g., Wegner & Wheatle, 1999). Experiments in social psychology have demonstrated that subjects are often not consciously aware of social sources of their own conformity (Nisbett & Wilson, 1977; Wegner & Bargh, 1998). Therefore, explicit metacognitive knowledge involved (if any) is often interpretative.

4.4.3. Need for the MCS

Given the above, one might wonder whether there is a need for a separate metacognitive subsystem. Some might observe that the MCS is similar, mechanism-wise, to the ACS. Since they are similar, can the MCS be considered a part of the ACS, as opposed to a separate subsystem?

At an abstract level, operations of the two subsystems are indeed similar: both involve making action decisions (internal or external) based on input information. However, content-wise, they are different: the MCS is solely concerned with a limited range of metacognitive actions (as described before), while the ACS is concerned with a broader range of actions (see Chapter 3).

If one ignores that content difference, then indeed the MCS may be considered modules of the ACS. However, for the sake of conceptual clarity, it is better to view it as a separate subsystem. Either way of seeing the MCS is fine and does not affect the essential framework of Clarion.

4.4.4. Information Flows Involving the MS and the MCS

Let us take a schematic look at the flow of information among the different subsystems. First, a direct ("reactive") loop is that input state information, including sensory inputs (external or internal), along with the current goal possibly, goes to the ACS as the basis for action decision making; decision making leads to actions, which in turn change the state of the world and thus the inputs. However, a second ("motivational") loop also exists: the input state information also goes to the MS where it triggers drive activations, which in turn trigger the processes of goal setting, reinforcement, and so on within the MCS. The goal, generated by the MCS on the basis of drives (and other information), goes to the MS, and then to the ACS for action decision making (along with other input information). The reinforcement, generated by the MCS on the basis of drives, goals, sensory inputs, actions, and so on, goes to the ACS for adjusting its action decision-making process (and possibly also to other subsystems, e.g., for adjusting goal setting within the MCS).

In addition, the ACS and the NACS interact with each other. Operations in the NACS are directed by the ACS, and results from the NACS are sent to the ACS. The MCS receives information from other subsystems and intervenes in other subsystems, including the ACS and the NACS.

4.4.5. Concluding Remarks

The objective of this chapter (as well as the previous chapter) is the construction of representations, mechanisms, and processes for the sake of explaining a wide variety of cognitive-psychological data in a unified and comprehensive way, even though this chapter (as well as the previous chapter) contained some computational and theoretical speculations.

This chapter addresses motivational and metacognitive representations, mechanisms, and processes, necessary for a comprehensive cognitive architecture. The need for implicit drive representation, as well as explicit goal representation, has been argued. The motivational mechanisms and their resulting dynamics help to make a cognitive architecture more complete and functioning in a more realistic way. On top of that, metacognitive functionalities have also been developed. These developments constitute a requisite step forward in making a cognitive architecture a more realistic model of the human mind taking into fuller considerations its complexity and intricacy.

As a result of these developments, Clarion has the potential to eventually provide a coherent account of a wide range of phenomena in motivation, emotion, metacognition, and personality, as well as many other cognitive-psychological aspects. More explorations of these aspects will follow in Chapter 6.

In addition, Clarion also helps to clarify mechanistically the notions of anxiety (linking it to avoidance-oriented drive activations), cognitive control (linking it to amount of explicit processing), effort (linking it to amount of explicit processing but also to goals), and so on (cf. Humphreys & Revelle, 1984). It also addresses resource allocation and resource availability. Details can be found in simulations in subsequent chapters.

Appendix: Additional Details of the MS and the MCS

A.1. Change of Drive Deficits

A mechanism is needed for adjusting (decreasing or increasing) the *deficit* of a drive that is affected by the current action, the current state, or the current goal. The change of deficits is not detailed in the description of the

MS earlier, because its causes are widely varied: it may be mainly physiologically determined (e.g., for the food deficit) but may also be psychologically and socially determined in a complex way (e.g., for the dominance and power deficit). A deficit change module may need to be specified for a simulation.

In a simulation, for the sake of simplicity, one often assumes that only one "winning" drive that gets to set the "winning" goal is impacted by subsequent actions: it is gradually reduced by the actions guided by the winning goal. This simplification may work in some circumstances. But in general, multiple drives may contribute to setting the winning goal (see the description of goal setting in Section 4.3.2.1), and many of them may be impacted, if the actions performed address these drives (reducing or increasing their deficits). Even other drives that did not contribute to setting the winning goal can also be impacted sometimes in some fashion.

A.2. Determining Avoidance Versus Approach Drives, Goals, and Behaviors

To decide whether a drive in the MS is approach oriented or avoidance oriented, the following principles are adopted:

- The division should be based on seeking positive rewards versus avoiding punishments (or negative rewards; Gray, 1987).
- It does not rely on complex reasoning, mental simulation, or other complex mediating processes, because drive activation is reflexive and immediate (Brooks, 1991; Dreyfus, 1992).
- Some drives may come with intrinsic positive rewards (e.g., *food*, *reproduction*, *recognition and achievement*, and so on—essentially all the approach-oriented drives). Others may not have intrinsic positive rewards (e.g., *avoiding danger*), and are mostly for avoiding negative rewards or punishments. This distinction has been argued before (Gray, 1987; Clark & Watson, 1999; Cacioppo, Gardner, & Berntson, 1999).[16]

16. For example, the *sleep* drive is mostly for avoiding negative signals, and mostly not in anticipation of positive rewards afterward. In contrast, the *food* drive is mostly for getting immediate positive rewards. In addition, some drives, such as *fairness*, have both orientations.

Based on the above, it can be justified one by one why each drive should be approach oriented, avoidance oriented, or both (Sun, Wilson, & Mathews, 2011). Table 4.2 contains the resulting division.

Based on this division of drives, goals and actions (behaviors) can also be classified as either approach oriented or avoidance oriented. Because goals and actions (behaviors) are task-specific, this classification may need to be performed for each simulation. To determine whether a goal is approach oriented or avoidance oriented, its associations to approach or avoidance drives are considered. That is, if a goal is more associated with approach drives as opposed to avoidance drives, it is an approach goal. Specifically, this can be accomplished through summing the strengths of a goal (as determined by the goal strength equation) over all of the scenarios under consideration (for a particular simulation), for approach and avoidance drives respectively (the activations of which are determined by a scenario under consideration). If the sum of strengths for a goal is higher when coupled with approach drives than with avoidance drives, then it is an approach goal. Otherwise, it is an avoidance goal. A goal can be both if the two sums are close (Sun & Wilson, 2014b; Wilson & Sun, in preparation).

Behaviors (actions from the ACS) can also be classified as being either approach oriented or avoidance oriented, based on associations with approach or avoidance goals. This can be accomplished by summing the values of each behavior (as determined by a trained network at the bottom level of the ACS), taken over all of the scenarios under consideration for a particular simulation, and over all approach and all avoidance goals respectively. If the sum of values for a given behavior is higher when coupled with approach goals than with avoidance goals, then the behavior is an approach behavior. Otherwise, the behavior is an avoidance behavior.

A.3. Learning in the MS

When the ACS receives reinforcement (feedback) for selecting and performing actions (in other words, for the mapping: state → action), learning occurs within the ACS to adjust action selection based on the feedback (as discussed in Chapter 3). Similar learning, with the same reinforcement, tunes drive activation within the MS (i.e., tuning the mapping: state → drive strength).

Specifically, to tune drive activation, the MS adjusts the drive gain parameter: $gain_d$ (for any drive d).[17] It uses the same reinforcement signals as used by the ACS. For instance, the following adjustment can be performed:

$$\Delta gain_d = \alpha \times sgn(r) \times ds_d$$

where α is the amount of change, ds_d is the strength (activation) of drive d, and $sgn(r)$ is the sign of reinforcement r. That is, $gain_d$ is increased if positive reinforcement is received when drive d is activated; it is decreased if negative reinforcement is received.

Such tuning, as one might expect, is limited to relatively minor changes: it involves a small learning rate; the value of $gain_d$ is limited by tight upper/lower bounds. Different learning rates may be used in tuning, for example, depending on whether a positive or a negative reinforcement signal is received (i.e., depending on $sgn(r)$). Different learning rates may reflect differential sensitivities to reward and punishment. Alternatively, in case of Q-learning being used in the ACS, r may be replaced by $\Delta Q(s, a)$ in the equation above (where s and a are the state and the action performed). Note that the reinforcement signals received may be socioculturally determined or influenced (Sun, 2001). Therefore, this learning is sociocultural to some extent.[18]

Additionally, tuning of drive deficits ($deficit_d$) within the MS is also possible (e.g., with a very small learning rate, and within tight upper/lower bounds). Because deficits may capture person types (as mentioned earlier; see Sun & Wilson, 2010, 2014), the following mapping has been specified earlier: type of a person → $deficit_d$. Thus, tuning of $deficit_d$ addresses the formation and adaptation of personality type. The tuning equation for this parameter is similar to the one used earlier.

Tuning of drive stimuli ($stimulus_d$) within the MS is also possible (e.g., with a very small learning rate, and within tight upper/lower bounds). This tuning addresses changing sensitivities of different drives to different situations. According to Clarion, for primary drives, $stimulus_d$ is

17. Here $gain_d$ refers mainly to g_d. See the definitions earlier.

18. In case a neural network is used (pre-trained) for implementing the mapping above, the tuning may adjust an input to the network, $gain_d$ (or alternatively, it may adjust the network weights instead).

the result of (mostly pre-formed) detectors for spotting drive-relevant situations (e.g., a detector for potential dangers or for potential mates). Therefore, the mapping, state → $stimulus_d$, can be embodied in a (pre-trained) Backpropagation neural network, and the network can be tuned as usual using the Backpropagation learning algorithm with the output changes determined by a tuning equation, which is similar to the one used earlier.

A.4. Learning in the MCS

A.4.1. Learning Drive-Goal Connections

At the same time as learning occurs in the ACS, tuning may be done in the MCS to strengthen or weaken strengths of particular goals (i.e., the mapping for goal setting: drives, state → new goal). The tuning can be accomplished using the same reinforcement signals as received by the ACS.

For example, in the simplest case, we have

$$gs_g = \sum_{d=1}^{n} relevance_{s,d\to g} \times ds_d,$$

where gs_g is the strength (activation) of goal g, $relevance_{s,d\to g}$ is a measure of how relevant drive d is to goal g in the context of s, and ds_d is the strength of drive d as determined by the MS. Then stochastic selection of a goal may be carried out based on gs_g.

In this case, tuning may be carried out on $relevance_{s,d\to g}$ when goal g is selected:

$$\Delta relevance_{s,d\to g} = \alpha * \ sgn(r) \ \times \ ds_d$$

where α is the amount of adjustment, and $sgn(r)$ is the sign of reinforcement r. That is, $relevance_{d,s\to g}$ is increased if positive reinforcement is received when drive d is activated and goal g is selected; it is decreased if negative reinforcement is received.

As before, different learning rates may be used, for example, depending on the sign of the reinforcement. Note that the reinforcement signals

received may be socioculturally determined or influenced. Therefore, this learning is also sociocultural to some extent.[19]

A.4.2. Learning New Goals

Beside strengthening or weakening the mapping from drives to existing goals, one may also need to learn new goals that did not exist before. New goals are created under two types of circumstances: when a goal has intrinsic connections to certain sequences of actions and when a goal has no such a priori connections. Because this aspect is not involved in any simulation discussed in this volume, I will not get into details here (but see Sun, 2003).

19. In case a neural network is used for implementing the mapping, the learning above can adjust the input to the network, $relevance_{s,d \to g}$ (or alternatively, adjust the weights connecting drives, as well as other inputs when present, to goals).

5

Simulating Procedural and Declarative Processes

In the next three chapters, I will discuss how Clarion may capture, and thereby provide useful interpretations and explanations of, psychological tasks, data, and phenomena, through computational simulation of these tasks, data, and phenomena. Among these three chapters, the present chapter focuses on simulating and explaining psychological processes that may be characterized as (mainly) procedural and/or declarative. The subsequent two chapters will address other types of processes: motivational, metacognitive, social, and so on.

As discussed before, the advantages of detailed computational simulation include the fact that it leads to more substantiated theories and explanations, bringing out detailed psychological processes involved. Computational simulation often requires detailed, mechanistic, and process-based descriptions that may replace (sometimes vague) verbal-conceptual theories with theories of more clarity and precision. Due to the specificity of computational models, a more complete, more precise, and more consistent explanation may be produced that reduces or eliminates inconsistency and ambiguity. Moreover, different explanations embodied by different models may be examined in detail and compared precisely. Therefore, a better understanding of psychological

phenomena may be achieved as a result (Sun, 2007, 2009b; Fum, Del Missier, & Stocco, 2007).

Below, four specific psychological tasks are examined (in sections 5.1–5.4), in addition to examining some general phenomena (in Section 5.5). Note that these specific tasks tend to be small laboratory tasks, chosen for the sake of illustrating the essential mechanisms and processes of Clarion. I deliberately avoided focusing on major theories that resulted from Clarion (such as the theory of synergistic interaction in skill learning or the theory of creative problem solving). For these major theories, the reader is referred to prior publications such as Sun, Slusarz, and Terry (2005), Helie and Sun (2010), as well as Sun (2002). For general psychological phenomena, see Section 5.5 for examples (especially in relation to reasoning).

As mentioned in Chapter 1, when discussing simulations, a balance needs to be struck between conceptual clarity and technical specificity. For the sake of conceptual clarity, a high-level conceptual explanation needs to be provided; for the sake of technical specificity, a computational description also needs to be provided (up to a certain point), corresponding to the high-level conceptual explanation.

However, the descriptions below omit minute technical details (e.g., parameter values), because (1) I intend to focus on broad interpretations of psychological data and phenomena, and thus minute technical details get in the way; (2) even within individual simulation studies, I want to focus on conceptual issues, such as why Clarion provides the right framework for explaining relevant psychological phenomena; minute technical details are not particularly relevant in this regard; (3) computational models are not pure mathematical models, and thus they are harder to describe completely; as a result, minute computational details are often naturally omitted; (4) exact parameter values may change as a result of even minor algorithmic changes, which, however, usually do not change essential conceptual issues.[1]

1. Note that details of statistical analysis are omitted below for essentially the same reasons. Note also that the figures included in the next three chapters are from old sources. As such, their formats are not uniform, their image qualities are often less than ideal, and extraneous information is sometimes included in them. However, they represent historical records in a sense.

5.1. Modeling the Dynamic Process Control Task

Below I will first look into a task that involves implicit procedural learning, as well as bottom-up learning (going from implicit to explicit procedural knowledge). I will examine the impact of the interaction between implicit and explicit processes.

5.1.1. Background

The role of implicit learning in skill acquisition has been gaining recognition in recent decades (Reber, 1989; Proctor & Dutta, 1995; Sun, 2002). Although both implicit and explicit learning have been investigated empirically, the interaction between implicit and explicit learning and the importance of this interaction were not widely recognized. The interaction has traditionally been downplayed in empirical research (but with a few exceptions). Research has been focused on showing the lack of explicit learning in various settings and on the controversies stemming from such claims. Similar oversight was also evident in computational models of implicit learning (but with a few exceptions).

Despite the relative scarcity of studies of the implicit-explicit interaction in skill acquisition, it has become evident that it is difficult to find a situation in which only one type of learning is engaged. Various indications of the implicit-explicit interaction can be found scattered in the literature. For instance, Stanley et al. (1989) found that under some circumstances, verbalization (generating explicit knowledge) could help to improve skill performance. Ahlum-Heath and DiVesta (1986) also found that verbalization led to better performance. However, as Sun et al. (2001) showed, verbalization might also hamper implicit learning, especially when too much verbalization induced an overly explicit learning mode.

Similarly, as shown by Berry and Broadbent (1988), Stanley et al. (1989), and Reber et al. (1980), verbal instructions given prior to skill learning could facilitate or hamper task performance. One type of instruction was to encourage subjects to perform explicit search for regularities that might aid in performance. Reber et al. (1980) found that, depending on the ways in which stimuli were presented, explicit search might help or hamper performance. Another type of instruction was explicit how-to instruction that told subjects specifically how a task should be performed,

including providing information concerning regularities. Stanley et al (1989) found that such instructions helped to improve performance significantly.

In a way, as discussed before, such empirical results indicated the possibility of synergy between implicit and explicit procedural processes, in the sense that under proper circumstances, the interaction of implicit and explicit procedural processes led to better overall performance.

Turning to the relationship between implicit and explicit learning, there was some empirical evidence that implicit and explicit knowledge might develop independently under some circumstances. However, there were also cases where a subject's performance improved earlier than explicit knowledge. For instance, as shown by Stanley et al. (1989), while the performance of subjects might quickly rise to a high level, their verbal knowledge might improve more slowly. Bowers et al. (1990) also showed delayed learning of explicit knowledge. In these cases, due to the fact that explicit knowledge lagged behind but improved along with implicit knowledge, explicit knowledge was in a way "extracted" from implicit knowledge. Learning of explicit knowledge might occur through the explication of implicit knowledge, that is, through bottom-up learning as discussed in Chapter 3 (Sun et al., 2001; Sun, Slusarz, & Terry, 2005).

In the remainder of this section, I will quickly present some data from experiments with process control tasks (Berry and Broadbent, 1984; Osman, 2010). I will then discuss the simulation of the data set and the analysis of the results. The discussions will be drawn from Sun et al. (2007).

5.1.2. Task and Data

The human data from the process control tasks of Stanley et al. (1989) were used. The data were typical of human performance in process control tasks and demonstrated the interaction between explicit and implicit processes in skill learning.

The task setting was as follows: human subjects were instructed to control the outputs of a simulated system by choosing their inputs into the system (from a set of available inputs). The outputs of the system were determined from the inputs provided by the subjects, through a certain relationship. However, this relationship was not known to the subjects. Subjects gradually learned how to control the outputs of the system

through trial and error. Many of them also developed some explicit knowledge of the relationship. Various experimental manipulations of learning settings placed differential emphases on explicit and implicit learning.

The data resulting from the task demonstrated a number of interesting effects of the implicit-explicit interaction, as touched upon earlier: (1) the verbalization effect (i.e., verbalization sometimes led to better performance), (2) the explicit how-to instruction effect (i.e., receiving how-to instructions led to better performance), and (3) the synergy effect (i.e., the enhanced role of explicit processes in the verbalization and the explicit instruction condition led to better performance; Sun, Slusarz, & Terry, 2005; Sun et al., 2007).

Specifically, in Stanley et al. (1989), two versions of process control tasks were used. In the person version, each subject interacted with a computer simulated "person" whose behavior ranged from "very rude" to "loving" (over a total of 12 levels), and the task was to maintain the behavior of the simulated "person" at "very friendly" by controlling his/her own behavior (which could also range over the 12 levels, from "very rude" to "loving"). In the sugar production factory version, each subject interacted with a simulated factory to maintain a particular production level (out of a total of 12 possible levels), through adjusting the size of the workforce (which also had 12 levels). In either case, the behavior of the simulated system was determined by $P = 2 * W - P_1 + N$, where P was the current system output, P_1 was the previous system output, W was the input from subjects to the system, and N was noise. Noise (N) was added to the output of the system, so that there was a chance of being up or down one level (a 33% chance respectively).

There were four groups of subjects. The control group was not given any instruction to help performance and not asked to verbalize. The "original" group was asked to verbalize after each block of 10 trials. Other groups of subjects were given explicit instructions in various forms. To the "memory training" group, a series of 12 correct input/output pairs was presented. To the "simple rule" group, a simple rule ("always select the response level half way between the current production level and the target level") was given. All the subjects were trained for 200 trials (20 blocks of 10 trials).

Statistical analysis was done based on "score," defined as the average number of on-target responses per trial block (where the exact target value plus/minus one level was considered on target). It showed that the score of the original group was significantly higher than that of the

Table 5.1. The human data from Stanley et al. (1989). Each cell indicates the average number of on-target responses per trial block. The exact target value plus/minus one level was considered on target.

	Sugar Task	Person Task
control	1.97	2.85
original	2.57	3.75
memory training	4.63	5.33
simple rule	4.00	5.91

control group. Statistical analysis also showed that the scores of the memory training group and the simple rule group were also significantly higher than that of the control group. See Table 5.1.[2]

Explanation is in order in regard to what the result suggested. First, the performance in this task involved mostly procedural (action-centered) processes, and moreover, mostly implicit procedural processes, judging from many experiments in the past (e.g., Berry and Broadbent, 1988; Mathews et al., 1989; Mathews et al., 2011). Second, the memory training and the simple rule condition led to more involvement of explicit processes, because of the emphasis placed on explicit knowledge in these conditions. Third, verbalization also increased the involvement of explicit processes, because verbalization necessarily placed more emphasis on explicit (verbalizable) knowledge. Thus these three conditions demonstrated the synergy effect (along with the verbalization and the explicit instruction effect). More detailed analysis may be found in Sun et al. (2007).

5.1.3. Simulation Setup

The simulation of this task demonstrated computationally the synergy between implicit and explicit procedural processes, resulting from the interaction of these processes, which led to better overall performance.

In accordance with the analysis earlier, the following was posited: the action-centered subsystem (the ACS) was mainly involved in this task, because this task relied on procedural knowledge (skills). In the

2. Note that subjects performed somewhat better in the person task compared with the sugar factory task. Subjects might have brought in their prior knowledge of interacting with other people in the real world into their performance of the person task.

Table 5.2. The order of IRL rules to be tested. a =1, 2, b = –1, –2, 0, 1, 2, c = –1, –2, 1, 2, P is the desired system output level (the exact target), W is the current input to the system (to be determined), W_1 is the previous input to the system, P_1 is the previous system output level (under W_1), and P_2 is the system output level at the time step right before P_1.

1	$P = aW + b$
2	$P = aW + cP_1 + b$
3	$P = aW_1 + b$
4	$P = aW_1 + cP_2 + b$

bottom level of the ACS, a neural network implemented Q-learning with Backpropagation. Reinforcement was determined by the outcome from the to-be-controlled system, based on the distance between the target value and the actual outcome.[3]

The inputs to the simulated subject included: the target value for the system to be controlled, the action at step $t - 1$, the output from the system to be controlled at step $t - 1$, the action at step $t - 2$, and the output from the system to be controlled at step $t - 2$. The output of the simulated subject consisted of one output dimension of 12 possible actions.[4]

At the top level, two types of rule learning, RER and IRL, were involved. Four sets of IRL rules were involved in the hypothesis testing process (without generalization and specialization), as indicated in Table 5.2. (Note that, if needed, other rule forms could be added easily. Adding more rules would not drastically change the working of the model.) Positivity was measured by whether or not the system outcome was on target—the exact target value plus/minus one (in accordance with the human experiments). It was used for calculating *PM*s and *NM*s for both RER and IRL rules, and also for extracting initial rules in RER. Based

3. For instance, one reward function was: *0.2 * (1 – |actual – target|)*. A few different reward schemes were tested, and essentially the same results were obtained.

4. The encoding at the bottom level was such that each value in each input dimension was represented by an individual node in the input layer of the neural network. The output encoding at the bottom level used a set of nodes, one for each possible action. Thus, 60 input units, 40 hidden units, and 12 output units were involved.

on that, the IG measures for RER and IRL, respectively, were calculated (as detailed in Chapter 3).

For capturing each of the experimental conditions, few parameter values were adjusted. To model the effect of verbalization (in the "original" group), rule learning thresholds were adjusted so as to increase rule learning activities at the top level (i.e., the IRL rule deletion threshold was raised, and the RER thresholds were lowered). The hypothesis was that verbalization tended to increase explicit activities, especially rule learning activities.[5]

To capture explicit instructions, given knowledge was wired up at the top level. In the "memory training" condition, each of the 12 explicit examples was wired up at the top level (in the form of "$P_1 \rightarrow W$"). In the "simple rule" condition, the explicit rule (as described earlier) was wired up at the top level (as a "fixed rule" or FR; see Sun, 2003).

For each group, a total of 100 simulation runs were conducted, representing 100 simulated "subjects". Each run lasted 20 blocks, for a total of 200 trials, exactly the same as in the human experiments.[6]

5.1.4. Simulation Results

The simulation setup as described above captured all the observed effects in the human data (Sun et al., 2007). First, the simulation captured the verbalization effect in the human data, as shown by Table 5.3. Statistical tests compared the simulated "original" group with the simulated control group, which showed a significant performance improvement due to verbalization, analogous to the human data.

The simulation also captured the explicit instruction effect (also shown in Table 5.3). Statistical tests compared the simulated "memory training" and the simulated "simple rule" group with the simulated control group, which showed significant improvements of these two groups over the simulated control group, analogous to the human data.

Together, they captured the synergy effect posited earlier. That is, more involvement of explicit procedural processes, on top of implicit procedural processes, led to better procedural performance.

5. Different from RER, a higher threshold in IRL leads to more rule learning activities.

6. To capture the fact that subjects performed better in the person task compared with the sugar factory task (presumably due to the fact that subjects brought their prior knowledge of interacting with other people in the real world into their performance of this task), some pre-training was conducted prior to performing the person task.

Table 5.3. The simulation of Stanley et al. (1989). Each cell indicates the number of on-target responses per trial block.

Human Data		
	Sugar Task	Person Task
control	1.97	2.85
original	2.57	3.75
memory training	4.63	5.33
simple rule	4.00	5.91
Model Data		
	Sugar Task	Person Task
control	1.92	2.62
original	2.77	4.01
memory training	4.45	5.45
simple rule	4.80	5.65

To better understand contributing factors in the model performance, a componential analysis was performed to tease out the contributions of various constituting components of the model. "Lesion" studies of the full model and tests of partial models were carried out to discover the respective effects of the top level versus the bottom level and RER versus IRL.

A "lesion" study of the full model was performed as follows. After optimizing the parameters of the full model (by trial and error), IRL or RER was removed respectively to form two partial models. With the same setting of parameters (optimized with regard to the full model), the two partial models were applied to the learning of this task.

See Table 5.4 for the "lesion" data. Removing IRL led to far worse performance than removing RER, in terms of the mean squared deviation from the human data. That was probably an indication that IRL contributed more significantly to the performance of the original full model than RER.

Was this effect an artifact of the parameter setting that was optimized with regard to the full model? To answer this question, partial models were also individually optimized and tested through optimizing only those parameters that were applicable to a partial model (in terms of maximizing the match between a partial model and the human data).

With the same training of 20 blocks of 10 trials each, the performance of the resulting optimized partial models was compared with that of the

Table 5.4. The simulation of Stanley et al. (1989) with "lesioned" models. Each cell indicates the number of on-target responses per trial block. "BL" denotes the bottom level of the ACS.

Human Data		
	Sugar Task	Person Task
control	1.97	2.85
original	2.57	3.75
memory training	4.63	5.33
simple rule	4.00	5.91
Model Data (BL+RER)		
	Sugar Task	Person Task
control	1.55	1.89
original	1.60	1.95
memory training	3.77	4.15
simple rule	4.08	4.45
Model Data (BL+IRL)		
	Sugar Task	Person Task
control	2.10	2.65
original	3.45	4.68
memory training	4.71	5.80
simple rule	5.06	6.29

full model. As shown by Table 5.5, (1) the partial model with the bottom level and IRL performed much better than the partial model with the bottom level and RER, but (2) neither performed as well as the full model. This result showed that IRL was more important than RER in matching the human data, but all of the three components were useful in matching the human data—all were necessary in order to maximize the match between human and model performance. See Table 5.5 for the data.

Furthermore, based on each of the two partial models, a complete model was built. For each partial model, all the parameters applicable to the partial model, which were optimized with respect to the partial model, were frozen, and then the missing component was added to complete the partial model, optimizing only those parameters that were applicable to the newly added component. The reason for doing so was to further identify the significance of each component through freezing other components.

Table 5.5. The simulation of Stanley et al. (1989) with optimized partial models. Each cell indicates the number of on-target responses per trial block. "BL" denotes the bottom level of the ACS.

Human Data		
	Sugar Task	Person Task
control	1.97	2.85
original	2.57	3.75
memory training	4.63	5.33
simple rule	4.00	5.91
Model Data (BL+RER)		
	Sugar Task	Person Task
control	1.68	1.81
original	1.64	1.96
memory training	4.23	4.46
simple rule	4.72	4.87
Model Data (BL+IRL)		
	Sugar Task	Person Task
control	2.23	2.76
original	3.43	4.55
memory training	4.55	5.63
simple rule	4.86	5.63

The resulting "partial-full" models were compared with each other and with the original full model. Adding IRL to the partial model with RER and the bottom level led to more improvements than adding RER to the partial model with IRL and the bottom level, which again showed the importance of IRL in this task. See Table 5.6 for the data. As before, this test showed that all the components above were useful in terms of producing a better fit with the human data.

Because there was no human learning curve available, there could not be a comparison between human and model learning curves. Learning curves are therefore not addressed here, but they will be discussed later with regard to other tasks where human learning curves are indeed available.

Variations of the model were also tested, for example, with different input/output encoding, with different implicit learning algorithms, with different rule learning algorithms, and so on. The results were comparable

Table 5.6. The simulation of Stanley et al. (1989) with partial-full models. Each cell indicates the number of on-target responses per trial block. "BL" denotes the bottom level of the ACS.

Human Data		
	Sugar Task	Person Task
control	1.97	2.85
original	2.57	3.75
memory training	4.63	5.33
simple rule	4.00	5.91
Model Data (BL+RER → IRL)		
	Sugar Task	Person Task
control	2.03	2.68
original	2.71	3.92
memory training	4.77	5.45
simple rule	5.32	5.55
Model Data (BL+IRL → RER)		
	Sugar Task	Person Task
control	2.02	2.27
original	2.89	3.90
memory training	4.51	5.43
simple rule	5.05	5.37

to the full model discussed above, also capturing the essential characteristics of the human data (for more details, see Sun et al., 2007).

5.1.5. Discussion

In all, the human data and the simulation of them both confirmed the verbalization effect and the explicit instruction effect. Thus they demonstrated the synergy between implicit and explicit procedural processes. The match between the simulation and the human data was analyzed, and thus the validity of the model was also demonstrated to some extent.

To account for these effects mentioned above, implicit learning, implicit-to-explicit extraction, and explicit hypothesis testing learning were all needed. The model was able to capture these effects because it used explicit rule learning along with implicit learning at the bottom level. Regardless of whether RER or IRL was involved, explicit rule learning led to explicit knowledge, which helped to enhance overall performance.

This simulation demonstrated the usefulness of explicit learning (Sun et al., 2007).

It was also apparent from the simulation that IRL contributed considerably more to the capturing and explanation of the human data than RER. So, the simulation suggested, correspondingly, that it was possible that in human learning of this task, explicit hypothesis testing learning (as captured by IRL) was more important than implicit-to-explicit extraction (as in RER).

However, there was a trade-off to be considered when dealing with the question of which version of the model should be preferred. The full model certainly produced the best fit compared with the partial models, but it was also more complex than the other models. Comparing the full model with the partial models, the question is whether the increased complexity of the full model was worth the increase in accuracy. What is needed here is a measure of complexity-accuracy trade-offs and a criterion for deciding when increased complexity is worth it. Because there is no consensus on a practical formal measure of the complexity-accuracy trade-off, the answer to this question remains a matter of judgment.

There was also the question of why RER helped to improve learning and performance, given that RER rules were extracted from the bottom level in the first place. This question was discussed in detail in Sun and Peterson (1998). The conclusion, based on a systematic analysis, was that the explanation of the synergy between the two levels was based on the following factors: the complementary representations of the two levels, the complementary learning processes, and the bottom-up rule learning criterion used. In other words, although rules were extracted from the bottom level, the very process of extraction and the resulting explicit representation make rules different and useful. Usefulness of rules learned from IRL may also be attributed, at least in part, to these factors, besides the fact that IRL rules often represent different knowledge to begin with.

The Clarion model of process control tasks appears to be more comprehensive than other models of process control tasks. It includes all of the following: implicit learning, explicit hypothesis-testing learning, and implicit-to-explicit extraction. It also includes both instance-based learning and rule-based learning in a unified manner. In this regard, note that the RER learning algorithm starts with extracting concrete instances (from the bottom level) and can be either instance based or rule based, depending on learning parameters within RER. The rule refinement

(generalization and specialization) parameters within RER can either discourage or encourage generalizing concrete instances (extracted from the bottom level) into more abstract rules. Moreover, the model includes both bottom-up and top-down learning. The model can also easily incorporate heuristics hypothesized in some other theories (e.g., Dienes & Fahey, 1995; Fum & Stocco, 2003).

In addition, Dienes and Fahey (1995) showed that either an instance-based or a rule-based model might be appropriate for explaining human performance in this task, depending on task settings. The key was the salience of task stimuli. Instead of having two separate models, the Clarion model encompassed both instance-based and rule-based processes. Therefore, it included the two models of Dienes and Fahey (1995) as two special cases. Taatgen and Wallach (2002) implemented a model in ACT-R using essentially the idea of instance-based learning from Dienes and Fahey (1995). Therefore, the comparison above applied to their model as well.

Of course, only suggestive evidence for the Clarion model has been provided from the work described in this section, not a definitive proof. This work is but one test of Clarion, as part of a much larger project—much more work has been (or is being) conducted.

Follow-up work, in the forms of both human experiments and computational simulations, is needed. Human experiments may attempt to verify

- the relative contributions of different learning processes as revealed by the simulation, and
- the various effects that have been captured in the simulation.

In terms of verifying the various effects of the implicit-explicit interaction, there have been many corroborating results as briefly reviewed earlier (see also Sun, Slusarz, & Terry, 2005). However, further exploration of this aspect may still be worthwhile.

5.2. Modeling the Alphabetic Arithmetic Task

5.2.1. Background

On the basis of the previous section, this section further explores both bottom-up and top-down learning. As originally defined in

Sun et al. (2001), top-down learning goes from explicit to implicit knowledge, while bottom-up learning goes from implicit to explicit knowledge (see Chapter 3). Instead of studying each type of learning (implicit or explicit) in isolation, their interaction in the forms of these two learning directions is explored. This section addresses how one type gives rise to the other and the effects of such interactions on learning, through modeling empirical data.

Furthermore, this section involves both action-centered and non-action-centered subsystems. In a way, this section explores the interaction between procedural and declarative processes, as well as its effects on learning. Therefore, this section presents a simulation that takes into account both implicit and explicit processes, both action-centered and non-action-centered knowledge, and both top-down and bottom-up learning.

Human data in the alphabetic arithmetic (i.e., letter counting) task will be tackled. This section shows how the data may be captured through comparing a variety of approaches. It shows that the quantitative data in the task may be captured more accurately using top-down learning, which constitutes a more apt explanation of the data. This work also shows, in a way, the benefit of including both action-centered and non-action-centered processes in simulating this task. Thus, synergy from the interaction between action-centered and non-action-centered processes can be argued for. These results provide a more integrated perspective on skill learning, incorporating the four-way division of implicit versus explicit and procedural versus declarative processes.

It has been argued that in cognitive science, detailed comparisons of different modeling approaches and choices are needed but lacking (see, e.g., Massaro, 1988; Pew & Mavor, 1998; Roberts & Pashler, 2000). Detailed comparisons may reveal new or better possibilities in terms of theoretical assumptions, modeling frameworks, and algorithmic/computational processes. Therefore, comparisons of models are performed in this section.

Below, I first discuss some human data and then the simulation of the data. The discussion relies on results from Sun, Zhang, and Mathews (2009).

5.2.2. Task and Data

The task of alphabetic arithmetic (letter counting) of Rabinowitz and Goldberg (1995) involved two sets of issues that were of

interest: action-centeredness versus accessibility (explicitness) and top-down versus bottom-up learning. In addition, there had been existing simulations that could be compared to the Clarion simulation. In view of the above, it was an attractive test domain.

The setting of the task was as follows (Rabinowitz and Goldberg, 1995; Johnson, 1998): Subjects (children) were asked to solve alphabetic arithmetic problems of the following forms: $letter_1$ + *number* = $letter_2$, or $letter_1$ – *number* = $letter_2$, where $letter_2$ was *number* positions up or down from $letter_1$, depending on whether + or – was used. Subjects were given $letter_1$ and *number*, and asked to produce $letter_2$.

In experiment 1 of Rabinowitz and Goldberg (1995), during the training phase, one group of subjects, the consistent group, received 36 blocks of training, in which each block consisted of the same 12 addition problems. Another group, the varied group, received 6 blocks of training, in which each block consisted of the same 72 addition problems. While both groups received 432 trials, the consistent group practiced on each problem 36 times but the varied group only 6 times. The addends ranged from 1 to 6.

In the transfer phase of this experiment, each group received 12 new addition problems (not practiced before), repeated 3 times. The findings were that, at the end of training, the consistent group performed far better than the varied group. However, during the transfer phase, the consistent group performed worse than the varied group. The varied group showed almost perfect transfer, while the consistent group showed considerable slowdown. See Figure 5.1.[7]

In experiment 2, the training phase was identical to that of experiment 1. However, during the transfer phase, both groups received 12 subtraction (not addition) problems, which were the reverse of the original addition problems used for training (for both groups), repeated 3 times. The findings were that, in contrast to experiment 1, during transfer, the consistent group actually performed better than the varied group. Both groups performed worse than their corresponding performance at the end of training, but the varied group showed worse performance than the consistent group. See Figure 5.2.

To investigate possible mechanistic (computational) explanations of the data pattern as well as the issues raised earlier, simulations were

7. Note that only correct responses were used in the analysis.

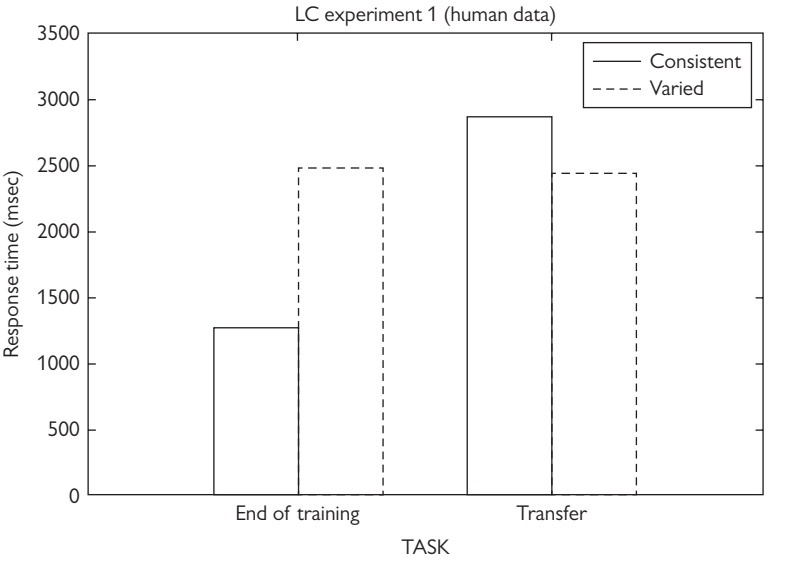

Figure 5.1. The results of experiment 1 of Rabinowitz and Goldberg (1995).

carried out based on a cross-product design: learning method (top-down versus bottom-up versus both) × action-centeredness (the ACS only versus both the ACS and the NACS), for the sake of comparing these alternatives.

5.2.3. Top-Down Simulation

5.2.3.1. Simulation Setup

One simulation was based on top-down learning: that is, a priori action rules ("fixed rules" or FRs; Sun, 2003) were coded at the top level of the ACS to begin with, for capturing prior knowledge concerning counting letters. Then, on the basis of these rules, performance was carried out and implicit learning at the bottom level of the ACS took place.

The set of action rules used included straight counting rules:

If goal=addition-counting, start-letter=x, number=n, then starting with x, repeat n times: count-up

If goal=subtraction-counting, start-letter=x, number=n, then starting with x, repeat n times: count-down

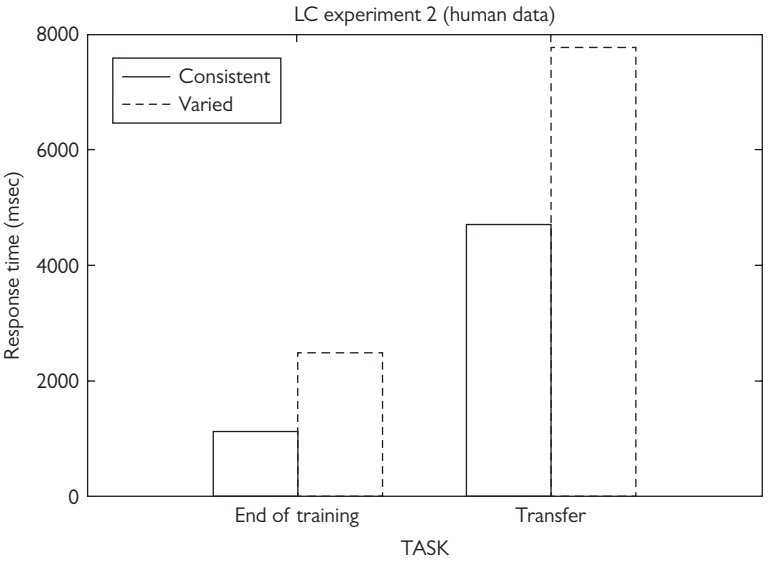

Figure 5.2. The results of experiment 2 of Rabinowitz and Goldberg (1995).

The action rule set also included instance-retrieving rules that generated solutions by using memorized instances (stored in the NACS in the form of chunks):

If goal=addition-counting, start-letter=x, number=n, then retrieve chunks with (dim_1 = x, dim_2 = +, dim_3 = n, dim_4 =?) and report dim_4.

If goal=subtraction-counting, start-letter=x, number=n, then retrieve chunks with (dim_1 = x, dim_2 = −, dim_3 = n, dim_4 =?) and report dim_4.

In these rules, "?" represented "don't care" conditions. These rules above were fairly straightforward.

However, it was also possible to perform "reversed retrieval" to address subtraction problems using addition instances, or vice versa, with the following action rules:

If goal=addition-counting, start-letter=x, number=n, then retrieve chunks with (dim_1 =?, dim_2 = −, dim_3 = n, dim_4 = x) and report dim_1.

If goal=subtraction-counting, start-letter=x, number=n, then retrieve chunks with (dim_1 =?, dim_2 = +, dim_3 = n, dim_4 = x) and report dim_1.

The NACS was used for storing experienced instances. In the NACS, each question and answer pair encountered was encoded as a chunk (with a chunk node at the top level and feature nodes at the bottom level). A retrieval rule from the ACS triggered all the chunks that overlapped in terms of features with the retrieval cue (indicated by the retrieval rule). Actual retrieval was limited to one chunk at each step. All triggered chunks (all those overlapping with the retrieval cue) competed to be the one retrieved, through a stochastic selection process (as discussed in Chapter 3).

All these action rules in the ACS competed based on their utility. In calculating the utility (based on the cost and the benefit of each rule as defined in Chapter 3), the benefit was set equal to the positive match rate of a rule. The rule condition must match the current state to count as a "match." A "positive match" was further determined by the outcome of the matching rule being correct. The cost was set based on the estimated average execution time of a rule.

A goal structure with one goal slot was used. There were two possible goals, *addition-counting* or *subtraction-counting* (as used in the rules above). A goal was set when instructions to count up or count down were given by experimenters.

Three inputs were provided: a starting letter, an arithmetic operator, and a number. In addition, the goal was also input. There were 26 possible outputs, each of which represented a letter. At the bottom level of the ACS, there was one network. Its output indicated the (guessed) target letter.

The response time when a counting rule was applied (without chunk retrieval within the NACS) was in part determined by the BLA of the rule applied (see Chapter 3). The total response time of the top level was the sum of the perceptual time, the decision time of the rule (which was in part determined by the BLA of the rule), the counting time, and the verbal answer time (as described in Chapter 3). The response time of the bottom level may be viewed as a constant (as estimated in Chapter 3).

The response time when a chunk retrieval rule was applied was determined in part by the BLA of the chunk retrieved from the NACS and the BLA of the retrieval rule applied (in the ACS), as described in Chapter 3. The total response time of the top level with chunk retrieval was the sum of the perceptual time, the decision time of the retrieval rule (in part determined by the BLA of the rule), the retrieval time in the NACS

(in part determined by the BLA of the chunk retrieved), and the verbal answer time.[8]

5.2.3.2. Simulation Results

Examine simulation results. First, at the end of the training phase of experiment 1, the simulation matched the response time difference between the consistent and the varied group. The difference was statistically significant in the simulation results, as in the human data. See the simulation data in Figure 5.3, which should be compared with Figure 5.1.

The Clarion simulation provided a plausible explanation of the human data. The simulated consistent group had a lower response time because it had more practice on a smaller number of instances, which led to the better-performing bottom level in the ACS, as well as better-performing instance retrieval from the NACS. The bottom level of the ACS of the simulated consistent group performed more accurately because of more focused practice on a smaller number of instances by the simulated consistent group (compared with the varied group). The NACS of the simulated consistent group was more accurate for the same reason. Thus the bottom level of the ACS and the chunks of the NACS were more likely to be used in determining the overall outcome of the simulated consistent group, due to the competition among different components.[9] Because these two components had faster response times (they were either inherently so, as in the case of the bottom level of the ACS, or due to more frequent use and thus higher BLAs, as in the case of the NACS), a faster overall response time resulted for the simulated consistent group.

Clarion also matched the transfer performance difference between the two groups in experiment 1, as shown in Figure 5.3. During the transfer phase of experiment 1, the performance of the simulated consistent group got worse, compared with its performance at the end of training. The transfer performance of the simulated consistent group was in fact

8. The values of most parameters were set based on estimates from prior simulations (Sun, 2002), except learning rate, temperature, and rule learning thresholds, which were domain-specific and were set to produce the best fit with the human data.

9. The simulation data indeed showed that there were a lot more retrievals from the NACS in the simulated consistent group than in the simulated varied group. The data also showed a higher selection probability for the bottom level of the ACS in the simulated consistent group.

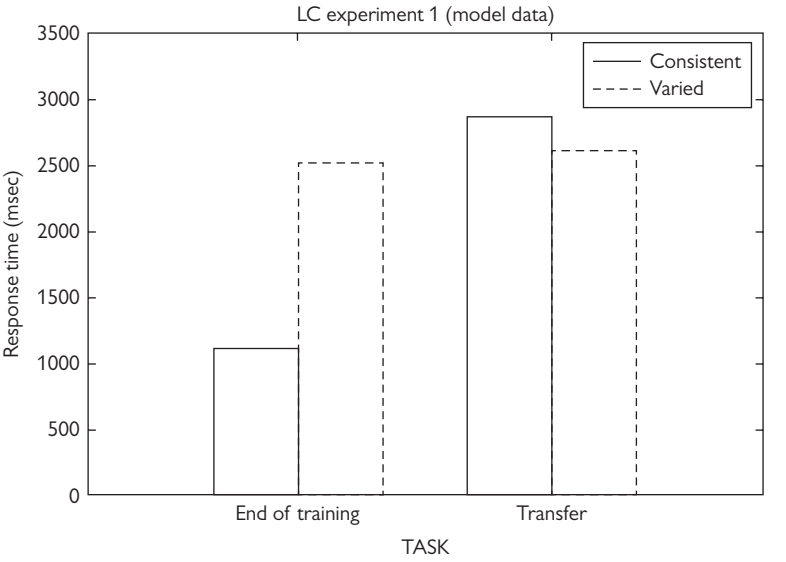

Figure 5.3. The Clarion simulation of experiment 1.

worse than that of the simulated varied group. The differences were statistically significant. These facts were consistent with the human data.

The Clarion simulation provided a plausible explanation of this aspect of the human data as well. The simulated consistent group relied more on the bottom level of the ACS and on the NACS during training and therefore the BLAs of its counting rules were lower. As a result, it took more time to apply the counting rules during transfer, which it had to apply due to the fact that it had to deal with a different set of problems during transfer.[10] The performance of the simulated varied group hardly changed, compared with its performance at the end of training. This was because it relied mostly on the counting rules at the top level during training, which was equally applicable to training and transfer problems. As a result, its counting rules had higher BLAs, and therefore it performed better than the simulated consistent group during transfer.

10. Even though instance retrieval from the NACS and decision making by the bottom level of the ACS might not be appropriate during transfer (because a different set of problems was used), the simulated consistent group was more likely to use them, due to probabilities resulting from probability matching during training (which led to a higher probability for the bottom level of the ACS) and the rule utility measures acquired during training (which led to favoring instance retrieval rules). These two tendencies were, of course, corrected later on through adaptation of these two aspects.

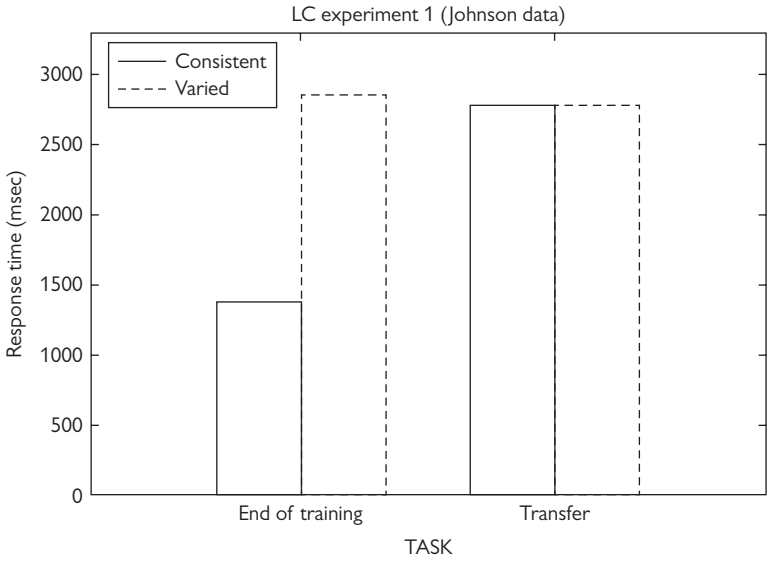

Figure 5.4. The ACT-R simulation of experiment 1 (Johnson, 1998).

The ACT-R simulation (Johnson, 1998) did not explain the human data fully. For instance, it did not capture the fact that the transfer performance of the consistent group was worse than that of the varied group, which was explained in the Clarion simulation by the fact that the varied group had more practice of the relevant rules during training (hence higher BLAs). See Figure 5.4 as a comparison. The separation of implicit and explicit processes in Clarion was also important. If there was no separation of the top and the bottom level in Clarion, even with all the other characteristics of Clarion (such as rule BLAs), there would be no performance difference between the two simulated groups.

Furthermore, this Clarion simulation also captured accurately the human data of experiment 2. The simulation results were as shown by Figure 5.5, which were similar to the corresponding human data in Figure 5.2.

Clarion provided the following explanation in this regard. During transfer in experiment 2, due to the change in the task setting (counting down as opposed to counting up), the practiced rule for counting up was no longer useful. Therefore, both simulated groups had to use a new counting rule for counting down, which had only the initial BLA for both groups. Similarly, both simulated groups might use a new instance

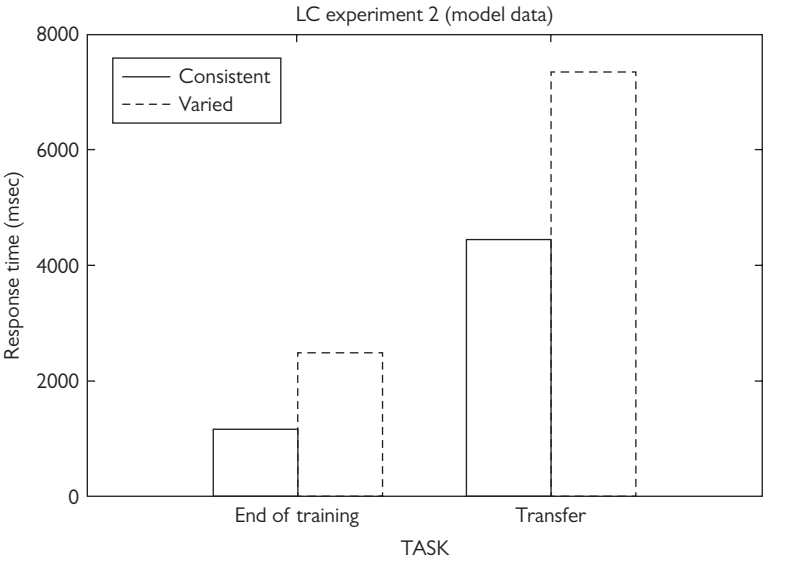

Figure 5.5. The Clarion simulation of experiment 2.

retrieval rule (for reversed retrieval), which also had only the initial BLA. Both simulated groups performed worse than at the end of training for that reason.

This explanation was not offered by the ACT-R simulation (Johnson, 1998). In the ACT-R simulation (see Figure 5.6), the transfer performance of the consistent group hardly changed compared with its training performance. This aspect of the ACT-R simulation was not consistent with the human data.

Moreover, the Clarion simulation captured the fact that the varied group performed worse than the consistent group during transfer (Figure 5.5). In the Clarion simulation, this difference was explained by the fact that the simulated consistent group had more BLAs associated with chunks encoding instances in the NACS than the simulated varied group, because the simulated consistent group had more experiences with these chunks. These chunks were used in reversed retrieval during the transfer phase of experiment 2, because of the reverse relationship between the training and the transfer problems used in this experiment. Therefore, the simulated consistent group performed better than the simulated varied group during the transfer phase.

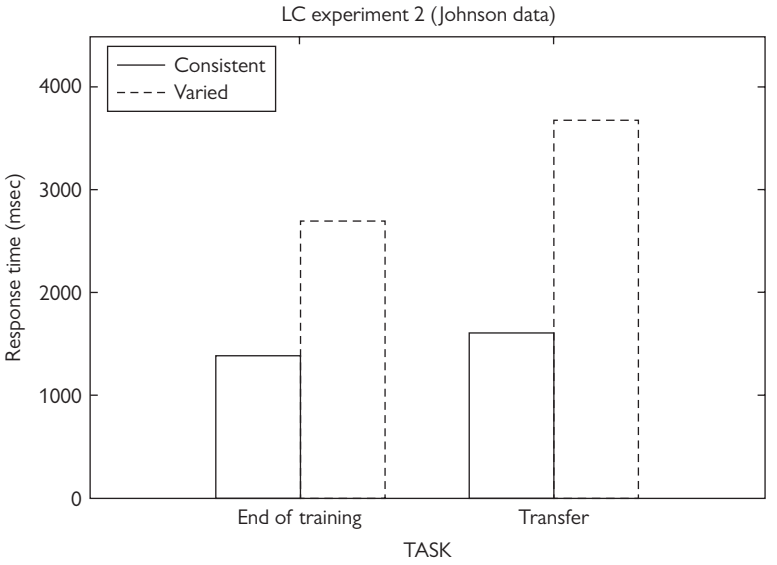

Figure 5.6. The ACT-R simulation of experiment 2.

The learning curves during the training phase of the Clarion simulation were as shown in Figure 5.7. The human learning curves were also included there for comparison. The match of the simulated and the human learning curves was much better than that from the ACT-R simulation (Johnson, 1998).

5.2.4. Alternative Simulations

A number of alternative simulations were also explored with Clarion. In one such alternative simulation, the role of the NACS was removed. This simulation (with the ACS of Clarion alone) produced a reasonably good match with the human data. See Figure 5.8 and Figure 5.9 for the results.

However, comparing this alternative simulation with the previous one involving both the ACS and the NACS, although the ACS alone could capture the human data of this task to a large extent, the use of both the ACS and the NACS led to the better capturing of the human data. Thus, this comparison suggested that both the ACS and the NACS were needed, although the higher degree of freedom of the original model should also be kept in mind.

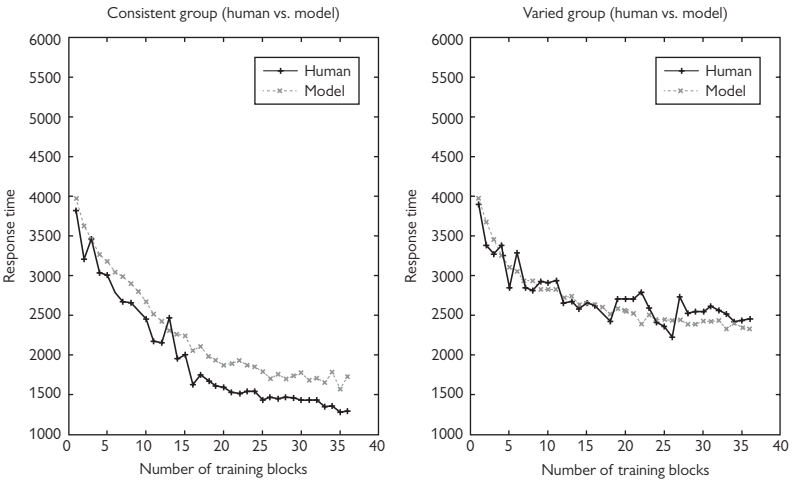

Figure 5.7. The learning curves from the human data and the Clarion simulation. Here "block" was defined as a set of 12 trials.

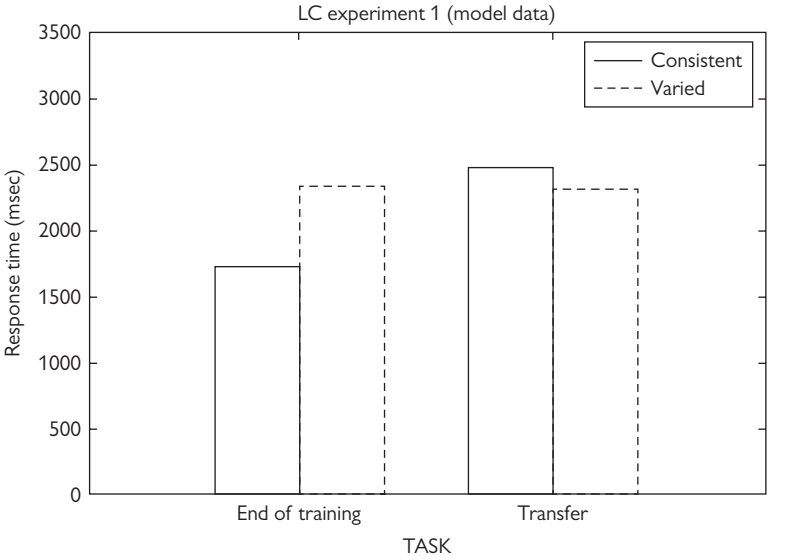

Figure 5.8. The Clarion simulation of experiment 1 without the NACS.

However, this alternative simulation still produced a better match than the ACT-R simulation, which suggested that the separation of implicit and explicit processes in Clarion (in particular, the separation of the bottom and the top level within the ACS) and other characteristics of Clarion

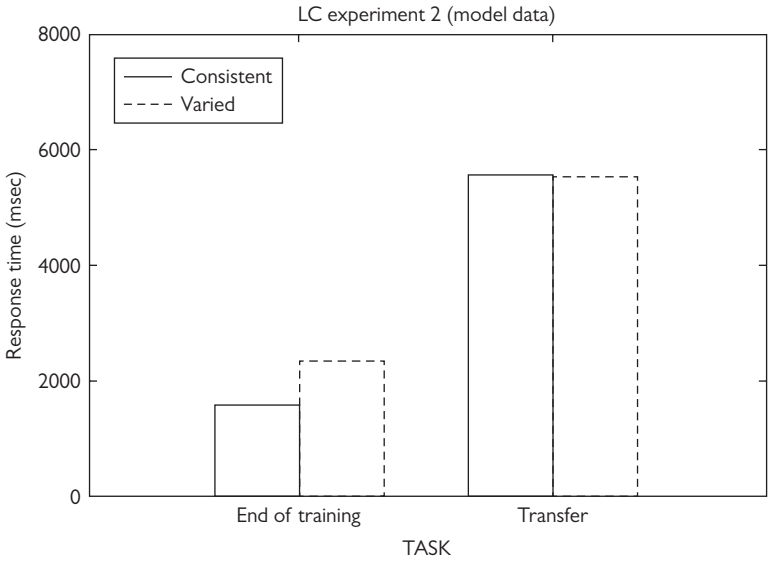

Figure 5.9. The Clarion simulation of experiment 2 without the NACS.

(e.g., including rule BLAs and chunk BLAs) had some advantages.[11] The framework of Clarion, and the separation of the top and the bottom level therein, contributed to the better capturing of human performance in this task.

In another alternative simulation, bottom-up learning (RER) was added. In this alternative simulation, top-down learning, which was still essential, and bottom-up learning, which was supplementary, were both involved. This alternative simulation captured the human data approximately equally well compared to the original simulation.

From the original simulation, it was evident that top-down learning could account for the human performance in this task, and therefore human learning in this task was likely top-down learning. This simulation added the possibility that bottom-up learning might also be involved in the human performance of this task, by virtue of the fact that the model with both top-down and bottom-up learning captured the human data

11. As pointed out before, if there was no separation of the top and the bottom level in Clarion, even with all the other characteristics of Clarion, there would be no performance difference between the two simulated groups, for example, in simulating the training phase of experiment 1.

approximately equally well, although strictly speaking bottom-up learning was not necessary in this case.

In yet another alternative simulation, there was no a priori "fixed rule" (FR) as used in the original simulation; only rules learned from RER were involved at the top level. This simulation was clearly a failure. This simulation could not capture most of the differences between the two groups.

Separately, simulation using the bottom level alone was also tested, to see if implicit learning alone was sufficient. Such a simulation could not capture the differences between the two groups. Moreover, the learning curves (in terms of response time) were flat for both groups.

So it was clear that the bottom level alone or bottom-up learning alone were inadequate for capturing human performance in this task. It appeared that a priori explicit knowledge and top-down learning were necessary for capturing human performance in this task, although bottom-up learning might be involved as well.

5.2.5. Discussion

This work shows that it is possible to separate the two dichotomies: implicit versus explicit and procedural versus declarative (action-centered versus non-action-centered). This separation leads to new possibilities of interpreting empirical data and new ways of understanding cognitive skill acquisition. The separation of the two dichotomies may lead to better, more psychologically realistic models, which is therefore worthy of further exploration.

The Clarion simulation provided some interesting interpretations of the human data. For example, it attributed the performance difference at the end of training between the consistent and the varied group to the difference between relying on implicit knowledge and relying on explicit rules. In so doing, Clarion went beyond existing simulations in providing interpretations that other models did not provide. Moreover, the Clarion simulation was more accurate than other simulations. The fact suggested, to some extent, the advantage of Clarion—synergy between action-centered and non-action-centered processes and between implicit and explicit processes. Note that similar comparisons were done in other tasks, such as Tower of Hanoi, artificial grammar learning tasks, and serial reaction time tasks (see, e.g., Sun & Zhang, 2004).

It is worth noting that this match between the simulation and the human data was obtained under the same set of parameters for all the groups (varied and consistent) and all the conditions (training, transfer 1, and transfer 2). Thus, this simulation was different from some other simulations using Clarion, such as the simulation of process control tasks discussed earlier. In these other tasks, experimental conditions varied across groups that required corresponding changes in model parameters. In this task, the only difference across groups was that of stimuli, which did not require any change of model parameters for simulating different groups. That is, there was no parameter estimation on a per group basis, which would have made the match easier to obtain but would have rendered the simulation less interesting. Because there were a total of three different conditions (training, transfer 1, and transfer 2), with two groups in each, it was not a trivial matter to obtain a good match using only one set of parameters. The match indicated, to some extent, the validity of the Clarion framework.

The comparison of the different Clarion simulations indicated that the best model, the model that most closely captured the characteristics of human performance in this task, was the one with both the ACS and the NACS and both implicit and explicit knowledge. Through these various simulations, the advantages of having separate action-centered and non-action-centered knowledge and the advantages of having separate implicit and explicit knowledge became evident (Sun et al., 2009).

The best model of this task generally followed a top-down direction: First, explicit knowledge (at the top level) was used to direct actions; gradually, implicit knowledge (at the bottom level) was learned from the guidance provided by the explicit knowledge; eventually (as in the case of the consistent group), implicit processes (at the bottom level) became competent. The simulation of this task showed that Clarion, despite its original focus on bottom-up learning, could fully accommodate this alternative direction of learning.

In this task, top-down learning was apparently more important than bottom-up learning. While combining top-down and bottom-up learning produced good results, top-down learning alone was successful in capturing the human data in this task. But bottom-up learning alone was insufficient. Implicit learning alone was also inadequate. Note, however, that different tasks and task settings may lead to different proportions of explicit and implicit learning, and different proportions of top-down and bottom-up learning (Sun, 2002).

So, Clarion captures a number of important psychological distinctions without confounding them. The simulations so far have shown that the separation of implicit and explicit processes and the separation of action-centered and non-action-centered processes are both important. Quantitatively, Clarion captured the human data in this task better than any existing model. Thus the simulations support this way of structuring the cognitive architecture. However, it should be pointed out that only suggestive evidence has been provided so far from the work described here. More work is needed, in terms of both human experiments and computational simulations.

5.3. Modeling the Categorical Inference Task

I now turn to declarative (non-action-centered) processes within Clarion and show how they account for various relevant empirical data. I will examine some specific data sets concerning human reasoning. (General empirical phenomena concerning human reasoning will be addressed later in Section 5.5.) Reasoning is an important cognitive faculty, which allows the generation of new ideas from existing ones. New ideas may be generated, for example, by the application of rules to particular cases but maybe also by the application of other mental structures (such as similarity).

Besides accounting for reasoning, declarative processes within Clarion are also relevant to accounting for human memory phenomena, concept and categorization phenomena, decision-making phenomena, and so on. The reader may refer to existing publications regarding these aspects, for example, Sun and Helie (2012), Sun and Helie (2013), and Helie and Sun (2014).

5.3.1. Background

Some interesting questions concerning declarative processes include: What is human everyday reasoning made up of? Is it fully captured by formal symbolic models (e.g., as proposed by logicians), or is it sufficiently different? Is it fully explicit or is it mixed involving both explicit and implicit processes? Computationally speaking, how does

one account for such reasoning within the NACS (as detailed in Chapter 3)?

A little background is in order here. Sun (1991) proposed a theory of human everyday reasoning,[12] which was further elaborated in Sun (1994, 1995). The basic tenet of this theory was that, to a significant extent, human everyday reasoning consisted of rule-based and similarity-based reasoning; much of human everyday reasoning was reducible to a combination of these two types of processes. Mixing rule-based and similarity-based reasoning could lead to complex patterns of inferences (as observed in human reasoning). Both of these two types of processes could be captured within a unified model.

The theory was backed up by empirical evidence in the form of verbal protocols from Collins (1978) and Collins and Michalski (1989). These protocols were analyzed in Sun (1994), which showed that the vast majority of the protocol data might be captured by intermixing rules and similarity. A model was developed that accounted for these protocols (Sun, 1991, 1995). Relevant to this theory, Sloman (1993) published a set of experiments showing that similarity played a significant role and that similarity might be characterized by feature overlapping (as hypothesized in Sun, 1991). Later, Sloman (1998) described further experimental results concerning category inclusion relations that supported the theory as well. The theory of Sun (1991) evolved into Clarion, which included not only reasoning but also skill acquisition, motivation, metacognition, and many other psychological aspects.

In the remainder of this section, data of human everyday reasoning are analyzed, and then the analysis is instantiated into a computational model in Clarion. Simulation based on the model is then described. In short, the simulation accurately captured human data, which illustrated the respective roles in human everyday reasoning played by rule-based and similarity-based processes, as well as the respective roles played by implicit and explicit processes. Furthermore, it demonstrated how such reasoning naturally fell out of Clarion. The simulation provided a detailed and plausible explanation of the human data. The discussion draws upon Sun (1991, 1994) and Sun and Zhang (2006).

12. Human everyday reasoning has also been termed "mundane" reasoning or "commonsense" reasoning (Sun, 1994).

Note that, in a way, the work described here in this section illustrates synergy between implicit and explicit declarative (non-action-centered) processes, which results from the interaction of these two types of processes and their associated knowledge, similar to and corresponding with the synergy resulting from implicit and explicit procedural (action-centered) processes described in the preceding sections of this chapter.

5.3.2. Task and Data

Below some data that illustrate the interplay of similarity-based and rule-based reasoning (SBR and RBR, respectively) in human reasoning are examined: the data from experiments 1, 2, 4, and 5 of Sloman (1998), which are most relevant to this issue.

In experiment 1 of Sloman (1998), subjects were given pairs of arguments, each consisting of a premise statement and a conclusion statement. Some of these pairs of arguments were in the form of "premise specificity" (with premises of different degrees of specificity leading to the same conclusion):

a. All flowers are susceptible to thrips. ⇒ All roses are susceptible to thrips.
b. All plants are susceptible to thrips. ⇒ All roses are susceptible to thrips.

Some other pairs of arguments were in the form of "inclusion similarity" (the same premise leading to conclusions of different degrees of similarity to the premise):

a. All plants contain bryophytes. ⇒ All flowers contain bryophytes.
b. All plants contain bryophytes. ⇒ All mosses contain bryophytes.

Subjects were asked to pick the stronger of the two arguments from each pair. Each subject was given 18 pairs of arguments (among other things not relevant here).

The results showed that the more similar argument from each pair of arguments was chosen 82% of the time for inclusion similarity and 91% of the time for premise specificity. Tests showed that these percentages were statistically significantly above chance, either by subjects or by argument pairs.

These arguments might be viewed as enthymematic (Sun & Zhang, 2006). But they were more than just enthymemes, due to the involvement of SBR (similarity-based reasoning). It should be apparent that if only RBR (rule-based reasoning, e.g., based on some deductive logic) had been used, similarity would not have made any difference, because the conclusion category was contained in the premise category and thus both arguments in each pair should have been equally strong. Therefore, the data suggested that SBR (as distinct from RBR that captures category inclusion relations) was involved to a significant extent.

In experiment 2 of Sloman (1998), subjects were instead asked to rate the likelihood ("conditional probability") of each argument. Ratings could range from 0 to 1. The results were as follows. The mean rating was 0.89 for inclusion similarity and 0.86 for premise specificity. Statistical tests showed that both were significantly below 1, by subjects and by arguments. Again, it would have been the case that the outcome was uniformly 1 if only RBR had been used, because the conclusion category was contained in the premise category. Thus SBR was significantly present here too. Indeed, statistical tests showed that across subjects there was a significant effect of similarity (low versus high). So was the case across argument pairs.

In experiment 4, subjects were asked to rate the likelihood of each argument. Ratings could range from 0 to 1. However, in this case, a category inclusion relation was specifically presented as part of each and every argument. For example,

> All plants contain bryophytes. All mosses are plants. ⇒ All mosses contain bryophytes.

The results showed that the mean judgment was 0.99. Most subjects gave all 1s. Most arguments received judgments of all 1s (excluding one individual who gave 0.99 throughout). In other words, SBR phenomena almost disappeared. Instead, it appeared that an RBR mode, based on category inclusion relations, was used.

Experiment 5 was similar to experiment 2, in that ratings were obtained. However, before any rating was done, subjects were asked to make category inclusion judgments. Thus, in this case, ahead of all the ratings, subjects were reminded of RBR involving category inclusion relations. Therefore, they were more likely to use RBR, although probably not as much as in experiment 4, due to the separation of category

inclusion judgments and argument likelihood ratings in the experimental procedure (unlike that of experiment 4).

The results showed that none of the subjects gave a judgment of 1 for every argument, indicating that SBR might be at work. Compared with experiment 2, having subjects make category inclusion judgments increased the likelihood rating. The mean judgment for experiment 5 was 0.92 (0.93 for inclusion similarity and 0.91 for premise specificity), as opposed to 0.87 for experiment 2 (0.89 for inclusion similarity and 0.86 for premise specificity). This increase arguably reflected the increased involvement of RBR. Nevertheless, statistical tests showed a significant effect of similarity (low versus high) across subjects and across argument pairs.

Based on the analysis above, RBR and SBR were both involved in these experiments, with varying proportions. Among them, experiment 1 and experiment 2 both involved SBR to a very significant extent. Experiment 4 involved explicit use of categorical relations, and thus mainly RBR. Experiment 5 involved more SBR, compared with experiment 4, along with RBR.

It is important to note that, given the co-existence of RBR and SBR, formal logics (or their psychological variants; e.g., Rips, 1994) may be suitable for capturing the RBR aspect of the human data, but not the SBR aspect. This is because they would not be suitable for distinguishing between the two arguments within each pair in terms of similarity. Although one may argue that logics could encode whatever similarity relationships that humans employed, such a "solution" would not be satisfactory for many reasons, including its ad hoc nature and its high representational cost (resulting from pair-wise similarity representations).

5.3.3. Simulation Setup

This task was simulated to validate the conceptual analysis above. The simulation demonstrated the significance of similarity-based reasoning (SBR). The significant role of SBR distinguished this type of reasoning from the kinds of reasoning naturally captured by logics, production systems, or probabilistic/Bayesian frameworks (see more discussions of this point in Sun, 1994, and Sun & Zhang, 2006). Furthermore, the simulation showed computationally the interaction and the synergy between rule-based and similarity-based reasoning, as well as those between

implicit and explicit processes (because reasoning was computationally carried out through the interaction of implicit and explicit processes within the NACS of Clarion).

In accordance with the conceptual analysis earlier, the following process was posited (Sun & Zhang, 2006): first, a premise statement was presented to a subject. The premise statement (e.g., "all flowers are susceptible to thrips") was then coded as a rule at the top level. The two concepts involved were coded as chunks, both implicitly (in distributed representation at the bottom level) and explicitly (in localist representation at the top level, in the form of chunk nodes). Inclusion relations, such as "roses are flowers," were already existent as rules at the top level, due to prior knowledge, coded using corresponding chunk nodes. When it came to dealing with the conclusion statement (e.g., "all roses are susceptible to thrips"), the first concept of the conclusion statement (e.g., "rose") was presented; that is, the corresponding chunk node was activated. Due to the similarity between the first concept of the conclusion statement and the first concept of the premise statement, the rule representing the premise statement was activated to a degree corresponding to the similarity between the two concepts. As a result, the target concept (i.e., the second concept of the premise statement, e.g., "thrips") was partially activated due to the application of the rule encoding the premise statement; the extent of its activation was determined by the aforementioned similarity.

When there was a pair of such arguments, this process was repeated. The two partial activations of the target concept were temporally stored. Then the two partial activations were compared. In case of forced choice, one of them was selected using a Boltzmann distribution of activations. The selection favored the more strongly activated one (although the selection was stochastic). Of course, the process was directed by the ACS, performing its executive control functions.

For simulating the different experimental settings of this task, the following manipulations were used: for simulating settings where SBR was dominant, RBR was de-emphasized by using lower weights for RBR. For simulating settings where RBR was dominant, RBR was not deemphasized. The relative emphasis of the two methods (RBR versus SBR) was accomplished through the scaling (balancing) parameters discussed in Chapter 3. The parameters were set at $\beta_1 = 0.50$ and $\beta_2 = 1.00$ for experiments 1 and 2, because of the heavy reliance on SBR (as opposed to RBR)

as suggested by the experimental settings and the analysis of the human data earlier. For simulating experiment 4, they were set at $\beta_1 = 1.00$ and $\beta_2 = 1.00$, because this setting prompted much more reliance on RBR as indicated by the human data. For simulating experiment 5, they were set at $\beta_1 = 0.88$ and $\beta_2 = 1.00$, because this experiment involved an intermediate level of reliance on RBR as indicated by the human data. In all, these values were set in accordance with the earlier interpretations of what happened under the different experimental conditions respectively.

Before simulating the experiments of Sloman (1998), pre-training of the model captured prior experiences that subjects had. Pre-training included presenting categorical features along with category labels to the NACS. Chunks were used to represent categories such as "flowers" and "plants." The dimensional values of these categories were represented in the bottom level, and the chunk nodes representing these concepts at the top level were linked to the dimensional values at the bottom level. Pre-training also included presenting relevant category inclusion relations, such as "flowers are plants" or "mosses are plants," and as a result they were coded as associative rules at the top level of the NACS.

In the bottom level of the NACS, although associative memory networks were present, they were not relevant to the performance of this task. This was due to the fact that subjects likely had no prior experiences with concepts such as "thrips," since this task used many little-known concepts presented only once. Corresponding to that, before simulation, there was no pre-training of the associative memory networks with any data related to these concepts.

During the simulation, when a category name was given, the corresponding chunk node was activated to the full extent (i.e., 1). Then, through associative rules and similarity-based processes (with top-down and bottom-up activation flows), conclusion chunk nodes were also activated, combining SBR and RBR according to the scaling (balancing) parameters. Conclusion chunks were retrieved along with their confidence levels. For simulating rating of conclusions (as in experiments 2, 4, and 5), the confidence levels of conclusion chunks were used. However, for simulating forced choices (as in experiment 1), a stochastic selection (based on a Boltzmann distribution of confidence levels) was used to choose between two outcomes.

The following action rules, among many other action rules, were implemented in the ACS for directing the performance of the NACS in this task:

If goal = forced-choice-task and no source category has been presented, then present the first source category, obtain the rating of the target category, and store it in the working memory.

If goal = forced-choice-task and one source category has been presented, then present the second source category, obtain the rating of the target category, and store it in the working memory.

If goal = forced-choice-task and both source categories have been presented, then conduct a stochastic selection from the two ratings stored in the working memory, and report the result.

If goal = rating-task and no source category has been presented, then present the first source category, obtain the rating of the target category, and store it in the working memory.

If goal = rating-task and one source category has been presented, then present the second source category, obtain the rating of the target category, and store it in the working memory.

If goal = rating-task and both source categories have been presented, then report the two ratings from the working memory.

These "fixed rules" (Sun, 2003) were presumably derived from a priori knowledge and task instructions (given to subjects prior to experiments). The goals involved in these rules were set in the goal structure when the task instructions were given.

5.3.4. Simulation Results

The experiments 1, 2, 4, and 5 of Sloman (1998) were simulated as described above. The results were as follows (Sun & Zhang, 2006).

Recall that in experiment 1, subjects were asked to pick the stronger of the two arguments from each pair. The simulation of experiment 1 showed, the same as the human data, that the more similar argument from each pair of arguments was chosen more often: 82% of the time for inclusion similarity and 83% of the time for premise specificity. These percentages were significantly above chance, either by simulated "subjects" or by argument pairs, the same as in the human data.

In the simulation setup of this experiment, there was a significant involvement of SBR. If only RBR had been used, then similarity could not have made a difference and thus both arguments in a pair would have been equally strong. This simulation demonstrated that the hypothesized role of SBR in the human data was a reasonable interpretation of this experiment, given the close match with the human data.

In experiment 2, subjects were instead asked to rate the likelihood of each argument. In the simulation of experiment 2, the mean rating obtained was 0.86 for inclusion similarity and 0.87 for premise specificity. Both were significantly below 1, both by "subjects" and by arguments, different from what would have been predicted if only RBR had been involved, the same as in the human data. Furthermore, across "subjects" and across argument pairs, there was a significant effect of similarity (low versus high).

With the same simulation setup as the previous experiment, this simulation again demonstrated the same pattern of significant involvement of SBR, as in the human data. This pattern could not be captured naturally by usual RBR.

Now move on to the simulation of experiment 4. Recall that in experiment 4, subjects were asked to rate the likelihood of each argument, which included the corresponding category inclusion relation. The simulation of experiment 4 produced the mean judgment 0.99, the same as the human data.

Compared with the simulation of experiment 2 earlier, RBR at the top level based on category inclusion was much more prominent in the simulation of this experiment, as appropriate from the analysis of the human data, and as specified in the simulation setup. This setup captured the human data accurately.

Now turn to the simulation of experiment 5. Recall that in experiment 5, subjects were asked to make category inclusion judgments before ratings were obtained from the subjects. In this case, subjects were reminded of RBR involving category inclusion relations and therefore they were more likely to use RBR compared with experiment 2, although not as much as in experiment 4 due to the temporal separation of category inclusion judgments and ratings. In this simulation, the mean judgment for experiment 5 was 0.91 for both inclusion similarity and premise specificity, as opposed to 0.86 and 0.87 for the two cases in the simulation of experiment 2. Furthermore, in the simulation data, there

was a significant effect of similarity (low versus high), across simulated "subjects" and across argument pairs.

This simulation replicated the human data well. This result showed that the interpretation embodied in the simulation setup, that is, less involvement of RBR compared with experiment 4 but more compared with experiment 2, was a reasonable one.

In all, the simulation of this task substantiated and validated the earlier analysis of human performance of this task. In particular, computational processes were formulated and carried out, which replicated accurately the human data. As a result of the match with the human data, the computational specification constitutes a detailed, plausible, mechanistic, and process-based explanation of corresponding human reasoning.

5.3.5. Discussion

The simulation of these experiments of Sloman (1998), with both rule-based and similarity-based reasoning, captured the human data well, and thereby demonstrated in a way the importance of similarity-based reasoning in human everyday reasoning, involving both implicit and explicit processes (Sun, 1994). The similarity-based approach is distinct from probabilistic reasoning and other methods implemented in other cognitive architectures. Let us compare some of these approaches.

ACT-R, for one thing, tries to capture all inferences in a probabilistic framework (Anderson and Lebiere, 1998). In doing so, it tends to lump together all forms of weak inferences. This approach has shortcomings. With this approach, similarity relations between any two objects must be explicitly represented with all the associated parameters, which would be unnecessary in Clarion. Thus in ACT-R, the representational cost for similarity-based reasoning may be high. Although partial match may be used in ACT-R to handle some limited forms of similarity-based reasoning, partial match alone is not sufficient to handle the full extent of similarity-based reasoning (Sun, 1994). In addition, with this approach, it is difficult to take context into consideration in similarity-based reasoning.

In general, the limitations of probabilistic reasoning include its neglect of many human heuristics, simplifications, and rules of thumb (Tversky and Kahneman, 1983; Sun, 1994; Gigerenzer et al., 1999), useful for reducing computational costs of formal mathematical models. As a result, this approach suffers from higher computational complexity. This approach is not directly adopted in Clarion (but see Section 5.5).

One may also look into the logic-based framework of Collins and Michalski (1989), which incorporated similarity-based reasoning through representing similarity in a complex logical formalism. Similarity was represented as a logical operator. So for almost any pair of objects, there would be a logical relation explicitly represented regarding their similarity. Inferences could be performed on the basis of these similarity operators using a search process. The cost of this process would be high.

Generally speaking, for capturing human reasoning (of which similarity-based processes are part), although logic-based models are useful, they suffer from a number of well-known shortcomings, including their restrictiveness and their inadequacy in dealing with various forms of inexactness found in the real world (Sun, 1994). They are generally also computationally costly.

In a different vein, psychological work on reasoning is relevant here. Work on deductive reasoning includes mental logic (Rips 1994; Braine and O'Brien 1998) and mental models (Johnson-Laird and Yang, 2008). Neither of the two approaches dealt with similarity-based reasoning as captured in Clarion. On the other hand, in research on inductive reasoning (see, e.g., Heit, 2008), many competing approaches exist, but they are not well integrated with deductive reasoning (and many other psychological functions). See Section 5.5 for more discussions of inductive reasoning.

The present model, combining similarity-based and rule-based reasoning, offers a plausible way of capturing some essential patterns of human everyday reasoning (although maybe not all patterns of human reasoning). It provides a unified explanation of a variety of reasoning patterns (Sun, 1994). It complements probabilistic/Bayesian or logic-based models, which are centered on strict mathematical formalisms and thus also limited by such formalisms.

The simulation validated, to some extent, some general postulates of Clarion concerning human reasoning. It is useful to posit the existence of two separate levels: explicit versus implicit. The interaction between implicit and explicit levels is also useful to posit. In addition, the Clarion approach may capture similarity-based, metaphoric, and analogical reasoning, as well as case-based reasoning in AI (see, e.g., Sun, 1995b).

However, this work is just one step toward fully accounting for human reasoning in a comprehensive, unified way. The simulation so far has shown the promise of this approach, as well as its distinction from other

approaches. Section 5.5 provides more discussions of accounting for general human reasoning patterns (see also Sun, 1994, 1995). The next section addresses another aspect—intuition and insight.

5.4. Modeling Intuition in the Discovery Task

I now turn to capturing and explaining human intuition and insight, based on the interaction of implicit and explicit processes within Clarion.

5.4.1. Background

Clarion captures human everyday reasoning of a variety of forms. While rule-based and similarity-based reasoning were explored earlier, here I explore yet another aspect—intuition and insight. It should be noted that Clarion includes all of the following: rule-based reasoning, similarity-based reasoning, and associative memory networks (at the bottom level of the NACS), and therefore their interplay can be explored for capturing intuition and insight beyond what was discussed in the previous section. In so doing, we can incorporate both explicit and implicit forms of human reasoning in one unified framework, rather than separating them following conventional wisdom.

Specifically, in this section, a "discovery" task is addressed, where insights often emerge from accumulating intuition (Sun & Zhang, 2006; Helie & Sun, 2010). Such a task is useful for understanding finer details of human reasoning, especially implicit processes underlying it. An analysis of the task was implemented in Clarion for describing the empirical data in a precise, specific way (as opposed to informal theories about intuition and insight). This work points to the significant role played by implicit associative memory networks in generating intuition and in leading to insight.

A clarification is probably needed here. Although intuition has often been defined as "the immediate apprehension of an object by the mind without the intervention of any reasoning process" (Oxford English Dictionary), I instead view intuition as a type of reasoning (on the basis of sensory information, motivation, and so on). Reasoning encompasses explicit processes (especially explicit rules and logics) on the one hand, and implicit processes (including intuition) on the other (Sun, 1994). In fact, intuition and insight

are arguably important components of human reasoning. They supplement and guide explicit reasoning.

Whereas explicit reasoning processes have been amply explored in cognitive science and in AI (e.g., Collins and Michalski, 1989; Johnson-Laird and Yang, 2008), implicit reasoning processes have not yet been extensively explored. Explicit reasoning is often ineffective when the problem is complex, ill-understood, or ambiguous. In such a case, an alternative approach relying more on intuition and insight might be more appropriate. Thus, studies of reasoning involving intuition and insight are needed.

In this regard, Helie and Sun (2010) proposed a Clarion theory of creative problem solving that centers on the interaction of implicit and explicit processes, relying heavily on intuition and insight. A computational model implemented the theory. The theory since then has been used to account for many observed phenomena and prior theories, while the corresponding computational model has been used to simulate a variety of empirical data. According to the theory, intuition and insight are the key to creative problem solving, which have been tested through computational simulation. The reader is referred to Helie and Sun (2010) for further details on this theory.

Below, I examine the "discovery" task (Bowers et al., 1990) and some data from this task. The task and the data can be simulated in a number of ways in Clarion, for example, with either auto-associative or hetero-associative memory networks (AAM or HAM). The following discussion draws upon Sun and Zhang (2006) and Helie and Sun (2010b).

5.4.2. Task and Data

First, some human data are examined that illustrate intuition and insight in human reasoning, from the discovery task of Bowers et al. (1990), which shows that a gradual "warming up" process may underlie intuition and sudden insight.

In the task, Bowers et al. attempted to assess the continuous nature of implicit intuitive processes leading to sudden insight. To test their hypothesis, in their experiment 3A, fifteen clue words were presented sequentially, one word at a time, to the subjects, and the subjects' task was to find a word (the target word) that was associated with them. Subjects were required to generate a target word with which each of the clue

words presented thus far was associated after the presentation of each clue word.

Specifically, first, a clue word was displayed, and the subjects had 15 seconds to generate a word associated with the clue word. Following this 15-second period, a second clue word was added, and the subjects had 14 seconds to generate a word associated with these two clue words. Each time a new clue word was added, all the previous clue words remained on the screen, and the time allowed to generate an answer was shortened by one second until it reached the sixth clue word. From that point on, the subjects had 10 seconds to generate an answer after every additional clue word (up to the 15th).

At any step, if subjects viewed a generated word as a potential solution, they were to checkmark it (indicating a "hunch"). When they were convinced that the word was a solution, they were to mark it with an X (indicating a "conviction").

Each subject solved 16 different problems. The dependent variables were the number of clue words needed to reach a hunch and the number of additional clue words needed to turn a hunch into a conviction.

Each clue word was a response to a stimulus word in the Kent-Rosanoff word association test (Kent and Rosanoff, 1910). The first 12 clue words occurred five or less times out of 1,000 as a response to the stimulus word, and they were randomly assigned to position 1–12. The last three clue words occurred more than five times and were randomly assigned to position 13–15. The clue words 13–15 were on average 13 times more frequently associated to the target word than the other clue words according to the Kent-Rosanoff word association test.

As reported in Bowers et al. (1990), subjects arrived at a hunch on average with 10.12 clue words. The average number of clue words needed to go from a hunch to a conviction was 1.79. The result might be interpreted as showing that continuous processes were involved in the task, and that the suddenness of insight simply reflected the reach of a "conscious" threshold.

As suggested by Bowers et al. (1990), subjects could respond discriminatingly to coherence that they could not explicitly identify, and this implicit recognition of coherence guided subjects gradually toward an explicit representation. Subjects "warmed up" to the solution in a gradual manner. That is, underlying implicit processes were rather continuous. An implicit representation might gradually gain strength; when the level of

activation reached a certain degree, the implicit representation triggered an explicit one. On the other hand, an explicit representation might also be activated as a result of relevant explicit knowledge (e.g., explicit rules) early on. However, there were often few such explicit rules and therefore they were often irrelevant, as indicated by the initial implicitness of recognition and by the gradual explication shown by human subjects in this task (Bower et al., 1990).

A "hunch" was indicated, presumably as a result of an activated explicit representation. Even after that point, its activation might continue to grow, and thus, eventually, subjects might indicate a "conviction"—presumably an even stronger explicit representation. Bowers et al. further speculated that hunches were often implicitly generated and explicitly tested; that is, they often resulted from implicit processing, while convictions were often more explicitly reached.

Mechanistically, one can easily imagine that people are frequently "trained," incidentally or deliberately, with word associations in everyday life (e.g., desk-chair, pen-paper, and so on). Such experiences help to form associations of various strengths, based in part on frequencies of co-occurrences in everyday life. Association formation happens mainly in implicit memory, because of the (mostly) incidental nature of association formation; furthermore, it happens mainly in implicit declarative memory, because it is not concerned with procedural processes.

During the experiment, clue words were presented one at a time. At the presentation of each clue word, all associated words were activated in implicit declarative memory. With each new clue word, more activations were accrued to some associated words. Gradually, activations of some words became stronger and stronger. As a result of this (as well as explicit rules possibly), explicit representations of some words were activated in explicit declarative memory. Eventually, a threshold (the threshold for "hunches") was crossed, and thus a hunch was found. Furthermore, when more clue words were presented, more activations were accrued to the explicit representations of some words. A second threshold (the threshold for "convictions") was crossed, and thus a conviction was declared.

Below this conceptual analysis of the processes underlying this task is implemented in two computational models within Clarion, which serve to substantiate and validate the analysis.

5.4.3. Simulation Setup

This task is simulated within the NACS of Clarion (because it is concerned mostly with declarative processes, not procedural processes). Within the NACS, it may be simulated with either a hetero-associative memory network (HAM) or an auto-associative memory network (AAM) at the bottom level. Thus two plausible explanations were produced, which were conceptually similar but technically different.

For simulating this task, models must be pre-trained. Training was needed to capture gradually formed implicit word associations that human subjects possessed through their prior experiences. The NACS was under the control of executive functions embodied by the ACS.

In the case of simulation involving a hetero-associative memory network (i.e., a Backpropagation network; see Sun & Zhang, 2006), during training, pairs of words were presented to the NACS. The input to the network at the bottom level included the current clue word and the previous clue words. The input nodes of the network corresponded to the microfeature (dimensional value) representations of the clue words. The output nodes of the network corresponded to the microfeature (dimensional value) representations of potential target words.[13] The bottom level involved distributed representation.

Each of the first 12 words on a list in the stimulus material, paired with the target word, was used for training for about 4% of the training time. Thus these words took up about 48% of the training time. These associations were under-trained, and thus the network did not perform well at the end of the training, capturing weak implicit associations between these pairs of words (as in the human experiment). This process was in fact the reverse of the word association test in Kent and Rosanoff (1910).

Each of the last three words on a list in the stimulus material was also used for training, paired with the target word. Each of these words was used for training for 17% of the time, for a total of 51% of the training time, reflecting the stronger associations as indicated by the word association test. A total of 10 lists of words were used.

13. The inputs and outputs should be phonological and morphological features of words. However, for the sake of simplicity, artificially constructed microfeatures were used in simulation. This simplification did not affect the outcome of the simulation.

As a result of the presentation of these words during training, each of these words was coded as a chunk node in the top level. The microfeatures (dimensional values) of these chunks were represented at the bottom level (with distributed representation), and chunk nodes at the top level were linked to the microfeature nodes at the bottom level.

At the top level of the NACS, explicit associative rules were learned, which captured explicit associations between words. However, due to the relatively infrequent presentation of association pairs, there were few explicit associative rules established in the top level. The frequency of invocation of these explicit associative rules would likely be below the minimum necessary to keep them (as discussed in Chapter 3), and thus they would be deleted. Similarity-based reasoning through the interaction between the two levels as discussed in the context of the categorical decision task in the previous section was not significant in this task.

During the test, clue words were presented one at a time. At the presentation of each clue word, the partial sequence of words seen thus far was presented to the NACS. Thus, the activation of the target word became stronger and stronger in activation as a result of the accumulating inputs to the NACS.

The integration of the two levels of the NACS was as explained before: to combine the bottom-up activation of a chunk node (due to the bottom level) with the activation of a chunk node from sources at the top level, a *max* function was applied. The integrated activations were then transformed into a Boltzmann distribution. An answer was selected from the Boltzmann distribution.

Due to accumulating evidence within the NACS, a tentative answer (a hunch) first emerged, and then a final answer (a conviction) was generated. If the confidence level (the integrated activation of the output) was higher than a low threshold, the output was marked as a hunch. If a hunch had already been identified and the confidence level was higher than a high threshold, the output was identified as a conviction.

During the test, the action rules in the ACS (acquired from a combination of a priori knowledge and task instructions given to subjects prior to the experiment) were used to control the NACS. The goal for the ACS was set in the goal structure when task instructions were given before the test began. The time limits imposed on subjects during the human experiment were not relevant to this simulation, because in this simulation

setup, at both levels, only one pass from inputs to outputs was involved. Therefore the response time was relatively fixed (which, notably, would not be the case for an alternative simulation to be detailed below).

Turn now to an alternative simulation involving an auto-associative memory network (an AAM; Helie and Sun, 2010b) at the bottom level. In this simulation, at the bottom level, each clue word was represented by a set of nodes (a distributed representation); each target word was also represented with a distributed representation at the bottom level. In the top level, as before, the condition of a rule contained clue words while the conclusion contained a target.

The stronger associations of clue words 13–15 with their target words were captured through training. Clue words 13–15 were presented more often than other clue words (on average 13 times more often, as consistent with the human data). Thus, at the end of training, associations between clue words 13–15 and their target words were stronger.

To simulate the test, a stimulus (clue words) activated both the top level and the bottom level. Both explicit and implicit processing took place. Because each iteration in the bottom level was assumed to take 350 milliseconds of psychological time (Chapter 3), 43 iterations were allowed for the first clue word (amounting to 15 seconds, as in the human experiment), and there was a decrement of 3 iterations for each subsequent clue word (a decrement of 1 second) until the fifth. From the sixth to the fifteenth, 28 iterations were allowed (10 seconds). Once the processing in the bottom level was completed, activations of the bottom level were sent to the top level and integrated with the activations at the top level. The integrated activations were then transformed into a Boltzmann distribution and an output was selected. If the confidence level (the integrated activation) was higher than a low threshold, the output was marked as a hunch. If a hunch had already been identified and the confidence level was higher than a high threshold, the output was identified as a conviction. Because of the accumulation of clue words in the inputs to the NACS, activation of the target word gradually increased in the bottom level.

5.4.4. Simulation Results

First look into the results of the simulation using the hetero-associative memory network in the bottom level of the NACS. Recall that the human

data of this task indicated that (1) the average number of clue words at which a hunch was arrived at was 10.12, and (2) the average number of clue words needed to go from a hunch to a conviction was 1.79. Matching the human data closely, the result from this simulation indicated that (1) the average number of clue words at which a hunch was arrived at was 10.23, and (2) the average number of clue words to go from a hunch to a conviction was 1.72. The simulation results were well inside the confidence intervals of the human data. Clearly, the match was excellent.

To better understand this simulation and the interpretation of the data embodied in this simulation, let us examine some details. From Figure 5.10, it was clear that there was a gradual accumulation of activation on the target word over time, due to successive additions of clue words. This accumulation was the result of both the bottom level as well as the top level of the NACS. For example, the contribution of the top level was shown in Figure 5.11, in terms of number of matching associative rules. There was an increase in number of matching rules toward the end of the list.

Implicit associations at the bottom level developed gradually during training. Figure 5.12 showed the gradual development of implicit

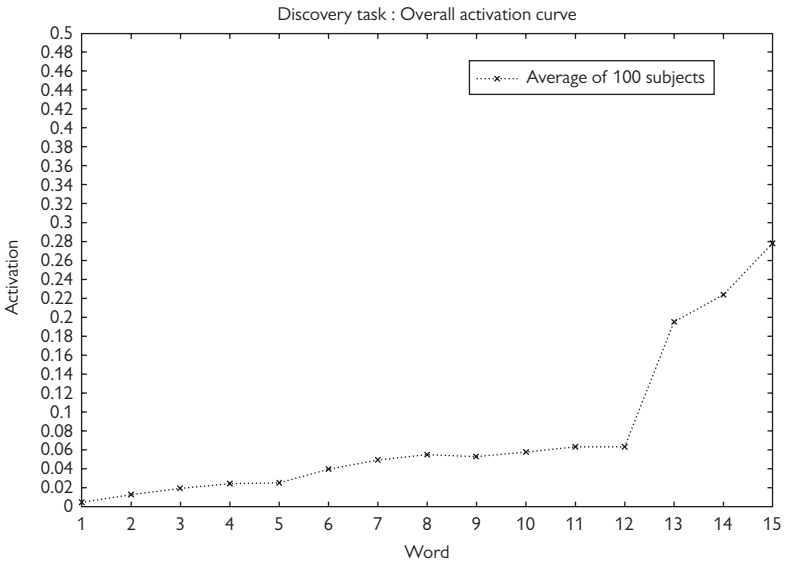

Figure 5.10. Accumulation of activation. The *x*-axis represents the number of words presented. The *y*-axis represents the activation of the target word.

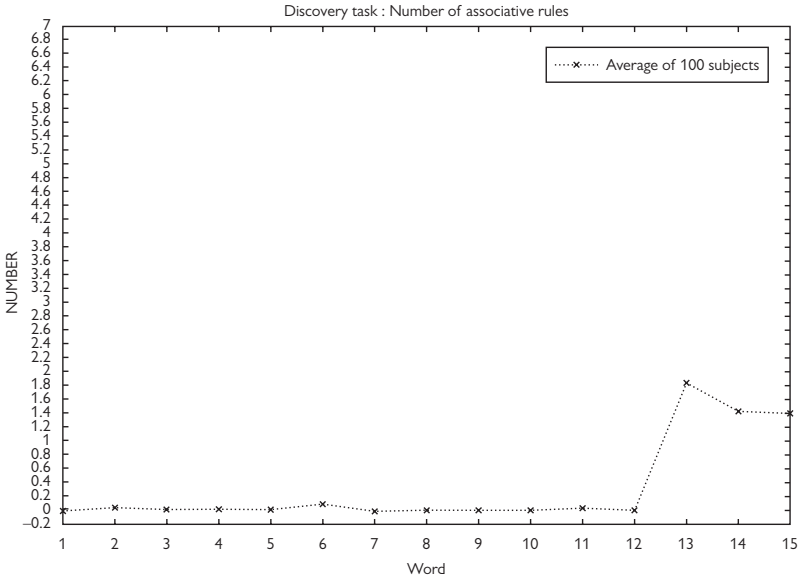

Figure 5.11. The number of matching rules for each clue word. The *x*-axis represents the number of words presented. The *y*-axis represents the number of matching rules.

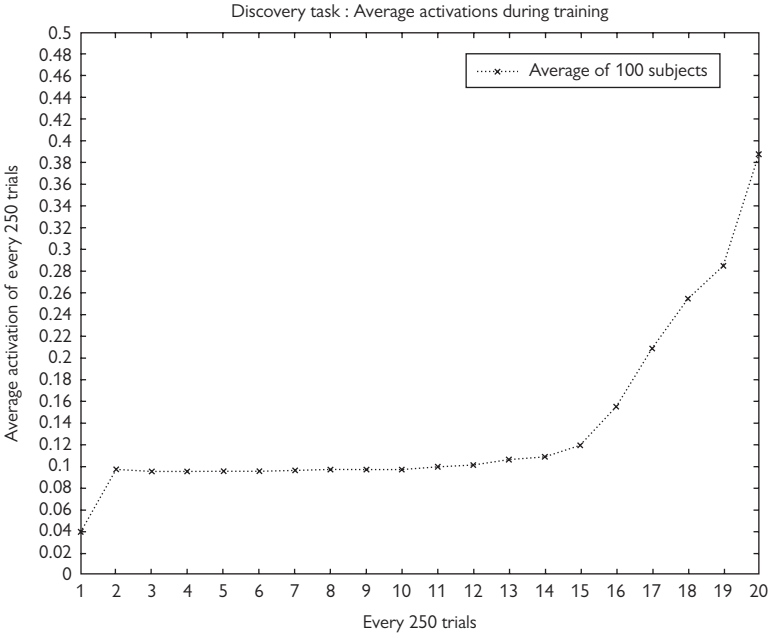

Figure 5.12. The learning curve of the bottom level. The *x*-axis represents the number of training trials. The *y*-axis represents the average activation of the target word.

associations over time during the course of training. During training, explicit associative rules were also formed occasionally at the top level. There were more rules for words toward the end of the list than toward the beginning (Figure 5.11), because the words toward the end were used more frequently during training and thus were more likely to form associative rules (besides developing stronger implicit associations in the bottom level).

To further validate the model, the bottom level alone (with the heteroassociative memory network) was tested for simulating this task; that is, the top level and rule learning were in effect removed through setting the scaling parameter for the top level to zero. The fit was significantly worse, despite repeated adjustments of parameters. Comparing the bottom level alone simulation with the full simulation indicated the necessity of including both types of processes (implicit and explicit).

The alternative simulation—the one with an auto-associative memory network (AAM) in the bottom level—produced similar results (Helie and Sun, 2010b). The average number of clue words needed to reach a hunch was 9.8, and 2.0 additional clue words were needed to reach a conviction, which were well inside the confidence intervals of the human data. Figure 5.13 showed the activations in the top level (the full line) and in the bottom level (the dashed line).

Because *max* was used for integrating the results from the two levels after the presentation of each clue word, hunches were on average generated by implicit processing (the dashed line was above the full line before 9.8 clue words), while convictions were more likely the results of explicit processing (the full line was above the dashed line after 11.8 clue words). All these results were in line with the human data and the analysis.

5.4.5. Discussion

Clarion captured well the human data of the discovery task. Clarion captured intuition and insight in this data set. Although not a typical topic in cognitive science, intuition and insight have been documented experimentally, and some data have been accumulated (e.g., Bowers et al., 1990; Schooler et al., 1993; Helie & Sun, 2010; and so on). The explanation of this phenomenon, however, is not as clear as one would like it to be, and thus computational modeling and simulation are useful in developing a better theory.

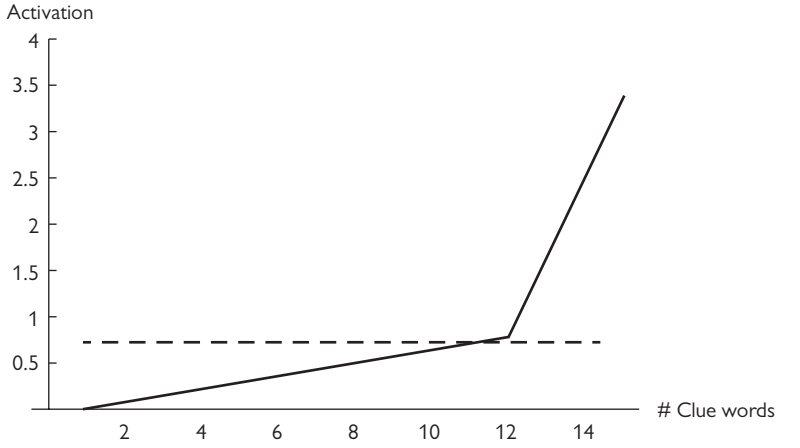

Figure 5.13. The average activation in the top and the bottom level after the presentation of each clue word. The dashed line represents the activation of the bottom level, while the full line represents the activation of the top level.

The simulation demonstrated the capability of Clarion in capturing and explaining this type of phenomenon. The simulation falls out of the overall framework and the existing mechanisms of Clarion, which include both implicit and explicit processes. In this regard, the simulation has shown that it is useful to posit the existence of these two types of processes. Whereas explicit processes tend to be all or nothing, implicit processes allow more gradual accumulation of information. Furthermore, the simulation has shown that the interaction between implicit and explicit processes, in the sense that implicit intuition gives rise to explicit awareness and vice versa, is important to human everyday reasoning.

In addition to what has been described here, a comprehensive theory of creative problem solving, involving intuition and insight, has been proposed in Helie and Sun (2010), based on Clarion. The theory is simple and yet powerful enough to capture a variety of psychological data related to incubation and insight generation. Corresponding simulations again suggested that human performance in a variety of tasks was affected by implicit processing even when no attention was being paid to it and that insight might be the result of a continuous implicit process that emerged into consciousness. Future work should be devoted to the simulation of many more such tasks, as well as the simulation of regular problem solving to further substantiate the theory (and Clarion as its basis).

Overall, the simulation and explanation of the two tasks above—categorical inference and insight from intuition—were based on the Clarion framework of mixed rule-based reasoning, similarity-based reasoning, and implicit associative memory. The simulations have validated, to some extent, the overall framework, the theoretical postulates, and the mechanisms and processes of Clarion concerning human reasoning (see also Sun, 1994). A coherent set of mechanisms was posited. It provides explanations of many different types of reasoning patterns within a unified architecture. The simulation of the two tasks provides a glimpse into human reasoning in a way that is different from existing work. As mentioned before, it is a step toward accounting for human reasoning in a comprehensive and unified manner.

5.5. Capturing Psychological "Laws"

I will now address the capturing of some general psychological regularities, that is, some presumed psychological "laws" (as termed by Sun & Helie, 2013), as opposed to simulating individual empirical data sets from specific psychological experiments. The following discussion is largely drawn from Sun and Helie (2012, 2013) and Helie and Sun (2014).

Clarion captures many psychological "laws" of categorization, concept learning, induction, uncertain reasoning, decision making, and so on, beyond what many other models have been shown capable of. In each subsection below, several "laws" within a particular domain are discussed. For many additional psychological "laws" captured by Clarion but not explained here, see Helie and Sun (2010, 2014) and Sun and Helie (2012, 2013).

It should be emphasized again that the goal of this section is not to simulate specific psychological data sets, but to show, at a high level, that the Clarion framework can account intrinsically for many psychological "laws." Therefore, most of the following explanations are parameter free.

5.5.1. Uncertain Deductive Reasoning

As discussed earlier, reasoning is an important cognitive faculty that generates new ideas from existing ones. New ideas may be generated by the application of general rules to particular cases, which has traditionally

been termed deduction. Or they may result from generalization based on observations of instances, which has traditionally been termed induction.

Past research has shown that human reasoning, including deduction and induction, is often a mixture of rule-based and similarity-based processes, as discussed earlier. Under many circumstances, human reasoning is uncertain (i.e., not guaranteed to be correct). A number of prototypical cases of human reasoning were identified in Sun (1994). They can be captured within the NACS of Clarion.

A conceptual description of the cases and their explanations in Clarion are presented below (but more details may be found in Helie and Sun, 2014). The explanations below are parameter free.

5.5.1.1. Uncertain Information

It has been observed that, for human subjects, when information regarding the premise of a rule is not known with certainty, a conclusion may still be reached albeit with a corresponding amount of uncertainty (Collins & Michalski, 1989; Sun, 1994).

Clarion captures this phenomenon computationally. In Clarion, this phenomenon can be accounted for by rule-based reasoning (RBR) within the NACS (see Chapter 3). Uncertainty of information is captured by partial activation (< 1; as opposed to full activation 1). If a premise chunk node of a rule is partially activated, the corresponding conclusion chunk node is also partially activated, proportional to the activation of the premise chunk node, as indicated by the equations governing RBR in the NACS (Chapter 3).

5.5.1.2. Incomplete Information

When a rule has more than one premise, a conclusion can be reached (with uncertainty) even if only some of the premises are known (Sun, 1994). For example, one could have a rule: "If A and B, then C." If it is known that A is true while B is unknown, the conclusion C can still be reached although with some uncertainty.

Clarion captures this phenomenon computationally. In Clarion, this phenomenon is accounted for by RBR in the NACS. Within the NACS, each premise in a rule (represented by a chunk node) has a weight, and the weights of the premises add to one (or less). Thus, according the

RBR equations, when not all the premise chunk nodes are activated, the conclusion chunk node is partially activated, proportional to the number of activated premise chunk nodes. Therefore, a partially certain conclusion is reached within the NACS of Clarion.

5.5.1.3. Similarity

When no known information exists that answers a question directly, one can make an inference based on similarity to other known information (Sun, 1994). For example, when asked, "Is the Chaco a cattle country?" one answered "It is like western Texas, so in some sense I guess it's a cattle country" (Collins & Michalski, 1989; Sun, 1994). That is, an answer may be based on similarity matching.

This phenomenon is captured computationally by similarity-based reasoning (SBR) within the NACS of Clarion, through the interaction of the two levels (as explained in Chapter 3). When two chunk nodes share (micro)features, the activation of one chunk node is partially and automatically transmitted to the other through top-down and bottom-up activation flows via shared (micro)features (which fulfill similarity calculation; see Chapter 3). Specifically, the activation of the chunk node representing "Chaco" activates partially the chunk node representing "western Texas" (through top-down and bottom-up activation flows), which in turn (partially) activates all the rules associated with western Texas (e.g., "western Texas is a cattle country"). Thus, activating the chunk node representing "Chaco" leads to activating (partially) the chunk node representing "cattle country," proportional to the similarity involved.

One may view this case (and some other cases above and below) as a weak form of deduction. According to Clarion, similarity-based reasoning may approach full activation but never reach it, unlike rule-based reasoning, which may lead to full activation.

5.5.1.4. Inheritance

In inheritance reasoning, one uses information regarding a superclass to answer a question about a subclass (Collins & Quillian, 1969; Sun, 1994). For example, when asked if sparrows fly, one may respond "yes" because a prototypical bird flies.

In Clarion, inheritance reasoning is captured as a special case of similarity-based reasoning. In Clarion, chunk nodes along with their (micro)features, through cross-level interaction, can capture a categorical hierarchy without constructing a hierarchy explicitly (see Chapter 3; see also Sun, 1993, 1994, 2003). Chunk nodes representing subclasses (e.g., "sparrow") usually have all the (micro)features of the chunk node representing their superclass (e.g., "bird"), plus additional (micro)features that make them unique (the reverse containment principle; see Chapter 3). Thus, superclass-to-subclass inheritance is naturally captured and explained in Clarion by SBR applied to superclass-subclass relationships.

5.5.1.5. Cancellation of Inheritance

Superclass-to-subclass inheritance may be cancelled, if specific information exists that contradicts what may be inferred through inheritance. For instance, from superclass-to-subclass inheritance, one infers that, because prototypical birds do fly, ostriches fly too. However, specific information exists that ostriches, although they are birds, do not fly. In such a case, inherited information is cancelled.

According to Clarion, superclass-to-subclass inheritance is a special case of SBR. In fact, it is the most reliable form of SBR. But still it is not as reliable as rule-based reasoning (i.e., its resulting activation is always less than the full activation 1). Therefore rule-based reasoning at the top level can be used to cancel such inheritance, thus capturing exceptions.

More specifically, in Clarion, similarity matching alone cannot fully activate a chunk node, because the denominator of the similarity measure (as embedded in the bottom-up weights) is superlinear (as explained in Chapter 3). In contrast, rule-based reasoning can fully activate a chunk node. Therefore rules can be used in a way that rejects conclusions reached by similarity-based reasoning, thus canceling inheritance.

The reverse "inheritance" (from a subclass to a superclass), as well as its possible cancellation, can be explained in a similar fashion (for details, see Helie & Sun, 2014; Sun, 1993, 1994).

5.5.1.6. Mixed Rules and Similarities

In human everyday reasoning, rule-based and similarity-based reasoning (RBR and SBR) may be mixed in many different ways, and possibly in long

chains going from an initial piece of information (activation of a chunk node) to final conclusions (activation of other chunk nodes). Different mixtures of RBR and SBR, including possibly long chains of them, have been explored in Sun (1994, 2003) and Helie and Sun (2014).

In the NACS of Clarion, RBR and SBR can be chained in many different ways. For instance, as explained before, a chunk node can activate another chunk node by similarity matching, and the newly inferred chunk node (resulting from SBR) can then fire a rule (initiating RBR). The opposite can also happen: a chunk node activates another by applying a rule (RBR) and then another chunk node is activated by similarity to the rule-inferred chunk node (SBR). With multiple iterative cycles within the NACS, a long chain extending these instances is possible. For instance, different cases can be generated just by chaining the two cases above in different ways.

5.5.2. Reasoning with Heuristics

Heuristics are prevalent in human everyday reasoning (Tversky & Kahneman, 1974; Gigerenzer et al., 1999). People are often unsure about the validity of their conclusions. Still, conclusions are drawn when they are plausible based on heuristics.

Three important heuristics and their associated phenomena, along with the explanations of them by Clarion, are discussed below. One parameter is varied (i.e., $v_k^{c_j}$, the weight of feature k in chunk j).

5.5.2.1. Representativeness Heuristic

In empirical research, human subjects have been shown to have the tendency of using the representativeness heuristic (Tversky & Kahneman, 1974). That is, the estimated probability of a situation by subjects is often positively related to how well the situation represents prototypical situations, or to put it another way, how much the situation is similar to stored prototypes.

Clarion provides a computational account of this phenomenon (Helie and Sun, 2014). In the NACS, each prototypical situation is represented as a chunk node due to prior experiences. Each chunk node representing a prototypical situation is linked to a set of (micro)features in the bottom level that describe the situation. When a new situation is encountered, a chunk representing this new situation may not be present in the

NACS and therefore a corresponding chunk node may not be present at the top level of the NACS, but the corresponding (micro)features at the bottom level are activated by the stimulus from the situation. These (micro)features activate, in a bottom-up fashion, existing chunk nodes at the top level representing prototypical situations that are similar to the new situation (through similarity-based reasoning). The activated chunks may then be sent back to the ACS, along with their confidence levels (chunk node activations), which are then used for probability estimation. Chunk node activations, as we know, are proportional to the similarity between the new situation and the stored prototypical situations. Thus, subjects' probability estimates are based on the similarity between the new situation and the stored prototypical situations. Similarity-based reasoning within the NACS is responsible for the representativeness heuristic, because it leads to similar prototypical instances.

The representativeness heuristic has been used to account for several known biases in human reasoning (see Tversky & Kahneman, 1974, for a review). Some of the most well-known biases are described below.

Base-Rate Neglect

In a normative sense as prescribed by Bayes's theorem from probability theory, when estimating the probability that a particular person, Steve, is a librarian or a farmer, the total numbers of librarians and farmers should be considered. However, in many cases, human subjects do not consider this base-rate information but rely on the representativeness heuristic (Tversky & Kahneman, 1974); that is, they focus on whether the description of Steve is more representative of librarians or farmers. If Steve is more representative of librarians, the estimated probability that Steve is a librarian is higher than the estimated probability that Steve is a farmer (notwithstanding the actual probabilities).

In Clarion, what accounts for the representativeness heuristic also accounts for base-rate neglect. In the example above, chunks representing "farmer" and "librarian" (with their corresponding chunk nodes) exist in the NACS. However, "Steve" might not be represented by a chunk node in the NACS (because he was presumably mentioned for the first time), but the description of him activates a set of (micro)feature nodes in the bottom level. The (micro)feature overlaps (similarity) between Steve's description and the existing chunks representing "farmer" and "librarian" activate the two corresponding chunk nodes, through bottom-up activation flows:

$$s_f = s_{c_s \sim c_f} \times s_s = \frac{n_{c_s \cap c_f}}{f(n_{c_f})} s_s$$

$$s_l = s_{c_s \sim c_l} \times s_s = \frac{n_{c_s \cap c_l}}{f(n_{c_l})} s_s$$

where s_f is the activation of the "farmer" chunk node, s_l is the activation of the "librarian" chunk node, s_s represents the activation of all relevant features of "Steve," $s_{c_s \sim c_f}$ represents the similarity between "Steve" and "farmer", and $s_{c_s \sim c_l}$ represents the similarity between "Steve" and "librarian." The probability judgments made by the ACS are based on the confidence levels returned from the NACS, which result from the two chunk node activations above from similarity-based reasoning. Thus base rates are not considered.

Note that the activations of (micro)features in the bottom level, in this case, are the direct result of stimuli, which are not processed by an associative memory network (e.g., an attractor neural network). Clarion naturally displays base-rate neglect when no elaborate implicit processing (using, e.g., an attractor neural network) is performed.

Conjunction Fallacy

In human experiments, subjects may be asked to estimate the probability that Linda is a bank teller, and the probability that she is a feminist bank teller (Tversky & Kahneman, 1983). The results showed that subjects often estimated the former to be less probable than the later, even though the first category (i.e., "bank teller") includes the second (i.e., "feminist bank teller")—a clear violation of probability theory. According to Tversky and Kahneman (1983), this anomaly resulted from the application of the representativeness heuristic, because Linda's description was more similar to (more representative of) a prototypical feminist bank teller than an average bank teller.

In Clarion, the explanation of this phenomenon is similar to that for base rate neglect. Although Linda may not be represented by a chunk node in the top level of the NACS (because she was presumably mentioned for the first time), chunk nodes exist at the top level of the NACS representing the categories "bank teller," "feminist," "feminist bank teller," and so on (due to prior experiences). The description of Linda activates a set of (micro)feature nodes at the bottom level of the NACS, which activate existing chunk nodes through bottom-up

activation flows (computing the similarity between Linda and existing categories):

$$s_t = s_{c_l \sim c_t} \times s_l = \frac{n_{c_l \cap c_t}}{f(n_{c_t})} s_l$$

$$s_{ft} = s_{c_l \sim c_{ft}} \times s_l = \frac{n_{c_l \cap c_{ft}}}{f(n_{c_{ft}})} s_l$$

where s_t is the activation of the chunk node "bank teller," s_{ft} is the activation of the chunk node "feminist bank teller," s_l represents the activation of all relevant features of "Linda," $s_{c_l \sim c_t}$ is the similarity between "Linda" and "bank teller," and $s_{c_l \sim c_{ft}}$ is the similarity between "Linda" and "feminist bank teller." The activated chunks of the NACS are sent back to the ACS along with confidence levels. If Linda is more similar to "feminist bank teller" than "bank teller," the chunk node for "feminist bank teller" should have a higher activation, thus yielding a higher confidence level and a higher probability estimate. This process naturally explains the conjunction fallacy.

5.5.2.2. Availability Heuristic

In many psychological experiments, subjects often estimate the probability of an event based on the ease with which similar events can be retrieved from memory (Tversky & Kahneman, 1974).

Clarion provides a plausible computational account of this phenomenon (Helie and Sun, 2014). Assuming there is no relevant explicit knowledge, this may be a case of similarity-based reasoning (SBR). As explained earlier with regard to the representativeness heuristic, in the NACS, stimuli that are more similar to existing chunks yield higher activations of their chunk nodes. A higher activation of a chunk node makes the corresponding chunk easier to retrieve because it increases the probability that the chunk is chosen in stochastic selection. One of the activated chunks is stochastically chosen based on a Boltzmann distribution and sent back to the ACS (i.e., retrieved from declarative memory). This process may be repeated a number of times, and subjective probabilities may be estimated by the ACS based on the frequency of retrieval of similar items.

Of course, similarity is not the only factor affecting retrieval. Other factors, such as salience of (micro)features, strength of long-term memory (e.g., size of attractors), short-term priming (e.g., residual

activations), and so on, can also make retrieval easier or more difficult. For instance, in situations where free recall is involved, similar explanations involving attractors may apply (more below). Regardless of factors, the key in accounting for the availability heuristic lies in degree of retrievability.

The availability heuristic has been used in the literature to account for several known biases in human reasoning (Tversky & Kahneman, 1974). Two cases are discussed below.

Effectiveness of Search Set

Cues help memory search, and some cues are better than others. According to Tversky and Kahneman (1974), subjects tended to estimate relative probabilities of categories by trying to recall as many examples as possible from each category and assign a higher probability to the one that led to more recalls. For instance, the first letter of a word is a much better cue to recall the word than its third letter. When trying to decide whether more words start with the letter "r" or have "r" in the third position, subjects recall words with "r" in the first and the third position, and tend to respond (incorrectly) that there are more words with "r" in the first position because they can retrieve more such words (Tversky & Kahneman, 1974).

In Clarion, cued recall works by reasoning from the cues (Chapter 3). When there is no explicit rule available concerning the given cue, it is a case of similarity-based reasoning. Items in the NACS (e.g., stored words) are represented by chunk nodes at the top level and (micro)features at the bottom level. The bottom-up activation of a chunk node may be proportional to the number of its (micro)features that are activated at the bottom level. However, some (micro)features are more closely associated with the chunk node; that is, they have higher cross-level weights (because they constitute more salient features, as discussed in Chapter 3), and thus they are better cues for recall. When (micro)features with higher weights to a chunk node are activated by the cue, the activation of the corresponding chunk node is higher. Chunks that are more highly activated are more likely to be selected (based on a Boltzmann distribution) and sent back to the ACS (as the recalled item). Therefore, some cues are better than others. Subjects choose the response that corresponds to the better cue.

Retrievability of Instances

In a psychological experiment, subjects were read a list of man and woman names, and asked to judge if there were more man names or more woman names on the list (Tversky & Kahneman, 1974). The results

showed that when famous man names were included in the list, subjects estimated that there were more man names on the list (notwithstanding the actual number).

In accordance with the availability heuristic, subjects tried to recall names from the list and made an estimate on that basis. If they could remember more man names, they assumed that there were more man names on the list. Famous names were easier to retrieve from memory and therefore led to overestimation.

Clarion naturally accounts for this phenomenon. In Clarion, every time a name is seen or used, it is learned or relearned by the attractor neural network at the bottom level of the NACS. Famous names are learned more often (because they are seen more often, e.g., from media sources). The attractor neural network is affected by training frequency: each time a name is encountered, the corresponding attractor is strengthened (Chapter 3). Therefore, attractors representing famous names have larger attractor fields.

In Clarion, for free recall, memory search is initiated by random activations in the bottom level (Chapter 3). Attractors with larger attractor fields are more likely to be settled into and thus retrieved and sent back to the ACS (after bottom-up activation and stochastic selection described before). Therefore, famous names (with larger attractor fields) are more likely to be retrieved, and thus the corresponding category (e.g., "man names") yields a higher estimate.

5.5.2.3. Probability Matching

In psychological experiments, subjects' response frequencies tend to match the frequencies of their prior exposure to the stimuli associated with these responses, known as "probability matching" (see, e.g., Garnham & Oakhill, 1994). For instance, if subjects are asked which of two lights is going to be turned on next, the probability of choosing the first light corresponds to the relative prior frequency of this light being turned on.

Clarion accounts for this computationally. In the bottom level of the NACS, the attractor neural network implicitly encodes (summarizes) past experiences with the lights, with each light represented by a different attractor. Previous work has shown that this network accurately estimates the underlying probability distribution of the environment (Helie et al., 2006).

In the absence of cues (i.e., in free recall), memory search in the bottom level is initiated from random activations of the (micro)feature nodes (as mentioned before; see Chapter 3). The probability of the network settling into each attractor is determined by prior training: the more frequent an item has been seen, the larger the corresponding attractor is, and consequently the more likely it is settled into, matching roughly the prior frequency (Helie et al., 2006). Then, the bottom-up activation flows activate chunk nodes at the top level, which lead to stochastic selection of a response as described before. This computational mechanism provides an account of the human tendency to behave as probability matchers.

5.5.3. Inductive Reasoning

Inductive reasoning generates generalized conclusions from observation of instances (Heit, 2008). While this form of reasoning may be error prone, it is essential: it allows one to function in an environment by making plausible predictions and choosing actions accordingly.

According to Clarion, inductive reasoning relies on retrieval from the NACS (i.e., from declarative memory), and the retrieval is very much similarity-based (i.e., utilizing SBR). Below, a few well-identified phenomena are examined, along with their explanations based on the NACS of Clarion. In this subsection, only one parameter is varied to account for these phenomena (i.e., $v_k^{c_j}$—the relative weight of feature k in chunk j).

Note that a variety of relevant simulations of inductive reasoning have been carried out before within Clarion, although they are not described here.

5.5.3.1. Similarity between the Premise and the Conclusion

It has been observed that human inductive reasoning is affected by the similarity between the premise and the conclusion, not just based on logic that involves categorical relations and other relationships (Osherson et al., 1990; Rips, 1975; Sloman, 1993; Sun, 1994). For instance, subjects make stronger inference from rabbits to dogs than from rabbits to bears (Heit, 2008).

Clarion accounts for this phenomenon with its similarity-based reasoning. Clarion was successful in capturing and simulating human data

concerning SBR. One case was discussed earlier in Section 5.3. In Clarion, the similarity between chunks i and j is a function of the number of overlapping (micro)features. Specifically,

$$s_j^s = s_{c_i \sim c_j} \times s_i$$
$$= \frac{n_{c_i \cap c_j}}{f\left(n_{c_j}\right)} \times s_i$$

where s_j^s is the strength (activation) of chunk node j from similarity-based reasoning, s_i is the strength (activation) of chunk node i, $n_{c_i \cap c_j}$ is the number of (micro)feature overlap between chunks i and j (by default), n_{c_j} is the number of (micro)features in chunk j (by default), and f is the superlinear function (defined in Chapter 3). Assuming that the strength (activation) of the premise chunk node s_i is fixed, the strength of the conclusion chunk node is a function of the relative number of overlapping (micro)features between the premise and the conclusion chunk, or in other words, a function of the similarity between the two chunks. Clarion thereby captures the similarity effect in inductive reasoning.

5.5.3.2. Multiple Premises

In human experiments on induction, it has been observed that the number of premises affects the strength of the conclusion (Nisbett et al., 1983; Osherson et al., 1990). For example, the argument:

Hawks have sesamoid bones.
Sparrows have sesamoid bones.
Eagles have sesamoid bones.
⇒
All birds have sesamoid bones.

is stronger than the argument:

Sparrows have sesamoid bones.
Eagles have sesamoid bones.
⇒
All birds have sesamoid bones.

Clarion provides a computational account of this phenomenon. In the NACS, due to the use of *max* in calculating the overall activation, the

strength (activation) of a chunk node is monotonic and nondecreasing. The result of the *max* operation cannot be decreased by adding more arguments.

That is, the strength of conclusion chunk j resulting from similarity from a set of premise chunks $\{i_1, i_2, \ldots, i_n\}$ is (as described in Chapter 3):

$$s_j^s = \underset{k}{Max}\left[s_{c_{i_k} \sim c_j} \times s_{i_k} \right]$$

where s_j^s is the strength of conclusion chunk node j, s_{i_k} is the strength of premise chunk node i_k, and $s_{c_{i_k} \sim c_j}$ is the similarity between i_k and j. Therefore, adding premises maintains or increases the strength of the conclusion, as shown in human data.

5.5.3.3. *Functional Attributes*

Although inductive strength is correlated with similarity (as has been discussed thus far), it is not always that clearcut. Compare the following two arguments:

Chickens have a liver with two chambers.
⇒
Hawks have a liver with two chambers.

and

Tigers have a liver with two chambers.
⇒
Hawks have a liver with two chambers.

The first argument is stronger than the second. This is because chickens and hawks are more similar to each other than tigers and hawks, as was described earlier. However, consider the following two arguments:

Chickens prefer to feed at night.
⇒
Hawks prefer to feed at night.

and

Tigers prefer to feed at night.
⇒
Hawks prefer to feed at night.

In this case, the second argument is often judged to be stronger by human subjects in experiments, although tigers and hawks are less similar to each other than chickens and hawks. This result may be explained by feeding habits being more similar between hawks and tigers, because they are both predators, than between hawks and chickens. This phenomenon has been referred to as "exception to similarity due to functional role" (Heit, 2008).

Functional attributes, such as feeding habits, are treated as (micro)features in Clarion. They can be readily incorporated into what has been described earlier regarding (micro)features in Clarion. These functional (micro)features may be given larger weights when they are emphasized by the context (e.g., through metacognitive modulation as described in Chapter 4). In turn, in Clarion, functional attributes are part of the similarity calculation and affect the strength of the conclusion being reached (especially when they are given large weights), without any additional assumptions or mechanisms. The phenomenon concerning the second pair of arguments above is thus explained. Even though the literature on categorization suggests that it is unclear what constitutes a feature, the interpretation above seems a reasonable one.

5.5.4. Other Psychological "Laws"

The NACS of Clarion is also capable of accounting for a large number of other psychological "laws" in other psychological domains and functionalities. To avoid the tedium of technical details, I will not get into the explanations of how Clarion accounts for these. The interested reader is referred to Helie and Sun (2014) and Sun and Helie (2012, 2013).

Clarion accounts for many phenomena of human memory (Helie & Sun, 2014). For example:

1. *Frequency effect*: In free recall tests, higher frequency words are better recalled.
2. *Positive priming*: In lexical decision tasks, subjects are usually faster at identifying the second word if it is related to the first word (e.g., the word "butter" when it follows the word "bread").
3. *Negative priming*: In lexical decision tasks, longer reaction times are expected when the second word is unrelated to the first.
4. *Cue effects*: In cued recall, efficiency of recall increases with the number of cues.

5. *List length effect*: When a list to be memorized grows longer, there is a decline in performance, in both recognition and free recall tests.
6. *Serial position effects*: In free recall, items at the beginning of the list are more likely to be recalled (the primacy effect). Items at the end of the list are also more likely to be recalled (the recency effect).

Clarion also accounts for many psychological phenomena of categorization (Sun and Helie, 2012). For instance:

1. *Features in similarity*: Judgment of similarity is affected by the number of matching features, as well as by the number of non-matching features.
2. *Asymmetry of similarity*: Judgment of similarity is not always symmetric, and in fact it is often asymmetric.
3. *Reliability of categorization*: Stimuli that are more frequent are easier to categorize correctly.
4. *Fan effect*: Features that are consistently associated with a category facilitate categorical decisions.
5. *Base rate effect*: Subjects are more likely to assign a new stimulus to larger existing categories.
6. *Variability effect*: Subjects are more likely to categorize a new stimulus in a category with higher variance among its existing members.

Clarion can also account for salient characteristics of human decision making (Sun & Helie, 2012). Human decision making is concerned with preferences and choices, as studied in psychology, economics, and business administration. Among psychological models of decision making, decision field theory (DFT) can account for many psychological phenomena (Busemeyer & Johnson, 2008).

Clarion embodies DFT in its NACS. As explained before, the bottom level of the NACS consists of multiple neural networks that can be either auto-associative or hetero-associative (Chapter 3). It includes a hetero-associative neural network implementing DFT, devoted to decision making. As a result of implementing DFT in the bottom level of the NACS, Clarion accounts for many psychological phenomena of decision making that DFT accounts for (Busemeyer & Johnson, 2008).

Besides implementing DFT, Clarion also enhances DFT in terms of the range of phenomena that DFT can account for. By capturing the duality and the interaction of explicit and implicit processes, Clarion adds new dimensions to DFT. For example, it enables rule-based reasoning and similarity-based reasoning to be carried out, which could not be carried out within DFT alone. Rules at the top level can also be used to validate the decisions chosen by the DFT network. So Clarion accounts for additional phenomena in these regards. In addition, implementing DFT in Clarion also eliminates all the free parameters in DFT. See Sun and Helie (2013) for details.

For many more psychological "laws" accounted for by Clarion, not only in the NACS but also in the other subsystems, see Helie and Sun (2014) and Sun and Helie (2013). Some related simulations of these "laws" can be found in Sun (1994), Sun (1995), Sun and Zhang (2006), Sun, Slusarz, and Terry (2005), Helie and Sun (2010), and so on.

5.5.5. Discussion of Psychological "Laws"

One possible objection to accounting for psychological "laws" as done above would be that a cognitive architecture necessarily involves rather complex interactions among components and therefore properties of one component (such as being able to account for a psychological "law" within the NACS) may not hold after the interactions are taken into consideration.

To address this objection, one should take note of the fact that psychological phenomena are known to vary with contextual factors—prior experiences, individual circumstances, environmental conditions, instructions, task demands, presence of other individuals, and so on. Although some psychological phenomena are relatively more stable than others, all are subject to influences of contexts. One can only identify regularities within certain contexts (generic or specific), and hope to account for them within the same contexts.

From this perspective, Clarion is indeed capable of accounting for these "laws" despite the interactions among components. Given the context for any one of these "laws," the interactions within Clarion would be limited and identifiable, and therefore can be taken into consideration.

Cognitive architectures are meant to be the antithesis of specialized "expert systems": Instead of focusing on capturing performance in only

one task domain, they are aimed at providing a broad coverage of a wide range of domains. Therefore, for a cognitive architecture, it is important to be able to account for many phenomena simultaneously, not simply serving as a platform for building many separate "expert systems." Accounting for psychological "laws" accentuates this point.

Furthermore, work on cognitive architectures is not about throwing together a set of small models so that the resulting system can do all of whatever each of these small models is capable of. On the contrary, a major focus of Clarion has been about selectively including a minimum set of mechanisms, structured in a parsimonious but effective way, to account for a maximum set of psychological data and phenomena. That is, the focus lies in: (1) minimal mechanism, (2) maximal scope, and (3) effective integration that leads to synergy of various types. Many past simulations and computational analyses demonstrated this point (e.g., Sun, Slusarz, & Terry, 2005; Helie & Sun, 2010).

5.6. General Discussion

In this chapter, I have explored the procedural and declarative processes, both of which are essential to the human mind, through modeling and simulating a range of psychological experiments and through accounting for some generic psychological "laws." The chapter highlights the importance of the interaction between implicit and explicit processes, in human skill learning/performance and in human reasoning (i.e., in both procedural and declarative processes). It demonstrates this point through the cognitive architecture that captures these processes and their interaction. This chapter points to the usefulness of incorporating both explicit and implicit processes in theorizing about cognition-psychology in general.

Clarion serves as a unifying model of a variety of psychological tasks and data. In developing Clarion, a variety of data were examined, compared, and captured within the Clarion framework. In turn, the simulations based on Clarion revealed something further in these tasks and data. With detailed comparisons between human data and simulation results, these simulations shed light on plausible mechanisms and processes underlying human data.

The contribution of Clarion lies not only in capturing a range of human data through the interaction of implicit and explicit processes, but also in demonstrating the computational feasibility and psychological

plausibility of bottom-up learning, which complements the abundant treatment of top-down learning in the existing literature and fills a significant gap. Furthermore, the possibility of synergy (as well as detrimental effects possibly) that may result from the interaction has also been shown through examining human data and through simulation studies.

Some comparisons of models can be summarized here. In relation to modeling implicit learning (of either procedural or declarative knowledge), which is important to this cognitive architecture, there have been, in general, two types of computational models. The first type is neural network models (such as Cleeremans and McClelland, 1991), and the second type is stored data models, which can be instance-based, rule-based, fragment-based, or a combination thereof. Although these models are different from each other, they share some common characteristics: (1) learning is incremental, (2) learning is autonomous (in the sense that it is mostly not controlled by other processes) and generally "self-organizing," and (3) learning is sensitive to statistical structures in stimuli (Cleeremans et al., 1998). It is justified in adopting neural network models that have these features to capture implicit learning in Clarion, while adopting radically different mechanisms for capturing explicit learning. Alternative models are certainly conceivable, provided that the difference in accessibility between implicit and explicit knowledge (as argued in chapters 2 and 3) can be accounted for computationally.

Accounts of reasoning by Clarion and other models have been compared. To re-capitulate the main points, the Clarion framework of mixed rule-based reasoning, similarity-based reasoning, and intuition (through implicit associative memory) has demonstrated some cognitive-psychological plausibility, through the simulations described in this chapter, along with other studies published elsewhere (e.g., Sun 1995, 1995b; Sun and Zhang 2006; Helie and Sun, 2010). Compared with other existing models, Clarion embodies some different assumptions, most notably the separation of the two dichotomies (action-centered versus non-action-centered and implicit versus explicit). These alternative assumptions enable Clarion to capture a variety of reasoning data that could not be easily captured otherwise. Clarion points to new avenues of understanding human everyday reasoning, beyond the current psychology of reasoning, and capturing some essential patterns of such reasoning.

Intuition and insight are not a typical topic in cognitive science (although they have been investigated experimentally). Their prior explanation was not as clear as one would hope. Clarion is capable of capturing

and explaining this type of situation. In this regard, it is useful to posit the coexistence of two separate types of processes. Furthermore, the interaction between implicit and explicit processes, in the sense that intuition gives rise to explicit awareness and vice versa, is important (Helie & Sun, 2010).

Besides this two-level framework, can a one-level model capture all the data simulated here? It is conceivable that a one-level model may be designed so as to capture some data. Human data do not unambiguously point to the Clarion simulations described here. One may argue that if a one-level model can account for some data, then there is no need for two levels. However, it is seldom, if ever, the case that human data can be used to demonstrate the unique validity of a cognitive architecture. One needs to rely on converging evidence from a variety of sources to justify a model. By such a standard, Clarion fares well.

Alternatives notwithstanding, Clarion provides a consistent, theoretically motivated, and principled framework. It succeeded in interpreting many findings in skill learning/performance and reasoning that had not been adequately captured and explained before (such as bottom-up learning and synergy effects) and incorporated these phenomena into a unified model. This is where the potential significance of Clarion may lie.

Finally, I should note that many other tasks involve the interaction of implicit and explicit processes in either the ACS or the NACS. Some of these tasks have been accounted for by Clarion. With the use of the ACS, a number of serial reaction time tasks were tackled, as well as more complex tasks such as Minefield Navigation and Tower of Hanoi (e.g., Sun, 2002). With the use of the NACS of Clarion, artificial grammar learning tasks, incubation tasks, insight tasks, and so on were tackled (e.g., Helie & Sun, 2010). Tasks involving multiple individuals may also be accounted by the use of the ACS and NACS, as will be discussed in Chapter 7 (which focuses on social interaction).

In the next chapter, I will turn to describe simulations that heavily involve motivational and metacognitive processes (i.e., the MS and the MCS of Clarion).

6

Simulating Motivational and Metacognitive Processes

In this chapter, I will address the modeling and simulation of motivational and metacognitive processes, going beyond procedural and declarative processes as discussed in the previous chapter, thereby also capturing emotion, personality, and other aspects of human psychology that are believed to be rooted in the motivational underpinning of the human mind.

The next two sections (6.1 and 6.2) address metacognitive processes. Section 6.3 then deals with motivational processes. The subsequent three sections (6.4-6.6) discuss personality, moral judgment, and emotion, respectively, on the basis of the motivational and metacognitive mechanisms and processes.

6.1. Modeling Metacognitive Judgment

6.1.1. Background

As discussed before in Chapter 4, metacognition refers to processes concerning one's own cognitive processes, and includes monitoring and regulation, usually in the service of some objective (Flavell, 1976).

For a comprehensive cognitive architecture, well-developed metacognitive mechanisms are important, because they are an essential part of the human mind—without them the human mind may not function as well. I contended that metacognitive mechanisms should be an integral part of a cognitive architecture, despite the fact that most existing computational cognitive architectures lacked sufficiently complex, built-in metacognitive mechanisms (Sun, 2007b).

I will look into two metacognition-related human experiments from the literature. The first experiment, described in this section, taps into metacognitive monitoring, while the second, described in the next section, taps into both metacognitive monitoring and metacognitive intervention (control and regulation) on the basis of metacognitive monitoring.

6.1.2. Task and Data

In a task used by Metcalfe (1986), subjects were given a sheet of paper that described a story. They were asked to solve the puzzle in the story. They were told to write down a number between 0 and 10, where 0 meant that they were "cold" about the solution (i.e., having no idea at all about the solution) and 10 meant that they were certain that they had the right solution. They were to do so every 10s at the sound of a click. When the subjects had arrived at a solution, they were to write it down on a piece of paper.

The findings were that, in general, subjects who came up with correct solutions gave lower warmth ratings than did subjects with incorrect solutions. In terms of the last two warmth ratings before reaching a solution, this effect was statistically significant. In terms of the last warmth rating before reaching a solution, this effect was also significant.

Warmth ratings evidently reflected metacognitive monitoring—keeping an eye on one's own cognitive processes. However, the difference in warmth rating was counterintuitive—one would normally expect that subjects who came up with the correct solutions gave higher warmth ratings than did subjects with the incorrect solutions, but the result was the exact opposite. The question was how this result should be explained, in particular, how this result should be explained mechanistically, within the general framework of a cognitive architecture.

6.1.3. Simulation Setup

The simulation of this experiment, originally described in Sun et al. (2006), captured metacognitive monitoring and provided a detailed and plausible computational explanation of the counterintuitive experimental results of Metcalfe (1986).

The conceptual explanation of this experiment on which the simulation was based was that when a subject came up with multiple potential explanations and had to evaluate their relative merits, his or her subjective certainty (a metacognitive judgment) would be relatively low due to the coexistence of multiple potential explanations. Hence, a lower warmth rating was produced. But, in this way, the subject was more likely to come up with the correct (the most plausible) explanation eventually.

On the other hand, when a subject came up with only one plausible explanation, there was no need to evaluate multiple possibilities, and thus his or her subjective certainty would be higher. But that sole explanation was more likely to be wrong, because of the ambiguity of the situation and the lack of careful evaluation of all possibilities on the part of the subject (Metcalfe, 1986).

Within Clarion, the action-centered subsystem, the non-action-centered subsystem, and the metacognitive subsystem were involved in this task. The NACS performed inferences under the direction of the ACS. Through the monitoring buffer, the MCS kept track of the progress of inferences in the NACS (which might also perform metacognitive control when needed, although not in this particular task).

Specifically, first, the goal of performing the inference (*regular inference*) was set up by the MCS. The MCS then selected relevant input dimensions to be used for reasoning within the NACS (which excluded information not relevant to the task at hand). The MCS also selected the reasoning method to be used in the NACS: in this case, *forward chaining with SBR*.

The monitoring buffer in the MCS kept track of how certain the conclusions reached by the NACS were (among other things). The NACS section of the buffer recorded the relative strengths of n most highly activated conclusions (see Chapter 4). When the buffer reported that there was one conclusion that stood out with a high relative strength, the conclusion was considered certain and its "warmth" level was high. Otherwise, the conclusions were less certain, and the "warmth" levels

were lower. Hence, "warmth" was captured in this simulation by the relative strengths (*RS*s) in the monitoring buffer.

The ACS directed the reasoning of the NACS. The following rules existed at the top level of the ACS:

> If goal = regular-inference, then perform one step of inference in the NACS (using the method selected by the MCS and the information filtered by the MCS).
>
> If goal = regular-inference, and chunk i is a conclusion chunk with $S_{c_i} > \text{threshold}_S$ and $\forall j\ S_{c_i} > S_{c_j}$, then retrieve chunk i.
>
> If goal = warmth-reporting, then report the warmth (i.e., the RS) of the chosen chunk from the monitoring buffer in the MCS.

where *S* stood for chunk strength, and *RS* for relative strength. The threshold for strengths was set at $threshold_S = 0.1$. Although the bottom level of the ACS was present, it had little effect, because of the stochastic selection of levels in favor of the top level (which was the result of the task instructions).

At the top level of the NACS, knowledge was encoded as associative rules and chunk nodes. Some subjects (those who turned out to have higher warmth ratings) had few of these rules, while other subjects (those who turned out to have lower warmth ratings) had more of these rules. For simulating this experiment, associative rules were of the following form: if event A happens, then B might be the answer. At the bottom level of the NACS, an associative memory network (a hetero-associative network) was present. The network was trained with the same knowledge as expressed by the associative rules in the top level of the NACS.

6.1.4. Simulation Results

In this simulation, as in the human data, on average, those simulated "subjects" that generated correct solutions gave lower warmth ratings than those that generated incorrect solutions. Thus, the simulation, within the framework of Clarion, accounted computationally for the counterintuitive findings in the experimental data of Metcalfe (1986).

Specifically, in the simulation results, there were statistically significant differences between the two groups of simulated subjects. The average of the last rating of the simulated subjects with correct solutions was 3.3, while that of the simulated subjects with incorrect solutions was 5.2. The average of the penultimate rating of the simulated subjects with correct

solutions was 3.3, while that of the simulated subjects with incorrect solutions was 5.1. Statistical analysis of the last rating showed that there was a significant difference between correct versus incorrect. Similarly, analysis of the penultimate rating showed that there was also a significant difference between correct versus incorrect.

6.1.5. Discussion

The conceptual explanation of the data of this experiment was confirmed by the simulation. That is, when a subject initially came up with multiple potential explanations (when multiple relevant rules were available in the NACS), the subjective certainty was lower (due to the coexistence of multiple potential explanations). Thus, a lower warmth rating was produced. However, the subject in this case was more likely to come up with a correct explanation, based on evaluations of the relative merits of different potential explanations. On the other hand, if a subject initially came up with only one plausible explanation (e.g., when only one relevant rule was available in the NACS), there was no need to evaluate multiple possibilities, and thus the subjective certainty was higher, which led to a higher warmth rating. But that sole explanation was more likely to be wrong because of the ambiguity of the situation and the lack of evaluation of multiple possibilities.

6.2. Modeling Metacognitive Inference

6.2.1. Task and Data

In this case, instead of dealing with numerical data, protocols that indicated metacognitive reasoning were examined. An example from Gentner and Collins (1981) was as follows:

Q: Have you ever shaken hands with Richard Nixon?
A: No. . . .How do I know? It's not something that one would forget. I don't think I've ever seen him live, in person. I'm sure I haven't. (He went on describing meetings with some other presidents.)

Another essentially similar example was from Collins (1978):

Q: Is the Nile longer than the Mekong river?
A: I think so. . . .Because in junior high, I read a book on rivers. . .the Amazon was in there and the Nile was in there, and they were big, and long, and important. The Mekong wasn't in there.

In these examples, inferences were made based on (1) the lack of knowledge about something, and (2) the importance of that knowledge. In each of these cases, because of the combined reason of the lack of knowledge of a particular proposition on the one hand and the general availability of related knowledge on the other, an inference was made that the proposition was not true (cf. Gigerenzer, Todd, & the ABC Group, 1999).

To make such inferences, first, metacognitive monitoring of one's own reasoning processes is necessary, the same as in the previous task. However, beyond metacognitive monitoring, active metacognitive intervention is also necessary. Based on information gained from monitoring one's own reasoning (such as the lack of an important piece of information), a metacognitive process intervenes and redirects the reasoning, leading to a conclusion.

Beside the protocol data, Gentner and Collins (1981) also presented data of metacognitive reasoning from a third-person view. Assuming that a protagonist in a story forgot about an event, they asked subjects to rate the likelihood of the event, which was either of low or high importance. This scenario was essentially identical to the protocol segments above, except that there was an additional projection of one's own metacognitive processes onto others. In this experiment, the likelihood ratings were found to be inversely correlated with the importance of events, reflecting metacognitive monitoring and intervention.

6.2.2. Simulation Setup

The simulation of the protocol data aimed to capture metacognitive reasoning from the lack of information (Sun et al., 2006); that is, it aimed to capture both metacognitive monitoring and metacognitive intervention (control/regulation).

It was reasonable to hypothesize that a negative conclusion was drawn only if subjects thought that they knew enough about a domain and yet they did not know about a particular proposition in that domain. If they did not know enough about a domain when they did not know about a particular proposition in that domain, a negative conclusion was not likely to be drawn.

Within Clarion, the ACS, the NACS, and the MCS were involved in this simulation. The NACS performed inferences under the direction of the ACS. The MCS selected relevant information and reasoning methods to be applied within the NACS. The MCS also monitored the progress of inferences in the NACS and performed metacognitive intervention accordingly, including starting "lack-of-knowledge" inferences.

Specifically, the goal of *regular inference* was first set up by the MCS. It then selected relevant input information to be used (excluding information not relevant to the task at hand). Then the MCS selected the reasoning method to be used in the NACS, *forward chaining with SBR*. As always, the ACS directed the reasoning of the NACS (using the selected reasoning method). When the lack-of-knowledge condition was detected (as indicated by uniformly low activation in the NACS performance section of the monitoring buffer of the MCS), the MCS initiated lack-of-knowledge inferences by setting up the goal of *LOK inference*. The LOK inferences were then carried out by the NACS (under the direction of the ACS).

At the top level of the ACS, the following rules were used for directing reasoning of the NACS:

If goal = regular-inference, then perform one step of inference in the NACS (using the method selected by the MCS and the information filtered by the MCS).

If goal = regular-inference and chunk i is a conclusion chunk with $S_{c_i} > \text{threshold}_S$ and $\forall j\ S_{c_i} > S_{c_j}$, then retrieve chunk i.

If goal = LOK-inference, there is no conclusion chunk with $S_{c_i} > \text{threshold}_S$ but there are many associative rules pointing to the conclusion chunks, then indicate that the conclusion is negative.

If goal = LOK-inference, there is no conclusion chunk with $S_{c_i} > \text{threshold}_S$ and there are not many associative rules pointing to the conclusion chunks, then indicate that the conclusion is indeterminate.

where *S* represented chunk strength. The threshold for *S* was set at $threshold_S = 0.1$. Another threshold determined how many rules constituted "many" (which was domain specific). In this simulation, a value of 2 was used. In the ACS, although the bottom level was present, it had little effect, because stochastic selection in favor of the top level was used.

At the top level of the NACS, relevant knowledge was coded as associative rules and chunks. The associative rules relevant to this task were generally of the following form:

River A → long-river
River B → long-river
River C → long-river

along with other rules that were not relevant to this task.

At the bottom level of the NACS, one hetero-associative memory network was present. The network was trained with the same knowledge as the associative rules in the top level.

6.2.3. Simulation Results

The simulation successfully captured the lack-of-knowledge inference as shown by the human subjects in the protocols described earlier.

As predicted, when a simulated subject had a relatively large amount of knowledge about a domain but could not reach a conclusion in a particular instance, then the lack-of-knowledge inference was initiated, and a negative answer was produced. On the other hand, when a simulated subject had a small amount of knowledge about a domain and could not reach a conclusion in a particular instance, then no conclusion was drawn. A large number of simulation runs testified to this outcome. In all of these cases, metacognitive monitoring led to metacognitive intervention, which led to lack-of-knowledge inferences. This simulation demonstrated not only metacognitive monitoring but also metacognitive intervention (control and regulation).

6.2.4. Discussion

Clarion includes specifically metacognitive mechanisms for monitoring, controlling, and regulating cognitive processes. The metacognitive subsystem may select information, adjust cognitive parameters, and intervene in regular cognitive processes.

As indicated by the simulation results above, Clarion succeeded in accounting for, computationally in a detailed way, the counterintuitive results in the experimental data of Metcalfe (1986) as well as the data of Gentner and Collins (1981). These metacognitive simulations, as described in this and the previous section, captured rather accurately the existing experimental data.

To some extent, the two simulations above validated metacognition as embodied in Clarion. The cognitive architecture contains detailed built-in metacognitive mechanisms on which the simulations were based. The explanation of Metcalfe (1986), based on amount of relevant knowledge, naturally fell out of Clarion. Similarly, the lack-of-knowledge inferences in Gentner and Collins (1981) were also naturally captured by Clarion.

So, metacognitive processes were, more or less, architecturally specified in Clarion.

The simulations not only help to better understand issues of metacognition, but also have implications for further development of cognitive architectures. Notably, most existing cognitive architectures do not include a sufficiently complex metacognitive component. In contrast to most existing cognitive architectures, in Clarion, metacognitive processes are architecturally specified to a large extent. They are architecturally specified to the extent that I believe is appropriate: that is, they are sufficiently detailed but yet flexible. Work in this area is not only useful but very much needed. The metacognitive subsystem developed in Clarion might be applied to other cognitive architectures (there has indeed been some work in that direction). As understanding of metacognitive processes grows, the metacognitive mechanisms in Clarion may be further refined to capture the exact range and scope of human metacognitive processes.

One specific point that can be derived from Clarion and the simulations based on Clarion is that metacognitive processes are intermeshed with regular processes. They interact with each other on a constant basis. Thus, they may be viewed either as separate or as integrated—both views are partially descriptive of the Clarion perspective on metacognition. In general, in Clarion, the relationship among different types of processes is highly interactive. Given this highly interactive relationship, one may view metacognitive processes as one with regular processes, because they are tied intimately together.

Norman and Shallice's (1986) view on metacognition may be related to the Clarion view. They posited the coexistence of two kinds of metacognitive processes: (1) fast, automatic processes, which are triggered by stimuli and are inflexible; and (2) slow, conscious processes, which are independent of stimuli and are flexible. The former is used in skill performance, while the latter deals mostly with novel situations. Although the explicit-implicit dichotomy in Clarion (which applies to the metacognitive subsystem as well as other subsystems) is similar to Norman and Shallice's (1986) view, Clarion further addressed more complex forms of implicit and explicit metacognitive mechanisms and processes, and advocated intermeshed metacognitive and regular cognitive processes, which differed somewhat from their view (Sun and Mathews, 2012; Sun et al., 2006).

One argument might be that, although many cognitive architectures do not have built-in metacognitive mechanisms, some of them allow metacognition to occur on the basis of regular cognitive mechanisms, and such simpler cognitive architectures provide "deeper" explanations. However, there are severe limitations in those cognitive architectures in terms of the range of metacognitive phenomena that they can capture. Therefore, being simpler is not necessarily better.

6.3. Modeling Motivation-Cognition Interaction

6.3.1. Background

Motivation and cognition interact with each other (e.g., Simon, 1967; Ryan and Deci, 2000; Locke and Latham, 2002; Markman & Maddox, 2005). Effects of motivation on cognitive performance have been observed empirically. For instance, performance motivation, such as a difficult performance target, affects actual performance (Kanfer and Ackerman, 1989; Locke and Latham, 2002). Relatedly, anxiety affects performance as well, often negatively (Lambert et al., 2003; Wilson et al., 2010).

An individual's confidence in meeting a performance target (i.e., "self-efficacy") is believed to be important (Bandura, 1997). In cases that the individual has low self-efficacy (i.e., the individual has low confidence in his or her ability of meeting the target), anxiety may develop and performance may be affected (Brooks et al., 2012). It has been suggested that in case a difficult performance target is present and self-efficacy is high, attention will be allocated to the achievement of the target and performance improves. However, when self-efficacy is low, a difficult performance target may negatively affect performance. In the latter case, the negative effects may (at least in part) be attributed to anxiety (Brooks et al., 2012).

In this regard, empirical studies have suggested two possible outcomes as a result of elevated anxiety levels. The first possibility is that, in situations where anxiety levels are elevated (but not extremely high), individuals may move toward more explicit (more "controlled") processing (i.e., "explicit monitoring theory"; e.g., Baumeister, 1984; Beilock & Carr, 2001). This increase in explicitness has been shown to have two possible effects. In situations where tasks are naturally more explicit, performance

may not be hurt (or may even be improved). However, in situations where tasks are either very well learned or naturally tend to be more implicit, focusing additional explicit attention may have a negative effect on performance (Beilock & Carr, 2001).

The second possible outcome of elevated anxiety levels is that, especially in situations where anxiety levels are very high, cognitive resources may be allocated away, for example, to attend to the anxiety (i.e., "distraction theory"; e.g., Wine, 1971; Lambert et al., 2003). Accordingly, task-related decisions must be made using more implicit (more "automated") processes (referred to as "losing control" by Lambert et al., 2003). As a result, task performance is typically hindered (Lambert et al., 2003; Beilock et al., 2004; Wilson et al., 2009, 2010). However, for well-practiced or naturally implicit tasks, performance may be unchanged (Beilock et al., 2004). Some claimed that anxious individuals might perform equivalently on some tasks by exerting more explicit cognitive control in order to compensate for the distraction.

Another suggestion was that individuals might be distracted not by anxiety per se but by worrying about the situation and possible consequences (Wine, 1971). A related suggestion was that individuals might be distracted by additional metacognitive observation and evaluation (Kanfer & Ackerman, 1989). Distraction may also lead to different outcomes in different situations (e.g., Lewis & Linder, 1997).

While these theories seem contradictory, Wilson et al. (2009, 2010) has argued that they can actually be unified with an inverted U curve. The idea of an inverted U curve is that a slightly higher arousal (due to a certain degree of anxiety) may often lead to more explicit processing. But much higher arousal may often lead to the opposite results—more implicit processing (Wilson et al., 2009, 2010; Brooks et al., 2012). This phenomenon is related to what was addressed in such early work as Yerkes and Dodson (1908) as well as their more recent variations.

In Clarion, the fundamental hypothesis, discussed in Chapter 4 and consistent with the inverted U curve, is that when anxiety increases, it leads an individual to become more explicit (more "controlled") when making action decisions, relying more on explicit processes at the top level of Clarion. However, when anxiety reaches a certain higher level, it can begin reducing "control" and the individual reverts to more implicit ("automated") processes, relying more on the bottom level of Clarion. Depending on the level of anxiety and the specific characteristics and

dynamics of a task, the effect of anxiety can either enhance or degrade performance.

Specifically, according to Clarion, when anxiety levels are elevated, it forces individuals to first shift their action decision making toward more explicit processing; however, if anxiety levels are further heightened, individuals can instead become more implicit. This increase or decrease in explicit ("controlled") processing has the effect of enhancing explicit or implicit processes and hindering their opposites, thus affecting performance, positively or negatively, depending on tasks involved and other circumstances. As discussed earlier, when a task is naturally more implicit, increase in explicit processing (e.g., as a result of slightly elevated anxiety levels) may have a negative effect. When a task is naturally more explicit, increase in explicit processing may have a positive effect. Decrease in explicit processing (e.g., as a result of extremely high anxiety levels) can have similarly opposite effects on tasks that are either naturally more implicit or naturally more explicit. Furthermore, Clarion assumes that for those individuals with high self-efficacy, anxiety should generally be lower than for individuals with low self-efficacy (Brooks et al., 2012).

Within the Clarion framework, the strength of avoidance-oriented drives (see Chapter 4) is hypothesized to capture the level of anxiety (Wilson et al., 2009). As discussed, for instance, by Carver and Sheier (1998), it is more difficult to avoid things; hence, given high strengths of avoidance-oriented drives, stress, anxiety, and so on arise. This link between avoidance-oriented drives and anxiety (and possibly other negative feelings) may be forged by evolution (Smillie et al., 2006). It is thus hypothesized that the strength of avoidance-oriented drives determines the probability of selecting explicit processes at the top level of Clarion (i.e., amount of explicit processing, or degree of cognitive "control"), through an inverted U curve (as discussed earlier; Wilson et al., 2009, 2010; Brooks et al., 2012).

As hypothesized in Brooks et al. (2012), the *gain* parameters within the drive strength equation (Chapter 4) are relevant. The *gain* parameters for avoidance-oriented drives (e.g., g_s where s = *avoidance*) are negatively correlated with the theoretical notion of self-efficacy, while the *gain* parameters for approach-oriented drives (e.g., g_s where s = *approach*) are positively correlated with self-efficacy. Therefore, according to the hypothesis, when self-efficacy is high, approach-oriented drives are more likely to be highly activated; when self-efficacy is low, avoidance-oriented

drives are more likely to be highly activated and thus the anxiety level is more likely to be high.

Self-efficacy as embodied by the *gain* parameters can be determined on the fly during task performance, based on the internal state of the MS and the MCS, the processes of the ACS (e.g., number of relevant explicit rules, randomness of action selection, and so on), and the processes of the NACS, in relation to a performance target.

The *stimulus* parameter in the drive strength equation (for both approach- and avoidance-oriented drives) represents the external conditions relevant to drive activation, including possibly a performance target (e.g., as externally assigned and provided to individuals). For example, a low *stimulus* value may result from a no pre-set performance target condition and a high *stimulus* value from a high pre-set target condition. Given a high *stimulus* value, it is likely that some drives, of either an approach or an avoidance type, will be highly activated, depending on self-efficacy (i.e., the corresponding *gain* parameters). In contrast, given a low *stimulus* value, drives of both types are less activated.

Individual differences also result from the drive *deficit* parameters within the drive strength equation. For example, Kanfer and Ackerman (1989) suggested that high-ability (and thus very likely high self-efficacy) individuals were more likely to set difficult goals for themselves without an externally imposed performance target. Thus, they might have higher *deficit* values for (some of) their approach-oriented drives. Note that the drive *deficit* parameters are determined internally regardless of drive *stimulus* (e.g., resulting from externally assigned targets). Higher *deficit* values lead to corresponding drives being more highly activated.

In addition, individual ability differences are captured within Clarion by differences in terms of the neural network learning rates, the rule learning thresholds, the probability of rule encoding, the temperature in a Boltzmann distribution, the level integration parameters, and so on.

Below I will present one detailed example of cognition-motivation interaction, in the form of elevated anxiety levels affecting sensory-motor performance, which draws upon Wilson et al. (2009). Then, some other tasks will be briefly sketched as well. Clarion is capable of providing a mechanistic and process-based explanation of motivation-cognition interaction. For instance, the implicit-explicit distinction has been referred to in different ways by different researchers (often with somewhat different meanings). The Clarion framework provides some clarification to these terms in a more exact, mechanistic way.

Beyond terminological issues, Clarion also provides a detailed computational account that helps to shed new light on underlying motivational and cognitive processes as well as their interaction.

6.3.2. Task and Data

Look into the golf-putting task of Beilock and Carr (2001). In their experiment 3, subjects were instructed to hit a golf ball at a target from different positions. Then, following training, they were presented with a high-pressure scenario (aimed at causing elevated levels of anxiety presumably). The results indicated that subjects who were trained in a single-task condition experienced performance degradation in a high-pressure post-test, while those who were trained in a "self-conscious" condition did not suffer performance degradation.

Specifically, subjects initially had little or no golfing experience. They were randomly assigned to the single-task condition, the self-conscious condition, or the dual-task condition. Here only the single-task and the self-conscious condition are considered. The objective was to putt a golf ball as accurately as possible from nine locations on a carpeted indoor putting green. The locations were 1.2, 1.4, or 1.5 meters from the target. All subjects putted from the nine locations in the same randomly determined order. The target was a red square, on which the ball was supposed to stop.

Each subjects completed 270 training putts, which were divided into three blocks of 90 putts each. The mean distances of the first 18 and last 18 training putts, respectively, were recorded. The training was followed by an 18-putt low-pressure (presumably low-anxiety) post-test and then an 18-putt high-pressure (presumably high-anxiety) post-test.

During the training phase, subjects in the self-conscious group were informed that they would be filmed by a video camera and the resulting videotapes would be reviewed later to gain an understanding of how individuals learned to putt. The camera was set up on a tripod that stood directly in front of the subjects. The camera was turned on and pointed at them during the training phase. After the training phase, the camera was turned off and pointed away.

The low-pressure post-test was the same for all groups. To subjects in the single-task condition, the 18 low-pressure putts following the training seemed just another set of putts. Subjects in the self-conscious

condition were made aware that the camera had been turned off and pointed away.

The high-pressure post-test was also identical for all groups. Subjects were informed of their mean putting performance for the last 18 putts during the training phase and were then provided with a scenario designed to create high pressure. Specifically, they were told that if they could improve accuracy by 20% in the next set of putts, they would receive $5. However, each subject was told that he/she had been randomly paired with another subject. In order to win the money, both had to improve by 20%. Each subject was told that the other had already improved by the required 20%. Each subject then took an 18-putt post-test.

The results from this experiment were as shown in Figure 6.1 (Beilock & Carr, 2001). Subjects' performance worsened during the high-pressure (high-anxiety) post-test for the single-task group but not the self-conscious group. Detailed statistical analysis confirmed the results (Beilock & Carr, 2001).

At first glance, Beilock and Carr's results might be explained by explicit monitoring theory. That is, it might be postulated that performance degradation resulted from increased explicitness. Performance pressure might elicit step-by-step explicit monitoring and control over complex, well rehearsed, implicit procedures that would be more automatic if such efforts had not intervened (Beilock & Carr, 2001).

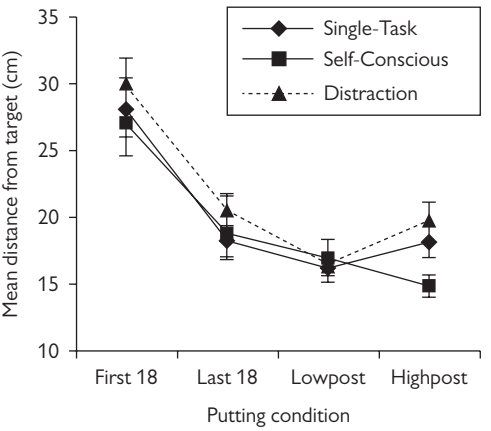

Figure 6.1. The human data from Beilock and Carr (2001).

According to this theory, when a task that is "automated" (i.e., implicit and procedural) is affected by certain situations (e.g., situations involving high pressure and thus high anxiety), execution of the task becomes more explicit. This "over-thinking" phenomenon may hamper performance. In Beilock and Carr's opinion, this happened when performance degradation under pressure occurred in the putting task. Practice under the self-conscious condition served to mitigate this tendency.

On the other hand, distraction theory assumes that involving a certain level of explicit processing is likely to be better. Explicit processes, when combined with implicit processes, often lead to better results (i.e., the idea of synergy; Sun, Slusarz, & Terry, 2005). In contrast, implicit processes are often faster but more susceptible to inaccuracies and mistakes (Reber, 1989). They are also more reactive in nature. It is reasonable to suggest that in usual circumstances, people tend to prefer acting in a somewhat more precise (i.e., somewhat more explicit) fashion than in a purely reactive and uncontrolled (i.e., purely implicit) manner (Sun, 2002). However, distracting contexts (e.g., anxiety) may hamper explicit processes (for a variety of possible reasons as discussed earlier), leading to more implicit processing, which may often hurt performance (e.g., Lambert et al., 2003).

The latter theory (distraction theory) can be derived directly from the basic postulates of Clarion. The former theory (explicit monitoring theory) can be implemented within Clarion (Wilson et al., 2009, 2010).

Within the Clarion framework, it is natural to hypothesize that subjects' performance worsened when faced with distracting contexts (e.g., anxiety), because they were prevented from using a sufficient amount of explicit processing. In this regard, it can be reasonably assumed that performance in the golf-putting task by those aforementioned subjects under the aforementioned experimental conditions was not completely implicit. While putting might (or might not) be an implicit task for beginning novices, it becomes somewhat more explicit with practice (as explicit rules for performing the task were extracted, or received from external sources). The notion of bottom-up rule extraction (see Chapter 3) has been explored within the Clarion framework and is pertinent here (Sun et al., 2001).

Specifically, in the putting task, a novice does not have much information on how to effectively putt. However, through trial and error,

the individual begins to learn implicit skills, and acquires explicit rules to help to increase accuracy as training continues. The improvement in performance during practice is, in part, the result of explicit rules being established (in addition to implicit skill learning; Sun et al., 2001). As the number of rules increases and they become more refined, accuracy improves. Experienced golfers may have a large set of explicitly accessible rules that can be recalled relatively easily. (However, there might be an inverted U curve here also: a gradual increase of explicit knowledge as experiences accumulate and then some decrease when one becomes a true expert. See, for example, the relevant arguments from Dreyfus & Dreyfus, 1987.)

Of course there is no guarantee that explicit knowledge that one possesses is actually used in action decision making, as opposed to post hoc rationalization. Judging from prior work, there are reasons to believe that at least some of that explicit knowledge is indeed used for actual action decision making (Mathews et al., 1989; Willingham, Nissen, & Bullemer, 1989; Sun et al., 2001). In general, people prefer to perform tasks in a somewhat explicit fashion (mixing implicit and explicit processes to a certain degree). Within the Clarion framework, the notion of synergy has been advanced in this regard as a possible explanation: mixing implicit and explicit processes leads to better performance than using either alone (though explicit processes are more effortful; Sun, Slusarz, & Terry, 2005). This synergy may be a reason to use both implicit and explicit processes in most skill domains (Sun, 2002).

6.3.3. Simulation Setup

Now look into how Clarion was applied to the simulation of the golf-putting task (Wilson et al., 2009). In the MS, one primary drive might be particularly relevant: "honor," which was roughly the need to avoid blame in this case (see Chapter 4). The drive strength was obtained using a (pre-trained) Backpropagation network. The inputs to the network specified the experimental conditions (i.e., the *stimulus* parameter) and the individual difference variable that indicated an individual's predisposition toward becoming anxious (i.e., the *deficit* parameter, capturing "trait anxiety" in this case).

For the single-task group, during the training phase, the drive strength was determined by: $tanh(.1x)$ (where x is the individual difference

variable; $0 \leq x \leq 5$)[1]. This function was also used for the low-pressure post-test of both the single-task and the self-conscious group (because these situations were essentially the same). During the training phase of the self-conscious group, the drive strength was determined by: *tanh(.15x)*, because of the (presumably) higher anxiety levels due to the self-conscious training condition.

During the high-pressure (high-anxiety) post-test, for those trained in the self-conscious condition, the function changed to: *tanh(.17x)*, in response to the anxiety-inducing cues. For those trained in the single-task condition, during the high-pressure (high-anxiety) post-test, the function changed to: *tanh(.5x)*, in response to the anxiety-inducing cues. The assumption was that the drive strength of the self-conscious group during the high-pressure post-test increased to be only slightly higher than that used during the training phase, because the simulated subjects trained in the self-conscious condition were exposed to an anxiety-inducing situation during training for an extended period of time, and therefore the effect that the high-pressure post-test had was mitigated to a large extent.[2] These subjects were affected, but the effect was not as strong as for those trained in the single-task condition where no mitigating factor was present during training. A graphical representation of the drive strengths was as shown in Figure 6.2.

The MCS determined the "proportion" of explicit versus implicit processing in the ACS. The MCS mapped the maximum avoidance-oriented drive strength, using an inverted U curve (as discussed before), to explicitness of processing, that is, the probability that the simulated subject would use the top level of the ACS when performing the task. The output was produced by a (pre-trained) Backpropagation network, with the input to the network being the maximum avoidance-oriented drive strength from the MS. Figure 6.3 shows a graphical representation of this, where the MCS selects probabilities between 0 and 1 based on: $-0.4x^2 + 0.2x + 0.6$ (where x is the maximum avoidance-oriented drive strength).

The ACS was set up the same way for all simulated subjects. The bottom level of the ACS included a Backpropagation network with input

1. The drive strength so determined corresponded to a drive strength equation (Chapter 4) with its *deficit*, *stimulus*, and other parameters that might vary from individual to individual.

2. This might be explained by the depletion of the *deficit* of the drive, as a result of prolonged exposure to an anxiety-inducing situation, which partially mitigated the increase of the *stimulus* of the drive (as a result of the high-pressure post-test).

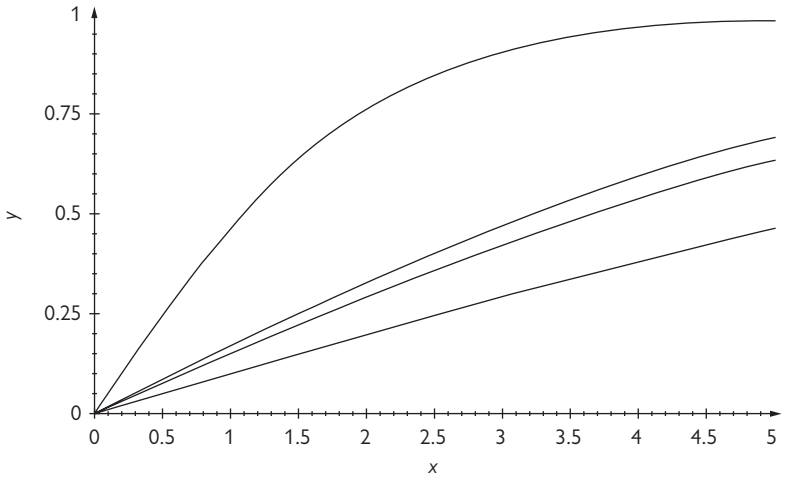

Figure 6.2. The *x*-axis represents individual difference ($0 \leq x \leq 5$); the *y*-axis represents drive strength ($0 \leq y \leq 1$). The topmost function is for the single-task high-pressure post-test. The second function is for the self-conscious high-pressure post-test. The next function is for the self-conscious training. The bottommost function is for the single-task training, the single-task low-pressure post-test, as well as the self-conscious low-pressure post-test.

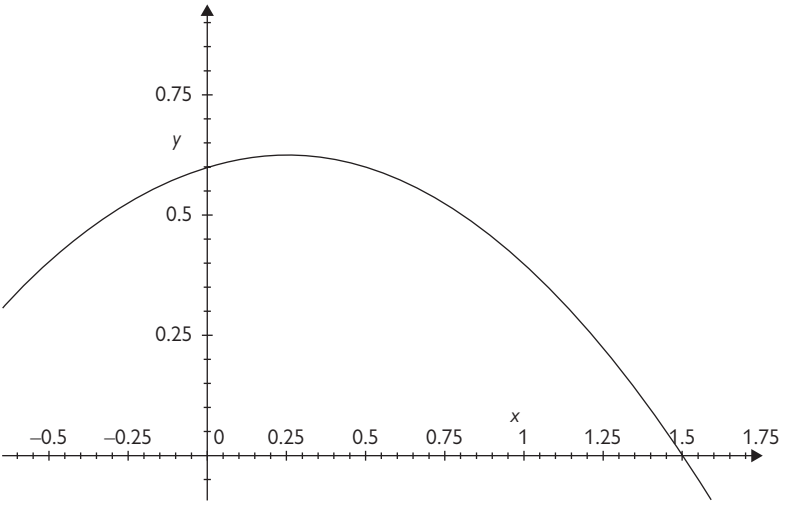

Figure 6.3. The *x*-axis represents the maximum avoidance-oriented drive strength from the MS ($0 \leq x \leq 1$); the *y*-axis represents the degree of explicit processing determined by the MCS for the ACS ($0 \leq y \leq 1$).

nodes representing information concerning the putting positions and their distances to the target. The three output nodes represented putting actions: *swing easy*, *swing medium*, and *swing hard*. The network started with no a priori knowledge, and learned through performing the task (using simplified Q-learning as explained earlier). Eventually, implicit knowledge of putting was captured by the network.

At the top level of the ACS, no explicit rules existed at the beginning of the task, because the subjects in this experiment had little or no prior golfing experience or knowledge. Rules were extracted from the bottom level of the ACS during the course of training (a rule was extracted when an action caused a putt to land within five centimeters of the target; see RER in Chapter 3). The ACS attempted to generalize the rules after they were extracted (with the RER algorithm).

The accuracy (i.e., the distance of the ball from the target) was calculated based on a prespecified function. The MCS sent reinforcement signals determined on that basis to the ACS for reinforcement learning.

6.3.4. Simulation Results

In this simulation, as in the original human experiment, the accuracy of the first 18 and the last 18 putts of the training phase was recorded, along with the 18 putts for each of the two post-tests (Wilson et al., 2009).

The simulated subjects of both the single-task and the self-conscious condition improved with practice; statistical analysis showed a significant effect of practice and no training condition/practice interaction, which was consistent with Beilock and Carr's (2001) human data.

In the simulation, accuracy in the low-pressure post-test was essentially the same between the simulated single-task and the simulated self-conscious group, the same as in the human data. In the high-pressure post-test, a statistically significant difference existed between the two simulated groups, the same as in the human data. In addition, there was a statistically significant interaction of training condition and post-test. This finding also matched that found by Beilock and Carr (2001).

Direct analysis of putting performance within each simulated group showed that the accuracy of the simulated single-task group significantly declined from the low-pressure to the high-pressure post-test, as in the human data. The accuracy of the simulated self-conscious group did not change significantly between the two post-tests, although the direction

of the change suggested a slight improvement, as consistent with Beilock and Carr (2001).

The simulation results were as shown in Figure 6.4. Looking at the figure, it is evident that the results from the simulation match the human data very closely. This suggests that the detailed, mechanistic, and process-based interpretation based on Clarion of the human results may have merit.

While explicit monitoring theory described earlier may seem an intuitively appealing explanation for performance degradation in low-level tasks like putting, this simulation points to the fact that explicit monitoring may not be the only viable explanation. Explicit processing requires more effort and control than implicit processing. When an individual is anxious or distracted, the amount of control or the level of effort that he or she has available might be negatively impacted. This might reduce the individual's ability to utilize explicit processes. Explicit monitoring theory points to "over-thinking" as the culprit of performance degradation under pressure. However, what occurred might not be "over-thinking" but might be an inability to engage explicit processes to a sufficient extent, as suggested by the simulation.

As has been pointed out earlier, there may be a difference between relative novices (using a mixture of implicit and explicit processes) and

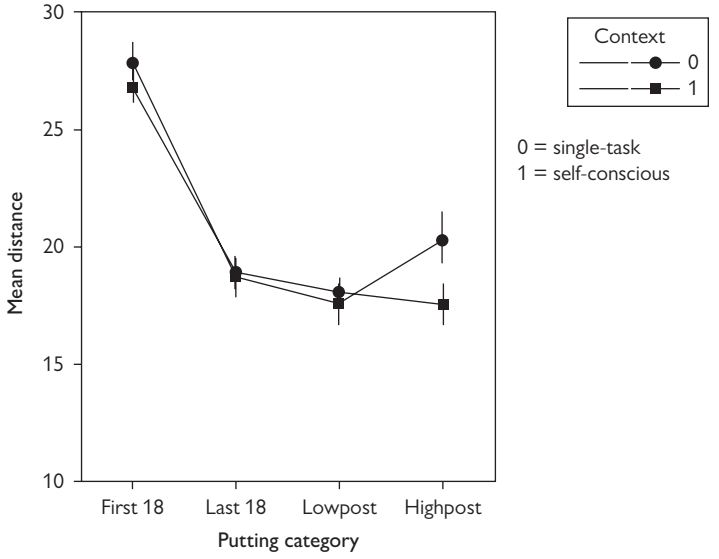

Figure 6.4. The simulation of Beilock and Carr (2001).

true experts (being equally proficient in either a somewhat explicit or a completely implicit mode, or even performing the best when completely implicit). In this regard, in the literature, highly practiced and thus automated expert skills show negative effects of explicit monitoring under some circumstances, while less well-developed skills may show effects of distraction in the sense of distraction theory. There is no sufficient evidence to conclude that those human subjects in this experiment had reached the true expert level; for example, what appeared to be asymptotic performance might turn out to be a temporary performance plateau that, with further training, might lead to still better performance. Therefore there is no conclusive evidence that "over-thinking" hurts their performance.

These simulation results above suggest that Clarion might be used to interpret at least some performance degradation phenomena seen in experiments involving sensorimotor tasks, as well as in many other types of tasks (especially higher-level tasks; Wilson et al., 2009, 2010).

6.3.5. Discussion

Clarion provides a computational account of the phenomenon of performance degradation under pressure on the basis of motivation. While the suggestion that motivation (e.g., drives) affects performance is not novel, the work described here has taken a step toward explaining exactly how and in what way performance is affected by motivational and environmental contexts. Within the Clarion framework, anxiety, as a function of the task context, affects implicit and explicit processes. The simulation provides a glimpse into how motivation acts upon cognitive processes, and it does so in a quantitative, process-based, and mechanistic way.

Clarion addresses the interaction between motivation and cognition, and in this way, it explains or substantiates some previous theories naturally. Moreover, Clarion may eventually provide a more general and yet detailed picture of self-regulation and control, in a generalized sense, in a mechanistic, process-based way.

It should be mentioned that a number of other tasks were also simulated and explained in a similar or related fashion. For example, Wilson et al. (in preparation) explored similar effects with regard to mathematical problem solving (namely, a modular arithmetic problem). Performance degradation under pressure in that domain was also captured and

simulated by elevated anxiety levels (as a result of avoidance-oriented drive activations). Anxiety led to reduction in explicitness of processing following the inverted U curve, thus affecting performance, consistent with distraction theory. For another example, Wilson et al. (2010) explored the stereotyping task of Lambert et al. (2003), with the inverted U curve resulting from anxiety. The simulation of this task explained, in a mechanistic, process-based sense, increased stereotyping biases under pressure as resulting from reduced explicitness of processing in response to anxiety (avoidance-oriented drive activations). For yet another example, Brooks et al. (2012) explored the effects of assigned performance targets in the Kanfer-Ackerman air traffic control task. The effects were explained using the inverted U curve, along with the interpretation of the notion of self-efficacy as outlined earlier. The explanations offered by these simulations contributed to the understanding of motivation-cognition interaction (but note that they did not rule out other possible motivation-cognition interactions in these or other circumstances).

6.4. Modeling Human Personality

6.4.1. Background

A generic and comprehensive cognitive architecture should be able to computationally capture and explain human personality as studied in social-personality psychology. This is because personality, as has been argued by many (Caprara & Cervone, 2000), involves many aspects of the mind and likely emerges from the interaction of many mechanisms and processes of the mind, all of which should have been included in a truly generic and comprehensive cognitive architecture. Therefore, a generic and comprehensive cognitive architecture should be well equipped to account for human personality, without much elaboration or any significant addition.

With Clarion, personality is captured based on an adequate representation of basic human motivation, and related motivational, metacognitive, action selection, and other processes. Such representations and processes capture the interaction of internally felt needs and external environmental factors in determining goals and actions by individuals, which are arguably the key to personality.

A model of personality within Clarion, based on the architectural features of Clarion, can be briefly summarized as follows: various subsystems (and various components within) interact continuously within an individual (including the ACS, the NACS, the MS, and the MCS). Within the MS, a set of basic drives are more or less universal across individuals, but individual differences are explained, in a large part, by the differences in drive activations (strengths) in different situations by different individuals. Different arrays of drive strengths (activations) lead to setting of different goals as well as setting of different major cognitive parameters by the MCS. Individual differences in terms of drive activations are consequently reflected in the resulting goals as well as major cognitive parameters. On the basis of the goals set and the major cognitive parameters chosen, an individual makes action decisions, within the ACS. Thus their actions reflect their fundamental individual differences (as well as situational factors) as a result. Their actions in turn affect the world in which they act.

The relative invariance of personality has been argued for in social-personality psychology (e.g., Caprara & Cervone, 2000; Epstein, 1982; Murray, 1938). Clarion can capture the relative invariance within an individual in terms of behavioral propensities and inclinations at different times and with regard to different situations (social or physical), through the relatively stable mechanisms and processes of motivational and other subsystems (including their relatively stable parameters), in addition to capturing behavioral variability.

Conversely, I would also argue that an adequate model of personality must be a comprehensive cognitive architecture. As has been argued by many (Cervone, 2004; Shoda & Mischel, 1998; Caprara & Cervone, 2000), personality is not a standalone "mechanism" or a separate "process." It is likely emergent from a complex system involving many psychological mechanisms and processes (Sun and Wilson, 2011; Sun and Wilson, 2014). In order to adequately account for human personality, a comprehensive cognitive architecture that captures most, if not all, psychological functionalities would be required (at a minimum). To put it another way, a model of personality must be a comprehensive model of human psychology.

Past work on personality measures (e.g., John & Srivastava, 1999) provides some evidence for a set of essential personality dimensions, known as the Big Five (the Five-Factor Model): *Extroversion*, *Neuroticism*, *Agreeableness*, *Conscientiousness*, and *Openness to Experience*. Despite

controversies, it is thus far the best-established line of work on this issue. So the Big Five can be used as a starting point for developing a model of personality.

However, the Big Five does not provide a model of the underlying mechanisms, nor the processes emerging from them, that generate this structure. Work on the structure of personality and work on the processes and mechanisms of the mind have developed largely separately. However, it is important to explain how psychological processes/mechanisms and personality structures relate to one another (Shoda & Mischel, 1998; Read et al., 2010).

There were a few process models of personality. Among them, Shoda and Mischel (1998) developed a recurrent neural network model of personality that captures personality in terms of "cognitive affective units" (such as goals, plans, and behaviors), and used the neural network to explore the underlying dynamic processes of personality. Read et al. (2010) presented a model involving a more complex neural network model to address some of the personality structures of the Five-Factor Model.

One general weakness of most existing personality models was that the motivational representations and processes specified in these models were often ad hoc. These models usually did not incorporate well-developed theories of motivation (e.g., Murray, 1938; Reiss, 2004). There was often no detailed account of how motives were triggered or how they interacted dynamically once triggered. Although biologically inspired neural network models were used, existing work was mostly not based on any comprehensive cognitive architecture (without any significant addition or modification).

Another general weakness of many existing personality models and theories was their conflation of reflexive and deliberative processes (i.e., implicit and explicit processes). Hence there was the "perplexing complexity" of empirical findings with respect to these models and theories (Smillie et al., 2006).

Yet another weakness was that these existing models often did not attempt to quantitatively capture real empirical data. Thus it was often unclear whether they could match any human data.

The aim of the Clarion personality model is to capture major aspects of the personality structure within a generic cognitive architecture. The rationales for developing the Clarion personality model follow from the brief review above:

- A personality model should be situated within a comprehensive framework of the mind (i.e., a cognitive architecture, without any significant addition or modification).
- A personality model should have well-developed, precisely specified mechanistic and process details (Sun, 2009b).
- A personality model should be based on a well-developed model of essential human motivation (Sun, 2009).
- A personality model should make contact with actual empirical data and capture and explain such data.

6.4.2. Principles of Personality Within Clarion

6.4.2.1. Principles and Justifications

Within the Clarion framework, a number of basic principles of human personality were identified (which together constituted the drive-goal-action theory of personality, described in Sun & Wilson, 2011; Sun & Wilson, 2014, 2014b). Below I describe these principles and present some brief justifications.

Principle 1
Human personality should emerge from the interaction among various components of the mind. That is, computationally, it should emerge from the interactions among various subsystems and modules of a cognitive architecture. The cognitive architecture should allow the emergence of different personality types, as well as the adaptation of personality (at least to some extent) through experience.

Principle 2
Among various processes of the mind, personality is especially rooted in the motivational processes. Among them, it is rooted in implicit drives, but also in explicit goals resulting from drives (possibly stochastically).

Principle 3
Action decision making (i.e., procedural processes, in both implicit and explicit forms), on the basis of the goal chosen and the situational inputs (possibly stochastically), is also important to personality. It is an integral part of personality, even though it is subject to learning and adaptation.

Principle 4
Declarative knowledge and reasoning (in both implicit and explicit forms) affect personality through affecting actions, although their effects are less direct.

Below, I present some justifications of these principles. Because these principles are derived from the Clarion framework, the justifications are relatively brief.

Justification for Principle 1

As touched upon earlier, it seems obvious that personality, a theoretical construct, should be the result of the existing psychological mechanisms and processes, and nothing else (Cervone, 2004). A cognitive architecture, by definition, should include all essential psychological components, mechanisms, and processes of the human mind. Within the cognitive architecture, the interactions among different subsystems and various modules within should be able to generate psychological phenomena of all kinds, which include personality-related phenomena (Sun & Wilson, 2014b). Thus, personality, if it is a valid psychological construct, should be accounted for by the cognitive architecture.

Similarly, Cervone (2004) argues that personality results from a complex system with dynamic interactions among interconnected processes. Personality should be understood by reference to basic cognitive processes that give rise to overt patterns of behavior. Mayer (2005) made similar points.

In addition, a model of personality should capture details of psychological processes (e.g., more than previous work on personality). Thus, it is necessary to go beyond abstract (somewhat ungrounded) notions of goals, plans, resources, beliefs, or cognitive affective units. It is one thing to argue abstractly that personality traits consist of configurations of goals, plans, resources, beliefs, or cognitive affective units, it is quite another to map personality traits to concrete, detailed, and grounded psychological processes and mechanisms. It is therefore useful to ground personality in a cognitive architecture so they are explained in a deeper and more unified way, along with many other psychological phenomena, based on the same primitives within a generic cognitive architecture.

Also, coupled with the account of learning in a cognitive architecture, a model of personality can potentially account for the emergence, shaping, and tuning of personality. Such explanations can be deeper than previous work, and more unified with models of other psychological functionalities.

Justification for Principle 2

Fundamental behavioral traits (i.e., personality) may map onto essential motivations, that is, onto drives in Clarion, because drives are the most fundamental (Chapter 4). Other processes may be more transient, due

to environmental factors, learning, and adaptation. Although drives are "tunable" also, they are, relatively speaking, more fundamental and more stable than other processes. Therefore, it is reasonable to ground personality, first and foremost, in drives, and then also in goals and actions on their basis.

Relying heavily on motivational representations to account for human personality has been justified from a variety of perspectives: philosophy, psychology, and computational modeling (Sun & Wilson, 2014b). For example, Schopenhauer (1819) contends that the ultimate principle is "Will"—the mindless, nonrational urge at the foundation of being. Schopenhauer asserts that one's body is given in two different ways—as representation (objectively, externally) and as Will (subjectively, internally). The action of the body is nothing but the act of Will objectified. Existence is, in essence, endless striving with blind impulses, which has precedence over reason and rationality. Such blind impulses (i.e., implicit drives) define the essential human condition.

Similarly, according to Buddhism, desire (similar to Schopenhauer's Will) lies at the root of human existence (as well as human suffering). According to Buddhism, life is a never-ending flow of desire, which one cannot stop (at least not easily). Therefore, everyday life is a transient, impermanent sequence of circumstances driven by various changing desires. These two schools of thoughts above, viewed at an abstract level, are similar to the Clarion view of the fundamental role of motivation.

To further justify this approach, the psychology literature on personality and motivation can also be examined. Existing work shows how personality traits can be closely related to human motivation. Deci (1980) made an elaborate case for this point, comprehensively reviewing the literature on motivation and personality and arguing for their close relationship. Reiss (2010) argued that "everybody is motivated by the . . . basic desires, but people prioritize them differently. Every person has his or her own hierarchy, which is highly correlated to normal personality traits . . . A powerful predictor of behavior in natural environments is how a person prioritizes the . . . basic desires." Shoda and Mischel (1998) also argued that personality could be understood in terms of cognitive-affective units, for example, goals, plans, expectancies, and so on. Some computational details were worked out, showing how individual differences in personality might emerge on the basis of cognitive-affective units (although no exact structural mapping was produced).

It should be emphasized that a broader perspective is also needed, besides the role that motivation plays in personality. Personality involves a variety of psychological mechanisms and processes beyond motivation (although they may be less important). Therefore, personality types, besides being mapped onto motivational structures, representations, and processes, are also mapped (to a lesser extent) onto other mechanisms and processes. The determination of personality types involves various motivational, cognitive, metacognitive, and other parameters.

Justification for Principle 3

Here the notion of action is defined in a broad sense, including, for instance, both physical actions and mental actions. Actions (behaviors) are the ultimate measure of personality. Without it, there would be no objective way of observing and classifying personality types. Therefore, action selection on the basis of goals is important to measuring personality. Moreover, it is also an important part of personality, because given the drives and goals, different actions may be used to address them.

For example, consistent with Principle 1, the personality trait of being dominating is captured by a drive state where dominating others is emphasized (Principle 2), a specific goal being chosen, actions for achieving that goal being carried out, and reasoning related to that goal applied (Principle 4 later). In terms of actions (behaviors), a dominating person, when in relevant situations and reacting to relevant cues, exhibits dominating behaviors at a higher frequency and/or intensity than an average person. On the basis of such behaviors (actions), that person is viewed as a dominating person.

However, a necessary condition is the learning by an individual of the connection between a goal and proper actions to achieve the goal within a given sociocultural and physical environment. Action selections (procedural processes) are learnable, as almost universally accepted (Sun, 2002), and they are subject to sociocultural influences.

Justification for Principle 4

Cervone (2004) argued for the importance of belief, schema, appraisal, reasoning, and so on, that is, declarative processes involving declarative knowledge, as determinants of personality.

As an example, the personality trait of being dominating is captured by a drive state where dominating others is emphasized, a goal of dominating others being chosen, actions for achieving that goal being carried out,

and so on. But beyond that, reasoning regarding the goal and the actions can also be carried out, for example, regarding whether one's actions would actually achieve the goal. Such reasoning, as well as the declarative knowledge on which the reasoning is based, is relevant to behavior choices and therefore to personality.

However, declarative processes (with declarative knowledge) can impact personality only on the basis of drives and goals, and affect procedural processes (action decision making) only indirectly. Declarative knowledge and processes are learnable and flexible, and subject to sociocultural influences.

6.4.2.2. Explaining Personality

Based on these principles outlined above, Clarion provides explanations of issues and phenomena of personality. For Clarion to serve as a model of personality, it must be capable of explaining many issues and phenomena, especially the structures of personality.

Within Clarion, action decisions are made by the ACS, but the action decisions are based on the current goal, which is (mostly) set by the MCS based on the drives in the MS. Therefore, drives in the MS are the foundation of behavior, according to Clarion. The actions are ultimately directed by the flow of "desires" (drives), that is, various impulses on a moment-to-moment basis. Therefore, it is natural to ground the notion of personality primarily within the MS of Clarion. This is consistent with the earlier argument that personality traits are largely motivationally based, so that personality reflects largely the dynamics of the underlying motivational processes (Sun, 2009).

Accounting for Trait Stability

In Clarion, as external situations change, behaviors (actions) can vary across situations, because the inner working within the MS, the ACS, and so on changes with the external situations. However, that does not mean that there cannot be stable personality traits. According to Clarion, the relatively stable structures and contents of the motivational and other subsystems capture relatively stable individual differences in behavioral inclinations and propensities (i.e., relatively stable personality traits), through differences in parameters (based on principles 2, 3, and 4). Therefore, Clarion, with its four subsystems, is capable of providing an account of stable personality traits (as well as an account of the behavioral variability across situations, as further explored below).

Accounting for Person-Situation Interaction

Past debates highlighted the importance of person-situation interaction in personality (Caprara & Cervone, 2000). Maslow (1943) argued that "the situation or the field in which the organism reacts must be taken into account but the field alone can rarely serve as an exclusive explanation for behavior. . . . Field theory cannot be a substitute for motivation theory." According to Clarion, person-situation interactions can occur through the interactions between the relatively stable characteristics of the motivational and other subsystems of an individual and the influence of situations (which are more transient).

In Clarion, activations of drives are the results of relatively stable structures and parameters of the MS (capturing some relatively stable personality traits), as well as stimuli received from situations that are transiently present on a moment-to-moment basis. This is important according to Principle 2. Which goals are activated at any given moment is a (possibly stochastic) result of the competitive interaction among drives (resulting in part from situational inputs) and which goal "wins" that competition (Principle 2). Behaviors are then determined (possibly stochastically) based on both the goal and the situational inputs through the competition of different possible actions (Principle 3). Furthermore, in Clarion, reciprocal interaction needs to be noted: individuals develop in interaction with the world that is partly shaped by their own actions (Bandura, 1997). Therefore, personality traits and situations do interact in Clarion.

Accounting for Individual Behavioral Variability

Clarion also provides for the possibility of within-person variability. As has been argued by some, within-person variability over time may be as high as between-person variability (Caprara & Cervone, 2000). Such variability is consistent with Clarion. In Clarion, the process of person-situation interaction and the concomitant competitions (e.g., the drive competition or the action competition, which may be stochastic and are important according to principles 2 and 3) result in varying behaviors both across situations and over time (in a stochastic way).

Accounting for Personality Structures and Types

Structural personality models focus on various relatively stable personality types and dimensions such as the Big Five (Digman, 1990; John & Srivastava, 1999; McCrae & Costa, 2010). Within Clarion, as discussed earlier, the structures and the contents of the motivational and other subsystems capture

stable individual differences in behavioral inclinations and tendencies—that is, personality traits (principles 2, 3, and 4). Clarion provides both an account of stable traits, as well as an account of behavioral variability across situations and over time.

In Clarion, personality structures and types are, first and foremost, the result of interaction and competition among drives activated by situational factors. However, there may not necessarily be a direct relationship between the characteristics of a single drive and a hypothesized personality dimension or trait (e.g., as is consistent with the view of Smillie, Pickering, & Jackson, 2006). Personality structures and types are also, among other things, the result of individually different processes of goal setting on the basis of drives, and the result of individually different processes of action selection on the basis of goals and situations.

Based on the ideas above, the major dimensions of personality (e.g., the Big Five) can be captured in Clarion. Clarion posits that individual differences in drive activation, goal setting, action selection, and so on can translate into behaviors indicative of different personality types. For simulations that computationally demonstrate this point, see the subsequent subsections (see also Sun & Wilson, 2011; Sun & Wilson, 2014; Sun & Wilson, 2014b).

Accounting for Sociocultural Influence on Personality

Personality is in part the result of sociocultural factors (D'Andrade & Strauss, 1992; Dweck, 2008). Clarion allows for sociocultural influences to take place. In Clarion, drive activation (based on situational inputs), goal setting (based on drive competition), action selection (based on the goal chosen and situational inputs), and so on can be adapted, tuned, or learned to various extents, as discussed in chapters 3 and 4. Such adaptation, tuning, or learning allow sociocultural factors to enter into personality.

One possibility of sociocultural influences is through reinforcement signals that underlie reinforcement learning (Montague, 1999; Sun et al., 2001), which may be socioculturally generated. For example, being humble is highly regarded by some societies and therefore receives positive reinforcement, while in some other societies it is negatively reinforced. Furthermore, situational changes (as a result of actions performed) may also be socioculturally determined or influenced. For example, being angry may bring about quite different results in different cultures. Adaptation, tuning, or learning taking place on the basis of such socioculturally specific

feedback lead to sociocultural influences on personality. (See details of learning and adaptation in chapters 3 and 4.)

Accounting for Approach Versus Avoidance Behavior

It has been argued that the approach and avoidance systems are separate and operate differently, and they are important for the extroversion and the neuroticism dimension of the Big Five, respectively (Gray and McNaughton, 2000). Clarion can account for the approach and avoidance distinction in personality.

First, it is necessary to be able to increase or decrease avoidance behavior, or to increase or decrease approach behavior. Within Clarion, such manipulations are possible, because there are two corresponding sets of drives in the MS, approach-oriented versus avoidance-oriented, and thus their parameters can be adjusted independently (see Chapter 4).

Second, according to Cacioppo, Gardner, and Berntson (1999), in the absence of situational cues, there is a tendency toward approach behavior. However, as situational cues become stronger, the avoidance system responds more strongly toward negative cues than the approach system does toward positive cues. Within Clarion, with the drive strength equation (Chapter 4), when situational cues are absent, the *baseline* parameters play a big (or sole) role in drive activations. However, when situational cues become stronger, the *baseline* parameters become less important, and situational cues play a bigger role in determining drive activations (especially when the drive *gain* parameters are large). Therefore, the drive *baseline* parameters may be such that approach drives have higher baselines than avoidance drives. Likewise, the drive *gain* parameters for avoidance drives may be higher than those for approach drives. The two sets of parameters together thus capture the observed phenomena (Sun & Wilson, 2014b; Read et al., 2010).

Accounting for Psychopathology

Clarion is able to account for some mental disorders, including personality disorders (such as the obsessive-compulsive personality disorder, the narcissistic personality disorder, and so on; Samuel & Widiger, 2008).[3] According to Clarion, these disorders may be largely attributed to the motivational subsystem (Principle 2; see also Reiss, 2008), but

3. The interested reader is referred to "Diagnostic and Statistical Manual of Mental Disorders, Fourth Edition, Text Revision" (DSM-IV-TR) published by American Psychiatric Association in 2000, or its recently updated version DSM-5 (which came out after this work had been completed).

not exclusively so, because other subsystems can play some roles and different disorders may differ with regard to their loci.

For example, to account for the obsessive-compulsive personality disorder, high activations of certain avoidance-oriented drives were posited (such as *conservation*, *avoiding danger*, and so on). The high activations of these drives lead to corresponding goals, which in turn lead to corresponding actions, that is, symptomatic obsessive-compulsive behaviors (Sun & Wilson, 2014b).

Such explanations seem simplistic without delving into details. However, detailed models and simulations have indeed been developed that substantiate such explanations; see, for example, Sun, Wilson, and Mathews (2011) and Sun and Wilson (2014b). So far, of course, Clarion does not yet provide definitive explanations for these disorders. Much more work would be needed. However, the work so far suggests that Clarion is relevant to understanding these disorders.

6.4.3. Simulations of Personality

To test these ideas, a detailed personality simulation model was implemented in Clarion. The Clarion personality model was mostly based on drive *deficit* parameters (Chapter 4), because given the current inputs (hence the drive-specific *stimulus* levels), the drive strengths were primarily determined by drive *deficit* parameters (along with *gain* and *baseline*). On that basis, goal setting and action selection took place. Thus, within Clarion, personality involved a variety of parameters within the various subsystems, with the MS being the most important part. Some key details are described below for each simulation test.

6.4.3.1. Simulation 1

Simulation Setup

This simulation test was designed to show that the Clarion personality model could respond reasonably with appropriate behaviors to different situations, and at the same time show sufficient behavioral variability (Sun, Wilson, & Mathews, 2011; Sun & Wilson, 2014b).

Within the MS, a (pre-trained) neural network generated drive strengths based on the drive strength equation (Chapter 4), within

Table 6.1. Drive deficits corresponding to seven personality types.

Drives	Sociable	Shy	Confident	Anxious	Responsible	Lazy	Generic
Food	0	0	0	0	0	0	0.1
Water	0	0	0	0	0	0	0.1
Sleep	0	0	0	0	0	0.5	0.05
Avoiding Danger	0	0.2	0	0.6	0	0.3	0.2
Reproduction	0.2	0	0.3	0	0	0	0.1
Avoiding the Unpleasant	0	0.6	0	0.7	0	0.7	0.2
Affiliation and Belongingness	0.9	0.2	0.3	0.6	0.2	0	0.6
Recognition and Achievement	0.5	0	0.8	0	0.8	0	0.2
Dominance and Power	0	0	0.7	0	0.2	0	0.2
Autonomy	0	0.3	0.6	0	0.7	0.2	0.5
Deference	0	0.7	0	0.8	0	0.3	0.3
Similance	0.5	0.8	0	0.8	0.1	0.8	0.7
Fairness	0.2	0	0	0.3	0.5	0	0.1
Honor	0.5	0	0.3	0.2	0.8	0	0.5
Nurturance	0.6	0	0	0	0.3	0	0.4
Conservation	0	0.4	0	0.6	0.7	0.1	0.3
Curiosity	0.6	0	0.5	0	0	0	0.4

which initially $deficit_d$ was set as shown in Table 6.1 for the generic type,[4] $baseline_d$ was set to be proportional to the corresponding initial deficit (i.e., $baseline_d = 0.1 * deficit_d$), and $stimulus_d$ was determined from the scenarios (Sun & Wilson, 2014b).

The drive strengths from the MS were sent to the MCS. A (pre-trained) neural network in the MCS generated goal strengths according to the goal strength equation (Chapter 4). The goal strengths were then turned into a Boltzmann distribution and the new goal was chosen stochastically from the distribution. The goals used in the simulations were as shown in Table 6.2.

The chosen goal was input to the ACS. The ACS also received sensory inputs indicating the current situation. The bottom level of the ACS was trained to determine Q values of actions. At the top level, *Rule Extraction and Refinement* (RER) was involved. The outputs by the ACS were turned

4. The deficit of a drive was "decayed" at each step using a multiplicative factor ($decay_d = 10\%$), when the drive was addressed (i.e., when the goal corresponded to the drive). This was a simplification for the sake of this simulation. See Chapter 4 for more details.

Table 6.2. The goals determined by the MCS.

Eat	Be self
Drink	Follow
Rest	Mimic
Flee	Be fair
Pursue sex	Follow code
Avoid	Be caring
Fit in	Organize
Stand out	Explore
Lead	

into a Boltzmann distribution and one behavior (action) was stochastically chosen from that distribution (see Table 6.3 for behaviors).

Fifteen scenarios, as shown in Table 6.4 (adapted from Read et al., 2010), were used to test the model. These scenarios were represented using a set of situational features (as shown in Table 6.4).

Each scenario was presented to the model for 100 time steps and the chosen behavior by the model at each step was recorded. The process was repeated 100 times, representing 100 different simulated "subjects" (each time with different random initial weights in neural networks).

Simulation Results

The percentage of appropriate behaviors for each of these scenarios was recorded, for the one most frequently chosen behavior and for the top three most frequently chosen behaviors, respectively. A behavior was considered appropriate if its Q value as output by the ACS was above a threshold (.5) given the current scenario and the most plausible goal for that scenario.

The most plausible goal was calculated based on the generic personality (see Table 6.1) to determine which goal was the most likely to be chosen given the scenario. This allowed the determination of the appropriateness of behaviors without having any access to the actual goals being chosen internally by the simulated "subject."

The results, averaged over all 100 simulated "subjects," were as shown in Table 6.5. As shown, according to both measures (top 1 and top 3), behavior choices were substantially more appropriate than chance (≈ 20%). The results provided some evidence for the appropriateness of the model, in terms of both accuracy and variability.

It was useful to see how sensitive the model was to noise in the representation (features) of the scenarios. So the 15 scenarios were again tested over 100 steps and for 100 runs. However, noise was added to the scenarios by flipping the values of two of the input feature nodes (randomly chosen) in the ACS for each scenario. The chosen behavior at each step was recorded and the appropriateness measures were calculated.

Table 6.3. List of behaviors (each with a letter code and a numerical index).

Eat/drink	Stay at periphery	Help others with work	Ensure work distributed fairly
E/D (0)	SP (11)	HOW (22)	EDF (33)
Drink alcohol	Self-disclose	Order others what to do	Wear something distinctive
DA (1)	*SD (12)*	OO (23)	WSD (34)
Relax	Ask others about self	Dance	Steal
R (2)	AO (13)	D (24)	S (35)
Play practical joke	Talk politics	Ask other to dance	Kiss up
PPJ (3)	TP (14)	AOD (25)	KU (36)
Tease/make fun of	Gossip/talk about others	Ask for date	Be cheap
T/M (4)	G/T (15)	AD (26)	BC (37)
Try new dance steps	Talk about work (job related)	Kiss	Mediate
TND (5)	TAW (16)	K (27)	M (38)
Intro self to others	Tell jokes	Do job	Give in
ISO (6)	TJ (17)	DJ (28)	GI (39)
Surf web	Compliment others	Extra effort job	Procrastinate
SW (7)	CO (18)	EEJ (29)	P (40)
Explore environment	Ignore others	Find new way to do job	Pretend to work
EE (8)	IO (19)	FNJ (30)	PW (41)
Leave	Insult others	Improve skills	Stay with comfortable others
L (9)	InO (20)	IS (31)	SCO (42)
Be silent	Clean up	Confront other about slacking	
BS (10)	CU (21)	COS (32)	

The results were as shown in Table 6.6. The appropriateness of behavior choices (averaged over the 100 runs) was well above chance for both measures. The results showed the robustness of the model.

It had been argued that within-person variability over time was as high as between-person variability (e.g., Caprara & Cervone, 2000). Such variability was demonstrated above by the model. As explained earlier, in the model, the activations of drives and the selection of goals and actions were dependent on input stimuli from situations. As different situations were encountered, different drives were activated, and the

Table 6.4. The fifteen scenarios with their respective features.

Individual assignment: at work; in office; w/ no others; work to do; urgent
Working with one other: at work; in office; work to do; urgent; w/ one other
Working together on urgent project: at work; conference room; in office; conflict situation; work to do; urgent; w/ two or more others; w/ subordinates; w/ disliked acquaintance
At a group meeting: at work; in office; conference room; conflict situation; work to do; w/ two or more others; w/ friends; w/ boss; w/ disliked acquaintance
Review with boss: at work; w/ boss; in office; conflict situation; urgent
Taking a break with coworkers: in break room; at work; TV; work to do; w/ two or more others; w/ friends
Taking a break by yourself: in break room; at work; in office; work to do; w/ no others
Party at work: party; conference room; at work; alcohol; work to do; w/ two or more others; w/ friends; w/boss; w/subordinates; age difference > 7 years
Social engagement at boss's house: w/ boss; w/ strangers; w/ disliked acquaintance; w/ friends; w/ two or more others; conflict situation; w/ subordinates; w/romantic partner
Dance: dancing; w/ friends; w/ potential date; w/ strangers; w/ two or more others; alcohol
Trying to get a date: party; restaurant; alcohol; w/ one other; w/ potential date
On a date: restaurant; alcohol; w/ one other; w/ date
Family birthday party: home; party; w/ two or more others; w/ romantic partner; w/ relatives; w/ kids; age differences > 7
Wedding party at a fancy restaurant: party; wedding/formal party; restaurant; dancing; alcohol; w/ two or more others; w/ friends; w/romantic partner; w/ kids; age differences > 7
Party in a restaurant that has a bar: party; bar; restaurant; dancing; alcohol; w/ two others; w/ friends; w/ strangers; w/ potential date

Table 6.5. Behavior choice appropriateness for simulation test 1.

	Clarion
Top 1	89.8%
Top 3	78.2%

Table 6.6. Behavior choice appropriateness when noise was involved.

	Clarion
Top 1	86.1%
Top 3	78.1%

activated drives competed with each other for the control of behaviors through setting goals. Depending on what drives were simultaneously activated that competed with each other, a goal was stochastically determined. Furthermore, once the goal was set, different behaviors (actions) competed with each other to be chosen through stochastic selection, on the basis of situational inputs. The process of multiple stochastic competitions resulted in varying behaviors.

6.4.3.2. Simulation 2

Simulation Setup

Once the initial validity of Clarion as a personality model was established, the possibility of different personality types within the model was explored (Sun and Wilson 2014b). The person-situation interaction was also explored, because many past debates highlighted the importance of person-situation interaction (Caprara & Cervone, 2000). With the model, one could vary either personality or situation (or both) in testing such interaction. For example, one could keep a particular personality constant and examine how it responded differently to different situations. Likewise, one could keep a particular situation constant and see how different personalities responded differently to the same situation (Read et al., 2010).

For this simulation test, six different personalities were set up as shown in Table 6.1 (adapted from Read et al., 2010). These personalities were designed to form three complimentary pairs: sociable-shy, confident-anxious, and responsible-lazy.[5] Each of these pairs was intended to correspond to the far ends of one of the dimensions of the Big Five (Digman, 1990; John & Srivastava, 1999). One pair consists of the shy and the sociable, at the two ends of the extroversion dimension. Another pair consists of the anxious and the confident, at the two ends of the neuroticism dimension. The third pair consists of the lazy and the conscientious, at the two ends of the conscientiousness

5. These terms are not precisely descriptive. They are used here nevertheless, to follow the usage in Read et al. (2010).

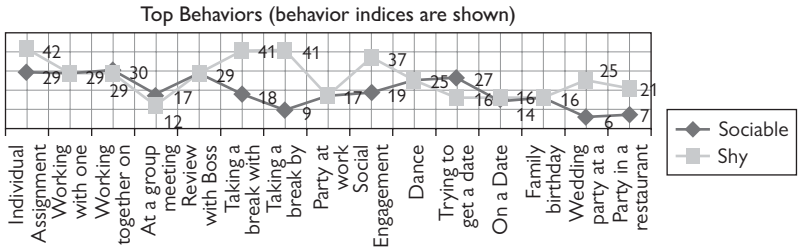

Figure 6.5. The most frequent behaviors of the sociable and the shy personality across the 15 scenarios. The *y*-axis shows the behavior indices (as specified in Table 6.3).

dimension. Their corresponding drive *deficit* levels, which determined these different personality types to a significant extent, were as shown in Table 6.1.

This simulation test used the same parameter settings as the previous one. The model for each of the six personality types was run on the set of 15 scenarios. Each scenario was tested for 100 steps and the chosen behaviors were recorded. The process was repeated for 100 different runs representing 100 different simulated "subjects."

Simulation Results

Figures 6.5–6.7 show the results of the simulation, using the three pairs of personality types where each pair consisted of two personality types at the opposite ends of one of the personality dimensions. These figures are separated by personality type, with the 15 scenarios on the *x*-axis and the index of the most frequently chosen behavior plotted on the *y*-axis (see Table 6.3 for the indices of behaviors). As shown by Figures 6.5–6.7, the

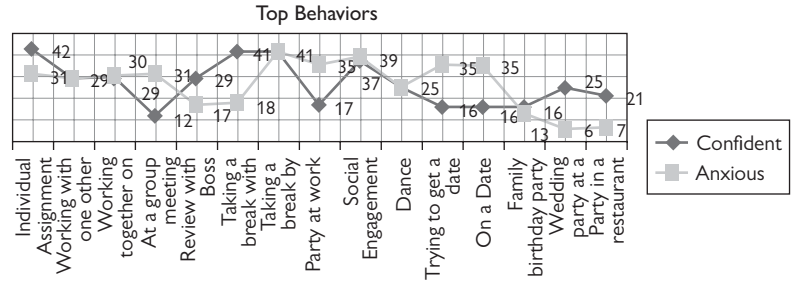

Figure 6.6. The most frequent behaviors of the confident and the anxious personality across the 15 scenarios. The *y*-axis shows the behavior indices (Table 6.3).

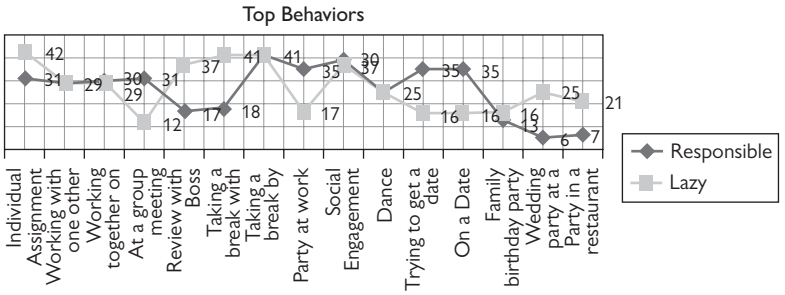

Figure 6.7. The most frequent behaviors of the responsible and the lazy personality across the 15 scenarios. The *y*-axis shows the behavior indices (Table 6.3).

two personality types in each pair behave differently across the set of 15 scenarios.

We drilled down to see how individuals of different personalities behaved in a given situation. As an example, Figure 6.8 showed the comparison between the sociable and the shy in the "urgent project" scenario. As shown, the sociable was more likely to talk about work and help others but was less likely to stay at the periphery; the sociable and the shy were almost equally likely to put in extra effort (because this was an urgent project scenario); and so on.

As shown by these figures above, the Clarion model demonstrated appropriate behaviors of the different personality types. An individual of a particular personality type acting appropriately within a given situation was the result of the interaction between the (relatively stable) characteristics of the motivational and other subsystems and the influence of the situations (which was more transient). For instance, in the model, the activations of different drives (within the MS) were the results of stable internal parameters (such as the *gain* and *deficit* values of different drives), as well as stimuli received from situations that were transiently present. Furthermore, which goal was activated at any given moment was partially a result of the competitive interaction among drives and which goal "won" that competition (within the MCS). Behaviors were then determined (stochastically) based on both the goal and the current situation (within the ACS).

In general, while structural models of personality focus on stable individual differences as captured by major personality dimensions that tend

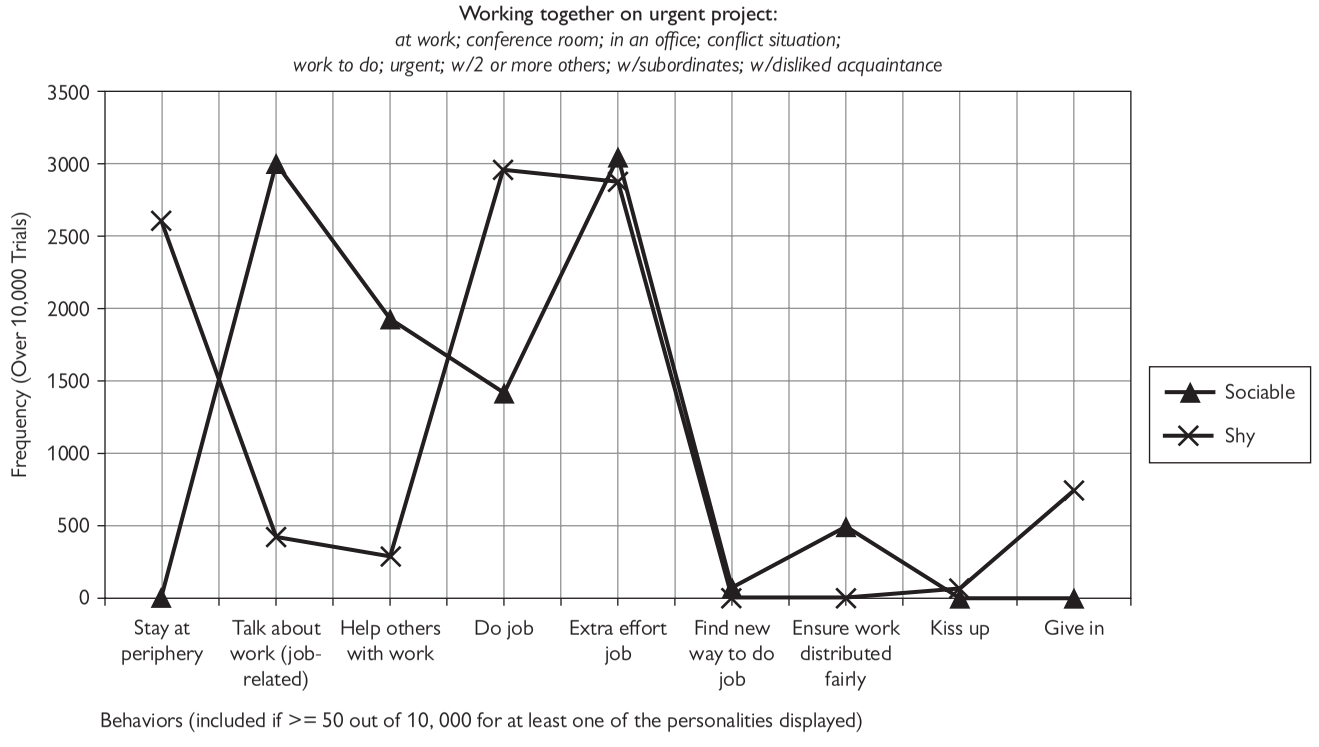

Figure 6.8. The sociable and the shy in the urgent project scenario.

to be relatively stable across situations and over time, mechanistic, process-based models involve more detailed constructs (i.e., mechanisms and processes), such as individual differences in drives, goals, knowledge and skills, and show, through the interaction of these constructs, how behavior persists or changes across situations and over time.

The Clarion model captures personality through underlying mechanisms and processes, so that as the external situation and the internal motivational state change over time, it generates behaviors that vary across situations and over time. At the same time, the structures and contents of the subsystems capture relative stable individual differences in behavioral inclinations and propensities, that is, personality traits. The model provides both an account of stable traits, as well as an account of behavioral variability across situations and over time—that is, the person-situation interaction.

Note that, as in other domains, individual differences, expressed as the different parameters in the Clarion model, might be attributed both to initial (in-born) biological differences (including genetic factors), as well as to different individual experiences, including different individual sociocultural experiences, that affect and adjust parameters of the cognitive architecture.

6.4.3.3. Simulation 3

It is important to validate the Clarion personality model with respect to actual human data. Below some empirical data are used for this purpose (Sun & Wilson, 2014b). First, data from Moskowitz et al. (1994) and Suh et al. (2004) are examined. Then, simulations are presented and compared to the prior (much simplified) simulations by Quek and Moskowitz (2007).

Human Data

Human data from Moskowitz et al. (1994). Moskowitz et al. (1994) investigated the influence of situational variables on interpersonal behavior. They showed that social role had an effect on an individual's behavior: Subjects behaved more submissively when interacting with bosses versus coworkers or subordinates. They were also more dominant with subordinates or coworkers than with bosses.

In their experiments, subjects were asked to monitor their social interactions for 20 days, using event contingent recording. Each subject completed a form for each significant social interaction (lasting at least five

minutes). Each subject was asked to indicate on the form the gender, working relationship, and personal relationship of each person involved. The form contained 46 behaviors that had been shown to be good indicators of dominance, submissiveness, agreeableness, and quarrelsomeness; subjects were asked to indicate which of the behaviors from the form they had engaged in.

Different numbers of dominant and submissive behaviors were observed in relation to social roles. In analyzing dominance, a statistically significant effect was found for social role: subjects reported more dominance toward subordinates or coworkers than toward bosses. In analyzing submissiveness, there was also a statistically significant effect for social role: subjects reported more submissiveness toward bosses than toward coworkers or subordinates.

Human Data from Suh et al. (2004). The experiment of Suh et al. (2004) was similar to that of Moskowitz et al. (1994). Subjects were asked to make event contingent recordings about non-work-related social interactions. Subjects were asked to provide information on a form about each such interaction: they were asked to indicate the gender and relationship of the person involved as well as to specify what behaviors took place.

Results concerning agreeable and quarrelsome behaviors were analyzed. For agreeable behaviors, a significant interaction between gender and relationship was found: agreeable behaviors with same-sex friends were significantly higher among women than men. Agreeable behaviors with romantic partners were significantly higher among men than women. For quarrelsome behaviors, a significant effect for relationship was found, as well as a significant interaction between gender and relationship. Quarrelsome behaviors with romantic partners were significantly higher among women than men.

Simulation of Human Data

Simulations of the human data described above were previously conducted by Quek & Moskowitz (2007). Their model was simple. A Backpropagation network was used. There were only three highly stereotyped scenarios. Also, instead of specific behaviors, Quek and Moskowitz's model simply chose between two outcomes (dominance versus submissiveness for one simulation, and agreeableness versus quarrelsomeness for the other).

The Clarion-based simulation added some necessary complexity and sophistication. The model for simulating the human data was set up

Table 6.7. Two classifications of behaviors.

Classification 1			Classification 2		
Dominance	Submissiveness	Neither	Agreeableness	Quarrelsomeness	Neither
Order others what to do (OO)	Be silent (BS)	Do job (DJ)	Tell jokes (TJ)	Insult others (InO)	Talk politics (TP)
Confront others about slacking (COS)	Stay at Periphery (SP)	Find New Ways to do job (FNJ)	Compliment others (CO)	Order others what to do (OO)	Talk about work (TAW)
Ensure work distributed fairly (EDF)	Kiss up (KU)		Kiss (K)	Confront others about slacking (COS)	
Insult others (InO)	Give in (GI)		Give in (GI)	Play practical joke (PPJ)	
Help others with work (HOW)	Leave (L)		Ask others about self (AS)	Tease/make fun of (T/M)	
			Gossip/talk about others (G/T)	Ignore others (IO)	

almost identically to the Clarion personality simulations described earlier. Behaviors (a subset of the previous set) were classified as belonging to some of the four categories: dominance versus submissiveness, and agreeableness versus quarrelsomeness, as shown in Table 6.7.[6] A few behaviors that did not fit into the classifications were also included (as in the real world).

Simulation of Moskowitz et al. (1994). The simulation of Moskowitz et al. was similar to the previously described personality simulations, except that it involved slightly modified scenarios with social roles added (necessary for this simulation). Whereas the prior simulation of the same data by Quek and Moskowitz (2007) only specified three highly stereotyped scenarios, the present simulation, in order to utilize the previous Clarion-based simulations, used two earlier scenarios (i.e., *urgent project* and *work with one other*). This leads to a total of six variations (i.e., *urgent project* with the "subject" as a boss, *work with one other* with the "subject" as a

6. The behaviors were categorized using the study by Moskowitz that constructed an item pool for assessing behaviors in which items were divided into experience-sampling scales for dominance, submissiveness, agreeableness, and quarrelsomeness.

Table 6.8. Scenarios for simulating Moskowitz et al. (1994).

Urgent project as boss: urgent; w/ subordinates; w/ two or more others; at work; work to do
Work with one other as boss: w/ subordinates; w/ one other; at work; work to do
Urgent project as coworker: urgent; w/ friends; w/ two or more others; at work; work to do
Work with one other as coworker: w/ friends; w/ one other; at work; work to do
Urgent project as subordinate: urgent; w/ boss; w/ two or more others; at work; work to do
Work with one other as subordinate: w/ boss; w/ one other; at work; work to do

boss, *urgent project* with the "subject" as a coworker, *work with one other* with the "subject" as a coworker, *urgent project* with the "subject" as a subordinate, and *work with one other* with the "subject" as a subordinate). These scenarios were coded using the same features as used previously but with the addition of roles (see Table 6.8).

The MS, the MCS, and the ACS of Clarion were set up in the same way as in the earlier simulations, using the generic personality and a subset of the behaviors from the earlier simulations (as specified in Table 6.7). No additional parameters were changed. Each scenario was tested for 100 steps, and the chosen behaviors were recorded. The process was repeated for 100 runs representing 100 simulated "subjects."

The findings from this simulation were consistent with the human data of Moskowitz et al. (1994). As mentioned before, Moskowitz et al. found that subjects behaved more submissively when interacting with bosses versus coworkers or subordinates, and they behaved more dominantly with subordinates or coworkers than with bosses. The same results were obtained from this simulation (Sun & Wilson, 2014b). The results were also consistent with those from the simplified simulation by Quek and Moskowitz (2006).

Simulation of Suh et al. (2004). The simulation setup of Suh et al. (2004) was the same as the previous simulation setup except that it involved non-work-related scenarios. Four scenarios were created for this simulation, as shown in Table 6.9. The same features were used for coding the scenarios as used in the previous simulations, except that gender was added. This alteration did not affect the overall design, nor did any other parameters have to be changed.

Table 6.9. Scenarios for simulating Suh et al. (2004).

Male with same-sex friend: *w/ same-sex friends; w/ one other; male*
Female with same-sex friend: *w/ same-sex friends; w/ one other; female*
Male with romantic partner: w/ one other; w/ romantic partner; male
Female with romantic partner: w/ one other; w/ romantic partner; female

Like the previous simulation, this simulation used a subset of the behaviors from the earlier simulations (Table 6.7). Each scenario was tested for 100 steps, using the generic personality, and the chosen behaviors were recorded. The process was repeated for 100 runs (representing 100 simulated "subjects").

The findings from this simulation were consistent with the human data from Suh et al. (2004). According to Suh et al., when interacting with same-sex friends, women exhibited significantly more agreeable behaviors than men. When interacting with romantic partners, men exhibited more agreeable behaviors and less quarrelsome behaviors than women. The Clarion simulation captured these findings (Sun & Wilson, 2014b). The simulation results were also consistent with the simplified simulation by Quek and Moskowitz (2007).

In a preliminary way, the two simulations of actual human data suggested some psychological validity of the Clarion personality model. Furthermore, they suggested some plausible explanations for the data patterns observed in empirical studies. For example, different behaviors when assuming different roles were attributed to roles as inputs from situations, rather than to personality changes.

Because of the involvement of detailed representations, mechanisms, and processes (including drives, goals, actions, and beyond), the simulations above provided deeper looks into the psychological underpinning of the human data than the previous simulations, and through the deeper looks, suggested some detailed plausible explanations.

6.4.4. Discussion

This detailed computational personality model derived from Clarion has been tested through a variety of simulations, including those reviewed

above and beyond. See, for example, Sun and Wilson (2011), Sun, Wilson, and Mathews (2011), Sun and Wilson (2014), and Sun and Wilson (2014b). These tests were useful. They led to a more comprehensive, mechanistic, process-based theory of personality. They also helped to clarify issues of personality in a mechanistic, process-based way. Practically speaking, the model can be applied in a number of practically relevant ways. In particular, a detailed model of personality is an indispensable part of simulating many social phenomena. For instance, it can be applied to cognitive social simulation (Sun, 2006), which will be addressed in Chapter 7.

One shortcoming of the simulations sketched above was the "tweaking" of parameters to get the desired outcomes. Because the parameters were set based on the consensus of the authors (the same as previous work on personality), some might regard them as being somewhat arbitrary. To remedy this problem, in Sun and Wilson (2014b), it was shown that personality models could be generated in a systematic way based on empirical data and could avoid parameters tweaking.

Note that previous computational models and simulations of personality have been compared to the Clarion model of personality. Some very brief comparisons were provided at the beginning of this section. For further comparisons, the reader is referred to Sun and Wilson (2011), Sun and Wilson (2014), and Sun and Wilson (2014b).

Overall, I should emphasize that personality, as well as emotion, moral judgment, and so on (as will be addressed in the next two sections), are all results of complex interactions among a large set of mental entities, mechanisms, and processes. Computational modeling and simulations enable one to see how exactly these entities, mechanisms, and process interact with each other in ways that are precise and detailed, which may not be possible with more traditional methods (as argued in Sun 2009b).

6.5. Accounting for Human Moral Judgment

In this section, I provide a computational account of moral judgment from the viewpoint of Clarion (Sun, 2013).

6.5.1. Background

Making moral judgment is an important capacity of the human mind (Thomson, 1985; Mikhail, 2007). It is also essential to the functioning of

human society in relation to order, cohesion, and cooperation. It is related to the philosophical and folk-psychological notion of "conscience," which has played a significant role in certain discourses (e.g., White, 2010).

A class of scenarios that have been used often in studying ethics is the trolley car problem (Foot, 1967; Thomson, 1985). In this class of scenarios, the key question in relation to understanding human moral judgment has been: how can different kinds of actions leading to more or less the same outcome differ in their moral acceptability?

For example, consider the following two scenarios:

> A runaway trolley is about to run over and kill five people, but a bystander can throw a switch that will turn the trolley onto a side track, where it will kill only one person. Is it permissible to throw the switch? (Foot, 1967).

> A runaway trolley is about to run over and kill five people, but a bystander is standing on a footbridge next to a large stranger. The bystander's body would be too light to stop the train, but he can push the large stranger off the footbridge onto the tracks, killing him but saving the five people. Is it permissible to push the large man? (Thomson, 1985).

In psychological experiments, subjects often judged that throwing the switch was permissible but objected to pushing the man off the footbridge. Thus, these two cases created a philosophical puzzle: what made it okay to sacrifice one person to save five others in the switch case but not in the footbridge case? There was also a related psychological puzzle: how did people come to the judgment that it was okay to switch the trolley but not okay to push the man off the footbridge?

If only the consequences mattered (five versus one), then both scenarios (throwing the switch and pushing the person, respectively) would be considered identical, but human subjects were evidently sensitive to additional factors. A few differing explanations were offered in the past (see, e.g., Greene et al., 2008, 2009; Mikhail, 2007; Sun, 2013).

Despite differences in opinions, there has been some agreement that (1) people do not simply maximize benefits relative to costs (such as five versus one), (2) aversion to killing people is an important factor, and (3) more proximal, intentional, or direct killing is more aversive.

Of particular interest is the notion of moral instinct. It has been argued that morality results not so much from conscious choice but more from instincts, habits, and predispositions (e.g., Monroe, 2012;

Sun, 2013). It is rooted in genetic factors, cultural norms, social roles, and so on. One may not be consciously aware of reasons because the fundamental part of morality is instinctual (Monroe, 2012). In contrast, moral reasoning may be more explicit and more culture-specific. For example, a meta-analysis of data from many countries (Henrich et al., 2010) found consistent evidence for postconventional moral reasoning in Western societies, but found no evidence in small-scale societies; even some highly educated non-Western populations did not show much evidence.

Such thinking led to dual-process (two-system) theories (Sun, 1994, 2002; Evans and Frankish, 2009)—responses to these two scenarios may reflect the working of two different "systems." According to Greene et al. (2009), on the one hand, there is a system that is more explicit, more "controlled", and relatively unemotional. It tends to think in a reasoned manner, for example, in utilitarian terms: better to save as many lives as possible. On the other hand, there is another, instinctual system that responds instinctively and emotionally to the action in the footbridge dilemma, but not so much to the action in the switch dilemma, which may explain why people tend to make utilitarian judgments in response to the switch dilemma but not in response to the footbridge dilemma (Greene et al., 2009). However, if the explicitly reasoned response is attractive and the instinctual, emotional response is also strong, a competition between them takes place (Greene et al., 2009; Sun, 2013).

Even though there have been a great deal of empirical data and a number of conceptual-level theories, one would like to know, in a more exact way, the mechanics of moral judgment: What psychological mechanisms and processes are involved in moral judgment? Are they exclusively used for moral judgment or are they shared among many psychological tasks? What different representations, mechanisms, and processes are involved? How different representations, mechanisms, and processes (including various modules or subsystems) interact in reaching moral judgment?

Clarion has the potential of providing needed mechanistic interpretations and process details to dual-process theories of moral dilemmas. Computational modeling and simulation in this domain can substantiate relatively vague conceptual-level theories and help to develop better theories.

6.5.2. Human Data

6.5.2.1. Effects of Personal Physical Force

Let us examine one experiment by Greene et al. (2009). In this experiment, subjects responded to one of the four versions of the footbridge dilemma, indicating the extent to which the proposed action was morally acceptable (on a nine-point scale).

In the standard footbridge dilemma, a person ("Joe") might save the five by pushing the large man off the footbridge using his hands. This action involved spatial proximity, physical contact, and personal physical force.

In the remote footbridge dilemma, Joe might drop the man onto the tracks using a trap door and a remote switch. This action involved none of the aforementioned factors.

The footbridge pole dilemma was identical to the standard footbridge dilemma except that Joe used a pole rather than his hands to push the large man. This dilemma involved spatial proximity and personal physical force without physical contact.

The footbridge switch dilemma was identical to the remote footbridge dilemma except that Joe and the switch were adjacent to the large man. This dilemma involved spatial proximity without physical contact or personal physical force.

Comparing the remote footbridge to the footbridge switch dilemma presumably isolated the effect of spatial proximity. Comparing the standard footbridge to the footbridge pole dilemma presumably isolated the effect of physical contact. Comparing the footbridge switch to the footbridge pole dilemma presumably isolated the effect of personal force.

Statistical tests of the human data showed that ratings of moral acceptability of sacrificing one life to save five differed significantly across the four versions. Pairwise comparisons revealed no significant effect of spatial proximity (remote footbridge versus footbridge switch), no significant effect of physical contact (standard footbridge versus footbridge pole), but a significant effect of personal force (footbridge switch versus footbridge pole). These results suggested that actions involving personal force were judged to be less morally acceptable. Spatial proximity and physical contact had no effect.

6.5.2.2. *Effects of Intention*

In another experiment by Green et al. (2009), effects of intention were examined. Each human subject in the experiment responded to one of four dilemmas (different from the previous four).

In the loop dilemma, Joe might save the five people by turning the trolley onto a looped side track that reconnected with the main track at a point before the five. There was a man on the side track who would be killed if the trolley was turned but would prevent the trolley from looping back and killing the five. The victim was killed intentionally, but without the use of personal force.

The loop weight dilemma was identical to the loop dilemma except that a heavy weight positioned behind the man on the side track stopped the trolley (not the man himself). The victim was killed as a side effect without intention and without personal force.

In the obstacle collide dilemma, a man was positioned on a narrow footbridge in between Joe and a switch that must be hit in order to turn the trolley and save the five. To reach the switch in time, Joe would have to run across the footbridge and would collide with the man, knocking him off the footbridge to his death. This involved personal force but not intention.

The obstacle push dilemma was identical to the obstacle collide dilemma except that Joe would have to push the man out of the way in order to get to the switch. This involved personal force and intention.

Statistical tests of the human data showed that there was a significant effect of intention (loop and obstacle push versus loop weight and obstacle collide) and no effect of personal force (loop dilemmas versus obstacle dilemmas). However, a significant interaction between intention and personal force was observed. The result suggested that the combination of the two factors, intention and personal force, might have the strongest negative effect on moral acceptability.

6.5.2.3. *Effects of Cognitive Load*

In the experiment of Greene et al. (2008), subjects were presented with moral dilemmas similar to the footbridge story in which one could kill one person to save several others. Subjects responded either under cognitive load (involving a concurrent digit-search task) or in a control condition.

Statistical analysis showed that there was no significant effect of load, but there was a marginally significant effect of type of judgment, with longer RT (response time) for utilitarian judgment than for nonutilitarian judgment. However, there was a significant interaction between load and judgment: RT for utilitarian judgment increased significantly as a result of load. Also, under load, utilitarian judgment was significantly slower than nonutilitarian judgment, but there was no such effect in the absence of load. This result suggested that utilitarian judgment might result from explicit processes, because explicit processes were more likely to be hampered by cognitive load (Sun, Slusarz, & Terry, 2005).

6.5.3. Two Contrasting Views

To justify modeling moral judgment with a cognitive architecture, one may look into possible alternatives, for example, simply using a Backpropagation neural network. Such a network could capture almost perfectly the various effects as summarized above. However, such a black-box-style simulation does not provide any deep account of the underlying psychological mechanisms and processes contributing to the outcome, and consequently it sheds little new light. It is therefore necessary to undertake more detailed modeling based on a more psychologically realistic model of the human mind.

With psychological realism in mind, there are two possible models that can capture the empirical data sketched above. Among them, one model is simple and straightforward. It corresponds to a reactive, situated view of the mind (e.g., Brooks, 1991; Sun, 2002). In contrast, the other model is more complex and more reflective of the motivational views of the mind (Murray, 1938; Maslow, 1943; Sun, 2009). Therefore, one pertinent question here is: should one use a "reactive" model or a more complex (e.g., motivation-based) model of the mind for this domain?

In general, Clarion can capture well this distinction and embrace both views. That is, it captures these different possibilities in different situations (chapters 3 and 4). In this domain, one can compare and contrast these two alternatives to hopefully shed new light on psychological processes of moral judgment.

To address this distinction, two Clarion models were developed. Model 1 was reactive: the decision making happened in the ACS, with the MS only expressing a generic desire to save life (without considering the complexity of the matter, e.g., having to kill one in order to save five).

Based on the generic goal from the MS, the ACS undertook the actual decision making considering the complexity of the matter (e.g., having to kill one in order to save five and the means necessary to do so). The implicit processes at the bottom level of the ACS made decisions based on its instinctual reactive routines (either biologically or socially formed, which favored not killing anyone, not intending to kill anyone, not using personal force, and so on; Greene et al., 2008, 2009), while the explicit processes at the top level of the ACS performed explicit decision making (e.g., based on explicit utilitarian calculation). This way of capturing moral judgment was justified (see, e.g., Brooks, 1991; see also Sun, 2002).

On the other hand, for model 2, which was motivational and more complex, the MS had to focus on a specific goal (not just a vague desire). The MS and the MCS together had to undergo a detailed process in order to come up with a specific goal, by taking into account the complexity of having to kill one in order to save five and the means to do so. After a specific goal was generated (e.g., to kill one in order to save more, or to do nothing), the ACS took that specific goal and other information into consideration when generating action outputs.

The explicit processes of the ACS, in this case, often made decisions in a deliberative way (e.g., by using the NACS to perform detailed utilitarian calculation). The implicit processes at the bottom level of the ACS might simply generate action recommendations in accordance with the goal dictated by the MS. This was because in this model, unlike in the previous one, the moral instincts necessary for moral judgment had been captured by the MS and the MCS in their selection of a specific goal.[7]

6.5.3.1. Details of Model 1

In this model, the MS and the MCS determined drive activations and goals, respectively. The MS included, among other things, the primary drive *nurturance* (i.e., helping others in need; see Chapter 4) and the generic goal *"save life whenever possible"* (as a result of drive activations). As noted earlier, the MS in this model only expressed a generic and vague goal to save life, without considering the complexity of the matter. On the basis of drive activations, the generic goal (*"save life whenever possible"*)

7. Note that according to Clarion, implicit processes are not necessarily emotional (Monroe, 2012). Conversely, emotion involves not just implicit processes (as discussed in the next section).

was chosen. Based on the goal, the ACS made actual action decisions taking into consideration of the complexity of the matter.

The ACS made action decisions based on the current goal and the current inputs denoting the situation; that is, the ACS decided on what to do (producing a rating of actions in this case) based on the situation and the goal, so the ACS had to deal with the dilemma of having to kill someone in order to save five people. The bottom level of the ACS made decisions implicitly (utilizing its instinctual reactive routines). The bottom level of the ACS used a Backpropagation network to generate action recommendations. The pre-training for the bottom level of the ACS was such that there was a bias favoring natural human instincts: for example, the tendencies (1) not to kill, either directly or indirectly, (2) to avoid personal force, (3) not to intend to kill, and (4) to save life whenever one can. After pre-training, the bottom level of the ACS so behaved. Given situational inputs, different instincts could be triggered, and they might contradict each other and thus compete. Different situations (with their different characteristics) might trigger these instincts to different extents and therefore generate different outcomes.

The top level of the ACS performed explicit action decision making based on knowledge embodied in explicit rules, for capturing explicit utilitarian calculation mostly but also explicit moral imperatives. The action recommendations of the two levels of the ACS were then integrated. If both levels had completed processing (within a possible time limit), a stochastic selection might be made between the two levels. Heavy cognitive load had the effect of slowing down explicit processes at the top level (Sun, Slusarz, & Terry, 2005), leading to slowdown of mostly utilitarian judgment.

6.5.3.2. *Details of Model 2*

This model was more complex and involved all the subsystems of Clarion. It followed a motivational view of human behavior, as briefly discussed earlier (cf. Maslow, 1943; Weiner, 1992; Sun, 2009; Wilson et al., 2009).

In this model, the MCS first decided on a specific goal based on activations of drives embodying moral instincts (which were triggered within the MS by situational inputs), taking into consideration the paradoxical

situation of having to kill one in order to save five. The MCS in this case came up with a specific goal for the MS (i.e., either to kill one in order to save five, or to do nothing), not just a vague desire to save life. Then, the ACS took the goal into consideration, possibly requesting inputs from the NACS, and made the final action decision taking into consideration all available information. The NACS, when invoked, might make recommendations in a more deliberative way (e.g., performing utilitarian calculation or other forms of reasoning) based on its declarative knowledge (from semantic and possibly episodic memory). So, in effect, the moral judgment was made twice in a row (though based on different knowledge).

Complex information flows occurred among these subsystems. The MS activated drives from situational inputs and then received the goal set by the MCS, which was based on drive activations within the MS. The NACS might be requested by the ACS to perform moral reasoning according to its declarative knowledge and thereby generate recommendations to the ACS. The ACS made the final action decision according to the current goal from the MS, the current sensory inputs, and the recommendations from the NACS, based on its own procedural knowledge. For similar complex motivation-cognition interaction, see, for example, Wilson, Sun, and Mathews (2009) and Sun and Wilson (2011).

In the bottom level of the MS, there might be the following "derived drives" (Chapter 4): *no killing*, *no battering* (i.e., no personal force), *no intending to kill*, and *saving life*. Given situational inputs, different drives might be activated and compete with each other. Different situations, due to their different characteristics, might activate each drive to different extents, and therefore each situation might lead to a different drive winning the competition and a different goal being chosen. In the top level of the MS, the selected goal was stored: "*act to save life*" or "*do not act to save life*". The goal was accessible to the ACS, and thereby regulated its operation.

The ACS then determined actions. Its top level used explicit rules for deciding on an action (e.g., initiating reasoning within the NACS, generating an output based on the NACS result, and so on), while its bottom level used a Backpropagation neural network to decide on an action implicitly. For utilizing the NACS (e.g., for utilitarian calculation), first, the ACS generated an action to initiate reasoning within the NACS; then, after getting results from the NACS, the ACS decided on an action. The bottom level of the ACS in this case tended to follow the goal from the MS, which in turn captured the moral

instincts. The top level, however, was more flexible: it might follow the recommendation from the NACS that might be contradictory to the goal from the MS. The final output was determined by stochastic selection of the two levels of the ACS (if both levels had completed processing within a time limit, if any).

6.5.4. Discussion

Computational modeling helps to disentangle the complexity of moral judgment. Computational modeling provides a detailed and precise picture of what is going on in the human mind, in a mechanistic and process-based way, at a level that no other methodology can replace.

The two models based on Clarion were able (1) to capture the effects of different factors found to affect moral judgment, for example, cognitive load, personal force, and intention (as reviewed earlier); and (2) to explore a number of dimensions of moral judgment, such as (2.1) to compare between explicit and implicit processing, (2.2) to compare between utilitarian calculation and moral instinct, and (2.3) to compare between a reactive and a motivational account (Sun, 2013).

In particular, one of the objectives was to compare and contrast two different ways of understanding moral judgment, through comparing the two models. In this regard, both captured the human data described earlier. However, the second model provided a more interesting look into the mind making moral judgment. It was more detailed, and its details were justifiable psychologically. For example, voluminous data point to the distinction between implicit and explicit processes (Reber, 1989; Seger, 1994; Sun, 2002), and similarly strong were the distinction between drives and goals and the dynamics within the motivational subsystem (Murray, 1938; Reiss, 2004; Sun, 2009).

In relation to the second model, it was hypothesized that moral judgment might have a lot to do with human motivation. Essential human motives might provide the basis for moral judgment. This motivational view of moral judgment was a hypothesis from Clarion, which was shown to be plausible, psychologically and computationally (Sun, 2013).

Moral judgment, however, is often more than just motivation, instinctual responses, and utilitarian calculation. It sometimes involves other, more complex forms of reasoning. A whole range of other mechanisms can be identified. These other mechanisms (as identified by, e.g.,

Bennis, Medin, & Bartels, 2010) include calculation-based decision making (decomposition of choice alternatives, and evaluation of outcome components and their integration), imitation-based decision making, advice-based decision making, case-based reasoning and decision making, identity-based reasoning and decision making (evaluating implications of decisions in relation to one's self concept), and so on. There are clearly a multitude of mechanisms in moral judgment. These different mechanisms may cooperate and compete with each other, as appropriate based on contexts, and their results are integrated. Therefore, what is needed for a better understanding of moral judgment is a generic and comprehensive cognitive architecture that is sufficient to account for many of these mechanisms and their interaction and competition (Sun, 2013).

In relation to various forms of decision making and reasoning, I should point out that in empirical work, it was found that people sometimes made judgment based on questionable criteria but then masked biased judgment by recruiting apparently justifiable reasons (e.g., Norton, Vandello, & Darley, 2004). Such findings can be captured and explained using Clarion (with its drive activation, goal setting, and metacognitive regulation). This could be important in understanding moral judgment.

It was also found in empirical work that the goal of getting at the truth and the goal of getting along with others led to different styles of cognitive processing (e.g., Chen et al., 1996): with the goal of getting at the truth, systematic processing often took place, while with the goal of getting along, shallower processing often happened leading to simply agreeing with others' opinions. This phenomenon can be captured and explained using Clarion with its motivational and metacognitive mechanisms. This could be another important aspect in understanding moral judgment.

The relation of moral judgment to the notion of conscience (e.g., White, 2010) is also of interest. Comparing the two notions, the relation might be that "conscience" represents the totality of moral judgments of an individual; or in other words, the essence behind one's moral judgments is one's "conscience." Thus, the notion of conscience should also be linked to, or be based upon, essential human motivation. The mechanisms and processes within Clarion serve as a plausible instantiation of the notion of "conscience."

Summarizing the discussions thus far, here are some of the principles concerning moral judgment according to Clarion (Sun, 2013), analogous to the principles of personality discussed in the previous section:

- There is no dedicated psychological module for moral judgment.
- Essential human motivations are the basis of moral judgment.
- There is a multitude of processes and mechanisms in moral judgment.
- Results of different processes are integrated, with variable contributions from each as determined by contexts and individual differences.

6.6. Accounting for Human Emotion

6.6.1. Issues of Emotion

The term "emotion" has come to denote a variety of phenomena. It is not entirely clear how one can identify some phenomenon as an emotion. Work in the fields of psychology and neuroscience has contributed to a better understanding of emotion, but many fundamental issues are yet to be understood. Computational models have also been developed, but they tend to be isolated models, not fully integrated into an overall cognitive architecture.

There are many open questions concerning emotion. Human emotion manifests itself as a complex of experiential, behavioral, psychological, and physiological characteristics with many underlying processes. How do these processes take place exactly? For example, one would want to know the following:

- What role does motivation play in emotion (in detailed, mechanistic terms)?
- What role does emotion play in behavior (in detailed, mechanistic terms)?
- What roles do explicit and implicit processes play in emotion respectively?
- How do implicit and explicit processes interact in emotion processing?

- How can emotion be controlled or regulated through metacognitive means?

and so on.[8]

There are reasons to believe that, psychologically, emotion is the collective outcome of operations throughout a cognitive system (similar in a way to personality). It should not be viewed as a unitary thing (although in engineering intelligent systems, a separate "emotion system" is often posited). That is, it is emergent. Its emergence may involve physiological reactions, action readiness, physical (external) actions, motivational processes, appraisal/evaluation processes, metacognitive processes, as well as decision making and reasoning of various forms. Emotion is the sum total of all of the above in particular circumstances (Sun & Mathews, 2012). Thus, in Clarion, emotion involves, for example, the ACS for actions, the NACS for evaluations, the MS for motivation, and the MCS for metacognitive regulation.

6.6.2. Emotion and Motivation

First, look into the motivational underpinning of emotion. One natural hypothesis within the Clarion framework is that emotion is deeply rooted in basic human motives (drives) and their possible fulfillment (Sun & Mathews, 2012). In this regard, I should mention that some other researchers, for example, Smillie et al. (2006), Carver and Scheier (1998), and Ortony et al. (1988), also stressed the importance of motivation and expectation in emotion. Within the Clarion framework, many kinds of emotion can be analyzed in terms of their motivational underpinnings.

For example, it has been hypothesized within Clarion (Sun & Mathews, 2012) that the emotion of *elation* is related to positive reward (including unexpected positive reward) and also, to a lesser extent, "expectation" of positive reward. Computationally, the intensity of elation may be (in part) a function of strengths of approach-oriented drives within the MS (Higgins, 1997).

On the other hand, the emotion of *anxiety* can be related to "expectation" of negative reward or punishment. The intensity of anxiety may be (in part) a function of strengths of some avoidance-oriented drives within the MS. Smillie et al. (2006) specifically identified the link between the avoidance

8. There are also questions concerning qualia of emotion: What constitutes an emotional experience? How is it different from other experiences in terms of controlling behavior? For a general treatment of these questions, see Sun (2002).

system and anxiety. Carver and Sheier (1998) also made related points. A related Clarion hypothesis was that the activations of avoidance-oriented drives often determined the proportion of implicit versus explicit processing within the ACS, as discussed earlier.

Furthermore, the emotion of *fear* may be due to "expectation" of more intense negative reward or punishment. Computationally, the intensity of fear can be determined in a similar way as anxiety—(in part) as a function of some avoidance-oriented drive strengths. Generally speaking, there has been a lack of clear distinction between anxiety and fear in clinical psychology and psychophysiology research (Smillie et al., 2006). Some demonstrated that what was constructed to represent "pure fear" was also a predictor of trait anxiety.

For another example, the emotion *anger* can be attributed to a mismatch between the "expectation" of a behavior from others and the actual behavior, when the actual behavior leads to more negative results compared with the expectation. Computationally, the difference leads to high drive strengths for some avoidance-oriented and approach-oriented drives (e.g., the *fairness* drive).

Such descriptions can be applied to other basic emotions as identified by various researchers (e.g., Ekman, 1999). Note that there may be some differences between some colloquial usages of emotion terms (and the folk psychology behind them), which are often loaded, and the usage here. Reinterpretation and clarification are necessary.

6.6.3. Emotion and the Implicit-Explicit Distinction

Psychological and neuroscience research suggested that emotional processes represented a more primitive information-processing and action decision-making mechanism (e.g., LeDoux, 1996). Experimental work indicated that emotional processing tended to quickly identify stimuli that were highly dangerous or beneficial (e.g., to an individual's survival). Emotions were often associated with hard-wired and specific responses (Ekman, 1999; Zajonc, 1980). Emotion might also induce processing states that biased toward specific types of behavior over a period of time.

Psychological research on implicit memory and implicit learning indicated the existence of distinct systems with distinct characteristics (as discussed in chapters 2 and 3), some of which were analogous to the characteristics of emotion-processing mechanisms. It was commonly believed

that the emotion-related system was often faster and less differentiated, while the other system was slower and more deliberative (Damasio, 1994; Zajonc, 1980). This distinction was similar to what was described as the distinction between explicit and implicit processes (Reber, 1989; Evans and Frankish, 2009).

However, the separation of the emotional and nonemotional systems is limited. For instance, researchers have described a variety of appraisal processes (which rely, to some extent, on explicit processes) that are involved in inducing an emotional state in reaction to a particular state of the world as perceived by an individual (Frijda, 1986; Smith and Lazarus, 1990). This situation is somewhat analogous to similarly complex interaction between explicit and implicit processes.

Such separation and interaction are clearly consistent with the Clarion framework. Therefore, the mechanisms and processes specified in Clarion can help to address questions regarding emotion. In Clarion, emotion involves various subsystems. Emotional processing mainly occurs in the bottom levels of these subsystems (Sun and Mathews, 2012). That is, emotional processing is mostly implicit (although not all implicit processes are emotional; LeDoux, 1996; Damasio, 1994; Zajonc, 1980). However, explicit processes (within the ACS and the NACS) have a role in emotion too, for example, through performing "cognitive appraisal" (Frijda, 1986; Smith and Lazarus, 1990), or through affecting implicit processes in other ways. However, they are not the main locus of the experience of emotion.

6.6.4. Effects of Emotion

It has been observed that various emotions involve or have effects on perception, action, and cognition.

First, emotion is closely tied to action. On the basis of motivation, emotion usually leads to action. In fact, emotion manifests, to a significant extent, through actions. Therefore, within Clarion, emotion leads to actions by the ACS, involving both implicit and explicit processes of the ACS, with implicit processes being more fundamental (as discussed earlier; Sun & Mathews, 2012). Frijda (1986), among others, indicated the importance of "action readiness" in emotional experience.

Emotions have various effects on perception. This phenomenon has been observed experimentally. For example, when in a state of anxiety, attention is heightened with regard to threatening stimuli. When in a

state of positive affect, stimuli are more elaborately processed (see Bower 1981; Mineka & Sutton 1992).

Research has also shown effects that emotion has on cognition (Simon, 1967). Emotion makes behavior more adaptive. Through incorporating emotion, one has both simple reflexive responses and complex cognitive processing at one's disposal, as well as their combination and interaction. Research has suggested that emotion involves and affects all functions studied in relation to cognition, namely, attention, learning, reasoning, memory, and so on. This means that, within Clarion, emotion involves and affects the ACS, the NACS, and the MCS, in addition to the MS as discussed earlier, including both implicit and explicit processes, with implicit processes being more fundamental.

The question is through exactly what mechanisms and processes emotion involves and affects these different functions. Relevant to addressing this question, a variety of computational models were proposed in the past, ranging from earlier ones such as Leven and Levine (1996) and Wright and Sloman (1997), to more recent ones such as Gratch and Marsella (2004, 2009). However, most of these computational models espoused rather explicit processing. As such, they dealt with only a limited kind of emotion, which was not necessarily the most fundamental kind. These models were also often standalone models (to a very significant extent at least). As such, they were not fully integrated into the overall cognitive architecture. Thus Clarion can provide a more comprehensive account through its mechanisms and processes resulting from modeling a large variety of cognitive-psychological functionalities. (Sun & Mathews, 2012; Wilson, 2012).

6.6.5. Emotion Generation and Regulation

Turn now to emotion generation and regulation. Emotion generation is accomplished through motivation, appraisal, and action (Wilson, 2012). Among these processes, motivation and action were addressed earlier, so I now look into appraisal.

In emotion generation, besides motivation and action, appraisal appears to be important. A principle tenet of appraisal theory was that emotion was a result of "cognitive appraisal" (Frijda, 1986). The model of Marsella and Gratch (2009), for instance, implemented a form of appraisal theory. It suggested that to adequately capture emotion, appraisal processes needed to rely on declarative knowledge and reasoning. Another model

by Reisenzein (2009), however, assumed that emotion arose when discrepancies were detected by continuously running, rapid, and automatic appraisal processes.

Within Clarion, two appraisal processes can be hypothesized. The automatic appraisal process (gut reactions) is usually fast, mainly involving implicit processes (within the ACS, the MS, and the MCS). The deliberative appraisal process is more explicit and usually slower, carried out mainly within the NACS (but involving other subsystems also). Thus, within Clarion, appraisal is carried out by a combination of the ACS, the NACS, the MCS, and the MS. Among these subsystems of Clarion, the NACS is mainly responsible for reasoning (implicit or explicit) needed for deliberative appraisal. The ACS, the MS, and the MCS, especially their implicit processes, are mainly responsible for automatic appraisal. However, the MCS (or the ACS) may trigger deliberative appraisal.[9]

With the generation of emotion, there is the need for action or coping. Coping of emotion, as identified by, for example, Lazarus & Folkman (1984), may be carried out in Clarion through the ACS and the NACS. Among them, coping by the ACS is obviously action oriented, but the actions may be either internally or externally oriented, while coping by the NACS may be centered on reasoning (implicit or explicit).

Now turn to emotion regulation of more complex forms (see, e.g., Gross, 2007). In general, control and regulation of emotion can be accomplished in a number of ways at different phases of processing: for example, (1) at the perceptual phase (e.g., by preventing the perception of threatening stimuli), (2) at the motivational phase (e.g., by changing priorities), (3) at the appraisal phase (e.g., by re-directing appraisal), or (4) at the action phase (e.g., by suppressing or enabling certain types of actions). Emotion regulation can be carried out through suppression, enabling, reappraisal, or other relevant means. It can be done either implicitly or explicitly (or in both ways; Gyurak, Gross, & Etkin, 2011).

Within Clarion, emotion regulation of these forms is accomplished mainly through the MS and the MCS. Emotion regulation often amounts to regulation by the MCS in the form of input filtering, goal setting, action output filtering, and so on (see Chapter 4 for details), in response to motivational states and sensory inputs, corresponding to these phases identified above.

9. Note that the outline above may be related to the model of moral judgment (discussed in the preceding section), in the sense that the distinction between the two types of appraisal maps roughly onto the second model of moral judgment.

Emotion regulation may thus affect action and reasoning within the ACS and the NACS. At a deeper level, drive activations within the MS may also be adjusted (see Chapter 4) as a form of emotion regulation, through the MCS, and thus action and reasoning (within the ACS and the NACS) change as a result. In this regard, a clear distinction between emotion generation (e.g., through motivation, appraisal, and action) and emotion regulation (e.g., of inputs, of action outputs, and of motivation) is unnecessary (cf. Kennedy & Bugajska, 2010).

For a comprehensive account of these facets of emotion, see, for example, Wilson (2012), which includes a detailed discussion of how exactly emotion processing takes place in Clarion. A number of related simulations can be found in Wilson et al. (2009), Wilson and Sun (2014), and so on.

6.6.6. Discussion

To summarize the discussion of this section, in Clarion, emotion involves various subsystems: the ACS (for action), the NACS (for appraisal), the MS (for motivational processes), and the MCS (for metacognitive regulation). Complex interactions occur among these different subsystems and among many components within. However, it is a complex dynamic system with clearly structured components (each with specific knowledge, mechanisms, and processes) interacting with each other.

Although still preliminary, Clarion has thus far shown potential for answering many questions regarding emotion. Addressing emotion within a comprehensive cognitive architecture enables its modeling to make contact with detailed, established psychological mechanisms and processes. As a result, the study of emotion is linked to other psychological functionalities such as memory, decision making, reasoning, metacognitive regulation, and so on, defined within a cognitive architecture (Wilson, 2012).

However, it would be a stretch to claim that Clarion can by now capture every aspect of something as complex as emotion. There is a lot more that needs to be done in this regard.

6.7. General Discussion

The work described in this chapter provides a glimpse into how motivation underlies cognition and how metacognition regulates cognition.

Clarion provides computational accounts and explanations of various motivational and metacognitive phenomena that have so far not been tackled within cognitive architectures. These include the phenomenon of metacognitive monitoring, the phenomenon of metacognitive intervention, the phenomenon of performance degradation under pressure, the phenomenon of human personality across a variety of circumstances, and the phenomenon of situational factors in moral judgment.

While the suggestion that motivation and metacognition affect cognition is not novel, Clarion integrates motivation and metacognition into a unified cognitive architecture. This work has also taken a step toward explaining exactly how and in what way cognitive performance is affected by motivational, metacognitive, and other factors. Clarion, in this regard, addresses the interaction among motivation, metacognition, and cognition, beyond what other computational cognitive architectures have done. Clarion does so in a detailed, process-based, and mechanistic way. Therefore, it provides detailed, mechanistic, process-based explanations (Sun, 2009b). In this way, it leads to some new theories and new explanations, but also embodies and substantiates some previous theories and explanations.

The Clarion cognitive architecture contains rather detailed motivational and metacognitive mechanisms. That is, many motivational and metacognitive processes are architecturally specified in Clarion. This approach makes simulations of motivational and metacognitive phenomena easier to construct, less ad hoc, and more uniform. This approach has been shown to be viable.

Beyond those discussed above, there are many other tasks and data sets that may be, or have been already, accounted for by Clarion, involving the MS and/or the MCS. In particular, a number of tasks were tackled in a fashion similar to those described in detail earlier. For instance, Wilson et al. (2010) provided a computational explanation of stereotyping under pressure, using metacognitive regulation resulting from drive activations, based on the experimental work of Lambert et al. (2003). For another instance, Brooks et al. (2012) explored the effects of assigned performance targets on performance, and performance differences were explained by differences in explicit and implicit processing as a result of metacognitive regulation based on external target assignment and consequent drive activations. For yet another instance, Chen et al. (1996) showed that with the goal of getting at the truth,

systematic processing often took place, while with the goal of getting along, shallower processing often occurred. This was explained using Clarion whereby different motivations led to different metacognitive regulation of cognitive processing. Similarly, Norton, Vandello, and Darley (2004) showed that people sometimes made judgments based on questionable criteria, but then masked biased decision making by recruiting apparently justifiable reasons. Such findings were also explained by Clarion, whereby different drives and goals were emphasized during judgment and during justification and thus resulted in different processing.

Various other examples have also been addressed and some of their descriptions can be found in prior publications (e.g., Sun, 2009). With the work ongoing along this direction, Clarion may eventually provide a comprehensive, yet detailed picture of motivation, emotion, self-monitoring, self-regulation, and other forms of motivation-cognition-metacognition interactions (Carver & Scheier, 1998; Weiner, 1992; Caprara & Cervone, 2000).

7

Cognitive Social Simulation

7.1. Introduction and Background

It has been pointed out that one (unfortunate) reality of the social and behavioral sciences is the relative lack of integration and communication between the cognitive and the social disciplines (Sun, 2006, 2012). Each discipline tends to consider a particular aspect and to ignore (more or less) the rest. Consequently, they often talk past each other instead of to each other.

However, in both the social sciences and cognitive science, the notion of agent has played a significant role in research, especially in recent decades. In particular, agent-based social simulation is becoming an increasingly important research methodology in the social sciences. It has been used to test theoretical models or to investigate their properties. A simulation may even serve as a theory or an explanation of a social phenomenon by itself. Issues addressed thus far by agent-based social simulation have been diverse. They include, for example, social norms, language evolution, social cooperation, culture formation, group interaction, opinion dynamics, stock market dynamics, tribal institutions, traffic patterns, collective decision making, organization design, and many others.

At the same time, computational models of agents have also been developed in cognitive science, often in the form of computational

cognitive architectures. However, despite that development, most of the work in agent-based social simulation still assumes rudimentary agents. Agent models in social simulation have often been custom-tailored to the task at hand. Often, they are not even remotely comparable to cognitive architectures in terms of complexity and sophistication. Although the approach may be adequate for achieving limited objectives of some social simulations, it is overall unsatisfactory intellectually and practically. For instance, it limits the realism and hence the applicability of social simulation. More importantly, it also limits the possibility of tackling the theoretical issue of the micro-macro link (Alexander et al., 1987; Sun, 2012b).

Detailed computational cognitive models, especially cognitive architectures, may provide a foundation for understanding social processes, issues, and phenomena. Incorporating realistic psychological constraints, capabilities, and tendencies of individuals in their interaction with their environments (both physical and social), these models take cognition-psychology of individuals into serious consideration. When trying to understand social processes, issues, and phenomena, it is desirable to do so, given that detailed computational models of individuals that incorporate a wide range of psychological functionalities have been developed.

There are possibly significant advantages in using cognitively-psychologically realistic models in agent-based social simulation. For instance, if a model is reflective of human cognitive-psychological processes, the explanations and predictions that it provides may be more detailed and more nuanced. The explanations and predictions that refer to human cognition-psychology may be more illuminating than those that refer to ad hoc parameters of a simplified model or to external measures only. For another instance, through psychologically realistic models of individuals, one may investigate the interactions among cognition, motivation, social institutions, physical environments, and so on. Some significant relationships among cognitive, motivational, social, and environmental factors may thus be revealed. Among many other relationships, there may be cognitive-environmental dependency, cognitive-motivational dependency, and motivational-environmental dependency (Sun & Naveh, 2007; Sun & Fleischer, 2012; more details to follow).

Max Weber pointed out that unlike the physical sciences, the social sciences need to gain an "empathetic understanding" of the "inner states" of social actors and thus gain an understanding at both the level of causation and the level of "meaning" (i.e., cognition and motivation; Weber,

1991). Alfred Schutz attempted to understand the construction of social reality from the point of view of an individual, in terms of meaningful actions, motivations, and a variety of social relationships (e.g., Schutz, 1967). Giddens's (1984) discussions regarding the relation between structure and agency are also relevant in this regard.

Similar points have also been made in various technical domains, for example, in the context of cognitive realism of game theory, or in the context of deeper models for human-computer interaction. In Axelrod's simulation work (1984), it was shown that even adding a cognitive factor as simple as memory of past several events into a model could completely alter the dynamics of social interaction.

The discussion above (as well as the work detailed in the remainder of this chapter) points to a more psychologically realistic approach towards social simulation, namely, *Cognitive Social Simulation* (Sun, 2006), as well as an area of research—exploring psychological-social-environmental interaction through cognitive social simulation.

Below, I describe a number of cognitive social simulations in different domains addressing different issues.

7.2. Cognition and Survival

In Sun and Naveh (2007), a tribal society was simulated in which the interaction between individual cognition and social (and environmental) factors was explored. With Clarion-based agent models, the results of the cognitive social simulation shed light on the role of cognition (in the narrow sense) in social processes. Below we look into some details of this simulation.

7.2.1. Tribal Society Survival Task

To understand the rationale behind this simulation, we may first look into a prior simulation by Cecconi and Parisi (1998). In their simulation, Cecconi and Parisi created simulated social groups (tribes). In these groups, to survive and reproduce, an individual must possess certain resources. A group in which each individual used only its own resources was said to adopt an individual survival strategy. However, in some other groups, resources might be transferred. Such a group was said to adopt a social survival strategy. For instance, in their simulated

world, the "central store" (CS) was a mechanism to which all individuals in a group transferred part of their resources. The resources collected by the CS could be redistributed in some way to the members of the group.

Cecconi and Parisi (1998) conducted simulations comparing groups adopting different strategies. They used neural networks to model individuals and a genetic algorithm to model evolution. Neural networks (representing individuals) survived and reproduced differentially based on the quantity of food that they were able to find and consume. Cecconi and Parisi explored what conditions determined group survival or extinction. This work was interesting, because it provided a fertile ground for exploring a range of issues, from individual behavior to social institution, from individual learning to evolution, and many others.

However, in this early work, there was very little in the way of human cognition-psychology. Investigation, modeling, and simulation of social phenomena need cognitive science, because such endeavors need a better understanding, and better models, of individual cognition-psychology, on the basis of which better models of aggregate processes can be developed. Cognitive modeling may provide better grounding for understanding social phenomena, by incorporating realistic constraints, capabilities, and tendencies of individuals in terms of their cognitive-psychological processes. Arguments along this line can be found in, for example, Sun (2001, 2006, 2012b).

Therefore, to redress the neglect of human cognition-psychology in agent-based social simulation, more detailed and more realistic models of cognitive-psychological mechanisms and processes need to be incorporated. Clarion has been successful in simulating a variety of psychological tasks. Therefore, it may be extended to the capturing of social phenomena.

In the work described below, the simple simulation of Cecconi and Parisi (1998) was revamped (Sun & Naveh, 2007). The general setup, however, remains essentially the same. The world was made up of a two-dimensional grid. Food items and agents were randomly distributed among the locations of the grid. The food crops grew by seasons, so food was replenished periodically. There were the harsh, medium, and benign conditions that were distinguished by the availability of food. Agents were of a limited life span, which varied from individual to individual depending on the energy consumption and the maximum lifespan. Agents looked for and consumed food in an effort to prolong their life spans.

There was a "central store" in some cases as in Cecconi and Parisi (1998). Agents might be required to contribute to the central store in these cases. However, different from Cecconi and Parisi (1998), some further social institutions were introduced. For instance, in case of mandatory contribution to the central store, a penalty was introduced for not contributing to the store.

Most notably, in this work, agents were more cognitively realistic than those of Cecconi and Parisi (1998). Therefore, these new simulations shed more light on the role of cognition in determining survival strategies and its interaction with social institutions. To investigate the interaction between social processes and individual cognition (i.e., the micro-macro link) and other issues, detailed statistical analysis was applied to different settings so that a more precise understanding could be achieved.

7.2.2. Simulation Setup

Specifically, in this simulation, agents were constructed based on the ACS. Each agent faced a certain direction (north, south, east, or west). Each agent received inputs regarding the location of the nearest food, relative to the current position of the agent and its current direction. Its perception was divided into four pie-slice-shaped quadrants. Each agent could generate an action output: either (1) turn 90 degrees right, (2) turn 90 degrees left, (3) move forward, (4) pick up food and contribute a portion, (5) pick up food and keep all of it. Action decision making was accomplished using both the neural network at the bottom level (trained with Q learning), and the rules at the top level (learned using RER).

Each agent lived for a maximum of 350 cycles, but it might die early due to lack of food. There were initially 30 agents to begin with, and the number of agents fluctuated due to birth and death, within the bound of a maximum of 30 agents.

The same as in Cecconi and Parisi's simulation, procreation was asexual (i.e., only one parent was required). Procreation occurred if: (1) an agent had reached 120 energy units or more, and (2) there were fewer than the maximum number of agents in the world. The newborn was placed in a random location. The parent handed out 60 energy units to the child upon birth. The child inherited its parent's internal makeup

(including the neural network and the rule set), but there was a 10% chance of minor mutation when a child was spawned.[1]

The world was made up of a 100 × 100 grid. Each of the locations at most contained one food item (50 energy units). At the beginning and every 40 cycles, the 100 × 100 grid was replenished: randomly selected locations were restocked with food items until the grid had 600 food items in all. Also tested was a more benign condition, in which 900 locations contained one food item each, as well as a harsher condition in which 300 locations contained one food item each.

Some of the simulations involved the institution of a "central store." In these cases, an agent was required to contribute 20 energy units to the central store when it picked up a food item (50 energy units). When a central store was used, at each cycle, 10% of the agent population (randomly selected) received 5 energy units each from the central store. A variation of this was that only agents with 10 or less energy units received distributions from the central store.

Each agent began with 60 units of energy and consumed one unit of energy per cycle to stay alive. For each agent, capturing and consuming a food item increased its energy by 50 units.

Each agent, when picking up a piece of food, decided whether to contribute to the central store or not. Enforcement mechanisms were introduced in this simulation, different from Cecconi and Parisi (1998). When an agent picked up a food item, if it decided to contribute, it contributed 20 energy units to the central store. If the agent decided not to contribute, there was a 30% chance of being caught if an enforcement mechanism was in place. If caught, the agent was fined 40 energy units (which were transferred to the central store). Otherwise, the agent kept all the energy units acquired.

There were four variations in this regard: no central store, central store but no individual choice (equivalent to extremely strict enforcement), central store with individual choice and enforcement, and central store with individual choice and no enforcement.

The reinforcement that an agent received was as follows: if the agent captured one food item and contributed to the CS, the reinforcement was 0.6. If the agent captured one food item and contributed nothing without being caught, the reinforcement was 1.0. If the agent captured one food item,

1. If mutation occurred, each of the weights in the neural network at the bottom level had a 20% chance of being randomly decreased or increased by 0.1.

Table 7.1. A list of cognitive, social, and environmental parameters and their values used in the simulation.

Probbl (probability of using the bottom level):	
#1	0.25
#2	0.75
Learn (learning rate):	
#1	0.25
#2	0.75
Gen (generalization threshold):	
#1	1.0
#2	3.0
Food (food availability):	
#1	300
#2	600
#3	900
Strat (survival strategy; cs = central store):	
#1	cs/enforcement/choice
#2	cs/no enforcement/choice
#3	cs/no enforcement/no choice
#4	no cs

cheated, and was caught, the reinforcement was 0.2. This reinforcement was proportional to what an agent had to keep in each scenario (based on 30/50, 50/50, or 10/50 energy units being kept, respectively).[2]

Because of the computational cost, this simulation had to focus on a small set of cognitive, social, and environmental parameters, and a small set of values for each of these parameters (usually two to three). See Table 7.1 for details of these parameters. Dependent variables (performance measures) were also limited to the following: (1) average individual energy acquisition per agent per cycle, (2) average population size (average number of agents in a population), as well as (3) average lifespan.

All combinations of these parameters were tested by a factorial design. There were a total of 96 combinations. Each simulation ran for a

2. More generally, the penalty for violating the norm can be from multiple sources: (1) social sources, such as an enforcement mechanism that extracts penalty from violators; and (2) internal sources, for example, from an internal feeling of guilt.

maximum of 2,000 cycles. Sampling was done 10 times per simulation. Thus 960 observations were gathered in total.

7.2.3. Simulation Results

Let us consider performance measures used in the simulation. A reasonable measure of average individual success seems to be the energy acquisition per capita per cycle. However, sometimes, as population sizes decrease, the performance of survivors actually increases due to less competition for food. For this reason, population size should be simultaneously examined as a complementary measure; the two measures should be cross-referenced. Another, more individual measure, average life span, measures how much (on average) each individual benefits from food (or suffers from the lack of food), different from the more global measures. With these measures in mind, let us look into some results from the simulations.

7.2.3.1. Effects of Social and Environmental Factors

One important finding was that in terms of average energy acquisition per capita per cycle, strategy mattered significantly (in a statistical sense). See Figure 7.1 for the result.

As indicated in the figure, the worst strategy for this performance measure was that of a central store with no free choice. This was probably because individuals in this case had no leeway whatsoever in contributing to and drawing from the central store, which led to less evolutionary pressure on individuals. Performance suffered in the long run as a result. Also as expected, the strategy of a central store with free choice and some enforcement was significantly better. The central store with free choice but no enforcement was in turn better than the central store with free choice and enforcement (the difference was small, however). The most interesting finding here was that the best strategy was that of no central store. This was probably because more evolutionary pressure led to better individual performance and the no-central-store strategy exerted most evolutionary pressure on individuals. As a result, individuals surviving in this environment fared better in general. The strategy of a central store with free choice but no enforcement was the closest to it. As has been shown before by Cecconi and Parisi (1998), the strategy of a central store with free choice but no enforcement often turned into the strategy of no central store at all.

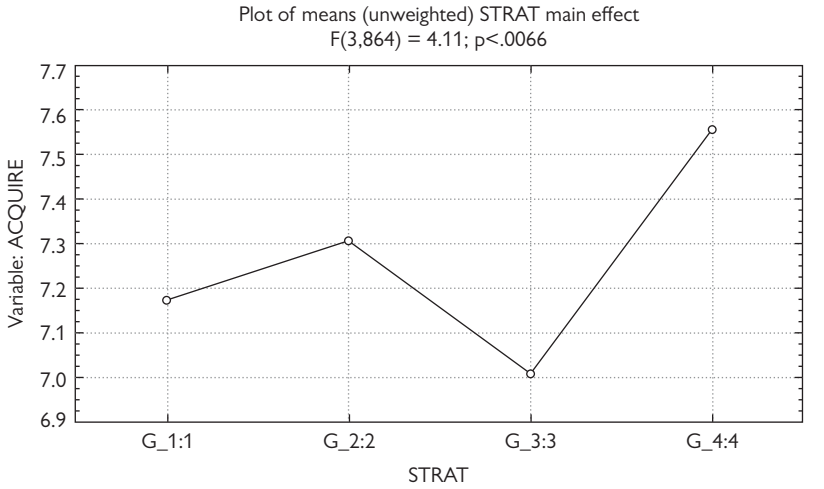

Figure 7.1. The significant differences in terms of energy acquisition with regard to strategies. The *y*-axis represents energy acquisition rate. The *x*-axis represents strategy. *Strat:* 1 = cs/enforcement/choice, 2 = cs/no enforcement/choice, 3 = cs/no enforcement/no choice, 4 = no cs.

In addition, in a separate simulation involving the use of the central store, it was found that distribution from the central store only to the needy was slightly better than random distribution. But their differences were not statistically significant.

In terms of population size, the two factors, strategy and environment (i.e., food availability), together had a significant effect (consistent with the findings of Cecconi & Parisi, 1998). Statistical analysis showed an interaction between strategy and food availability, and that strategy had an effect on population size only under some environmental conditions (when food was less than abundantly available).

In terms of average lifespan of individuals, there was a statistically significant effect of strategy. The strategy of the central store with no choice performed well, which showed that the mandatory social welfare system did help the survival of individuals. The strategy of the central store with free choice and enforcement was comparable. These two strategies were the best among the four strategies. However, the strategy of no central store was the worst, for the lack of a social cushion against adverse circumstances that an individual might encounter. See Figure 7.2.

This ordering was the opposite of that resulting from the measure of energy acquisition rate. This was because lifespan was arguably a measure that was more sensitive to individual performance, as it captured

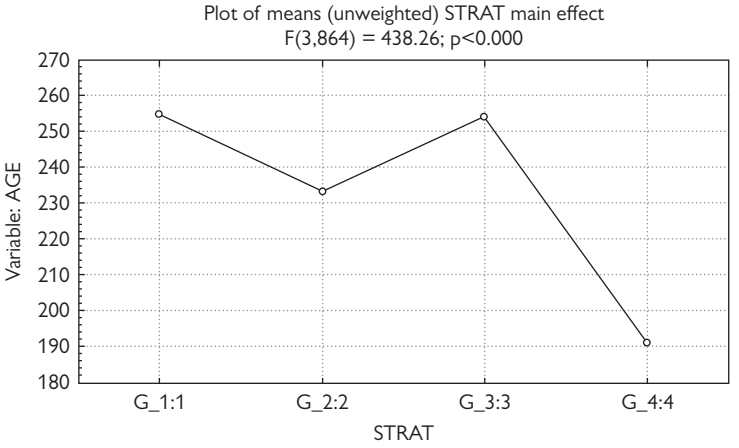

Figure 7.2. The significant differences of lifespan with regard to strategies. The *y*-axis represents lifespan. The *x*-axis represents strategy. *Strat*: 1 = cs/enforcement/choice, 2 = cs/no enforcement/choice, 3 = cs/no enforcement/no choice, 4 = no cs.

how an individual benefited from the availability of food or suffered from the lack of it (on average). Therefore, some contrasting effects might be observed: whereas some individuals might die of starvation, the average rate of energy acquisition might be high nevertheless. Conversely, the starvation of some individuals might be prevented, but the welfare of the population as a whole might suffer. That is, some strategies might lead to benefiting the overall population while at the same time hurting some individuals within that population, and vice versa.

Separately, distribution from the central store to the needy only, as opposed to random distribution, improved average lifespan significantly, in contrast to the result from using energy acquisition as the performance measure. So there was another contrasting effect.

Overall, the results indicated that strategies did matter for individuals surviving in a particular physical and social environment (with energy acquisition rate or lifespan as dependent measures). There were some noteworthy contrasting effects using energy acquisition rate and lifespan as dependent measures.

7.2.3.2. Effects of Cognitive Factors

There is more to cognitive social simulation than generating performance measures. Because Clarion captures a variety of cognitive factors, one can

vary parameters that correspond to specific cognitive factors and observe their effects on performance.

This approach has an important advantage. With Clarion, the parameters being altered are presumably important aspects of cognition, and thus observed differences in performance are likely to stem from real differences in individual cognition. Thus, in the following simulations, a number of cognitive parameters were varied within the ACS, and their effects on performance were observed. In particular, the interaction between cognition and social institution and that between cognition and physical environment were of interest: that is, what cognitive parameter settings were suitable for what kind of social institution and physical environment (a central store or not, enforcement or not, under the condition of abundant food versus scarce food, and so on).

Let us first look into an analysis using average energy acquisition per capita per cycle as the dependent variable. There were some expected, not-so-surprising effects of cognitive parameters, including probability of using the bottom level, learning rate, and so on. For instance, a significant effect of probability of using the bottom level was found. Using explicit processes (explicit rules) more at the top level was beneficial (up to a certain extent, of course). This was because, as demonstrated before (Sun, Slusarz, & Terry, 2005), explicit processing at the top level helped implicit learning at the bottom level and thus the overall performance. Hence there was the performance difference as shown in Figure 7.3.

There was also a significant effect of learning rate. Higher learning rates were beneficial (up to a certain extent). This was because higher learning rates (up to a certain extent) helped individuals to quickly adapt to situations and exploit their physical and social environments to ensure their survival. See Figure 7.4.

Some interesting two-way interactions were also found. There was an interaction of learning rate and food availability. Under low food availability, a higher learning rate was better; under medium or high food availability, it did not matter. This finding might be explained this way: under low food availability (i.e., under a harsh environmental condition), it was more important to exploit the environment and the social institution in order to survive, while in less harsh conditions, slacking off was less of a problem. See Figure 7.5.

Interestingly, there was the interaction between probability of using the bottom level and strategy. As indicated by Figure 7.6, the strategy

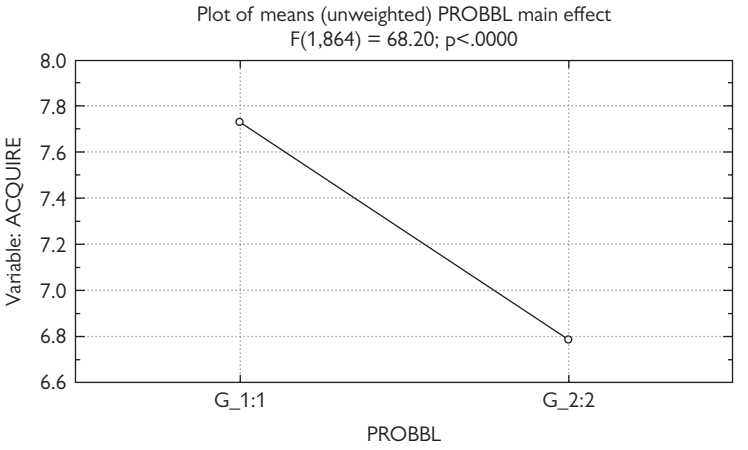

Figure 7.3. The significant effect of probability of using the bottom level. The *y*-axis represents energy acquisition rate. The *x*-axis represents probability of using the bottom level. *Probbl:* 1 = 0.25, 2 = 0.75.

of no central store was the best strategy when explicit processes were not used much (i.e., when the probability of using the bottom level was high), while it was merely average when explicit processes were heavily used (i.e., when the probability of using the bottom level was low). One explanation was that when cognition was highly explicit (when explicit rules were heavily used), individuals were more likely to learn better and

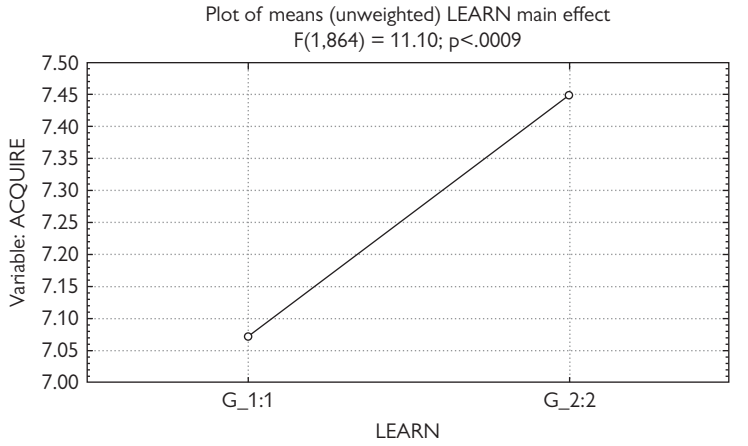

Figure 7.4. The significant effect of learning rate. The *y*-axis represents energy acquisition rate. The *x*-axis represents learning rate. *Learn*: 1 = 0.25, 2 = 0.75.

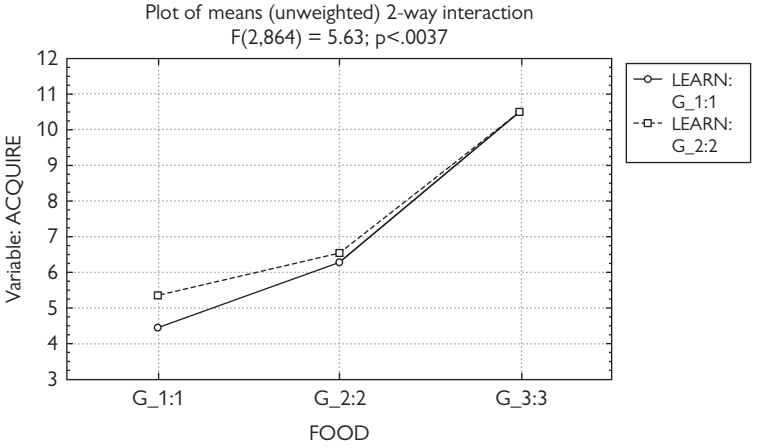

Figure 7.5. The significant interaction between learning rate and food availability. The *y*-axis represents energy acquisition rate. The *x*-axis represents food availability. The different lines represent different learning rates. *Learn*: 1 = 0.25, 2 = 0.75. *Food*: 1 = 300, 2 = 600, 3 = 900.

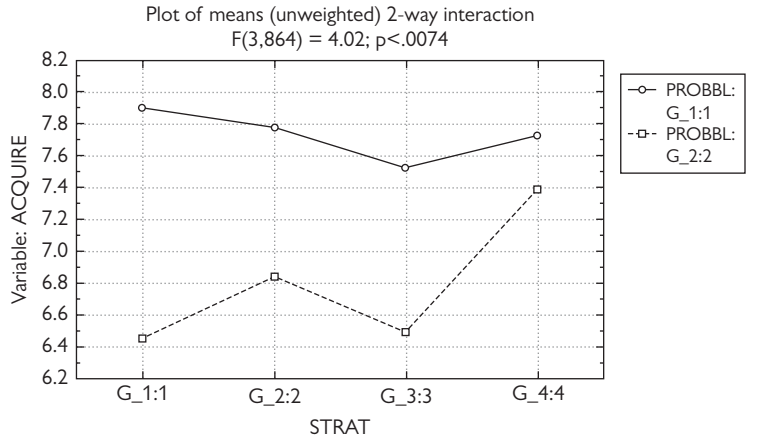

Figure 7.6. The significant interaction between probability of using the bottom level and strategy. The *y*-axis represents energy acquisition rate. The *x*-axis represents strategy. The different lines represent different probabilities of using the bottom level. *Strat:* 1 = cs/enforcement/choice, 2 = cs/no enforcement/choice, 3 = cs/no enforcement/no choice, 4 = no cs. *Probbl*: 1 = 0.25, 2 = 0.75.

explore given situations better (Sun, Slusarz, & Terry, 2005). Situations created by the strategy with a central store, choice, and enforcement were more complex than others and therefore required better learning abilities. However, given a better learning ability, individuals might perform better in these more complex situations by exploiting them effectively. Therefore, individuals with more explicit processing performed better in these complex situations. In contrast, when explicit processes were used less, individuals did not learn as well. Therefore, in this case, the no-central-store strategy—the simplest strategy—turned out to be the best. In general, the differences among strategies were greater when explicit processes (explicit rules) were used less, because individuals with poorer learning abilities did not learn to handle more complex situations as well.

There were also some interactions between cognitive parameters. For instance, there was the interaction of probability of using the bottom level and generalization threshold. I will not get into these here (see Sun and Naveh, 2007 for details).

Now I turn to the analysis using population size as the dependent variable (which was a more global measure of performance). Generally speaking, similar statistically significant effects were found, as in the case of using energy acquisition rate as the dependent variable.

For instance, a statistically significant effect of probability of using the bottom level was found: that is, using more explicit processing (explicit rules) was beneficial (up to a certain extent). This was the same as the previous analysis using energy acquisition rate as the dependent variable, for the same reason. A significant effect of learning rate was also found: a higher learning rate was better (up to a certain extent). This was also the same as the previous analysis using energy acquisition rate, for the same reason.

However, unlike the previous analysis using energy acquisition rate, in this case, a significant effect of generalization threshold was found: using a higher generalization threshold was worse, presumably because it led to too few explicit general rules (see Sun et al., 2001 for an analogous situation).

In terms of two-way interactions, as before with regard to energy acquisition rate, here an interaction between probability of using the bottom level and strategy was also found. As before, no central store was the best strategy with less explicit processing but became worse off with

more explicit processing, which might be attributed to the same reason as speculated before.

The analysis using lifespan as the dependent variable led to essentially the same conclusions, and thus details are omitted here.

7.2.4. Discussion

In previous social simulations, often rather arbitrary assumptions were made, simply because they were needed for producing simulations that matched observed data. Here assumptions were instead made at a lower level, in a more cognitively realistic way. By using cognitively realistic models of individuals in a social simulation, one may generate explanations of social phenomena based on individual cognitive processes (among other possibilities). This allows one to do away with many assumptions that are not cognitively grounded.

In this simulation, interactions occurred (1) between cognitive parameters and physical environmental variables (such as food availability) and (2) between cognitive parameters and social variables (such as survival strategy). Let us look into such interactions.

For one thing, the relation between various cognitive parameters and physical environmental variables may be such that certain cognitive attributes are universally good or bad (e.g., a lower probability of using the bottom level, up to a certain point, is always better for performance), while the effects of some other cognitive attributes (such as learning rate) are more dependent on environmental conditions (such as food availability, as shown by the analyses earlier). Existent cognitive attributes may have been selected (through evolution) to work within certain physical environments, which may be termed *cognitive-environmental dependency* (Cosmides & Tooby, 1994).

Similarly, some cognitive attributes have universal effects for all possible values of a social variable, while other attributes (such as probability of using the bottom level) have less universal effects and depend more on specific values of a social variable (such as survival strategy). Consequently, the relation between various cognitive parameters and social variables indicates that what social attributes, for example, institutions or norms, are adopted may have something to do with cognitive abilities and cognitive tendencies of individuals involved (Tetlock & Lebow, 2001; Boyer & Ramble, 2001; Atran & Norenzayan, 2004; Kluver et al., 2005). This relation may be termed *social-cognitive dependency*. This point has significant

ramifications: there may be some social institutions that are suitable for certain cognitive characteristics while unsuitable for certain others. They may not be universally better or worse than others. It may in fact depend on a host of other factors, in particular cognitive factors. Sun (2006) and Sun (2012b) provided substantial discussions of the close relationship between cognitive and social processes, and advocated the exploration of cognitive principles of sociocultural processes (Boyer & Ramble, 2001; Lustick, 2000).

The same point can be made of the dependency of social institutions on characteristics of physical environments (e.g., based on the interaction between strategy and food availability). This relation may be termed *social-environmental dependency* (Doran et al., 1994; Reynolds, 1994). Recent evidence (e.g., discussed by van de Vliert, 2013) has borne out such predictions.

Finally, in the reverse direction of social-cognitive dependency, cognitive attributes may have been selected (through evolution) to work with certain social and cultural environments (Zerubavel, 1997; Kluver et al., 2005), which may be termed *cognitive-social dependency*. One may explore sociocultural principles of cognition, the opposite of cognitive principles of sociocultural processes mentioned earlier (Durkheim, 1962; Bourdieu and Wacquant, 1992).

Together, these types of dependencies form a complex system of interwoven relationships. In such a system, it is important to understand not just direct effects of dependencies but also indirect effects that are not obviously related to their causes but are often crucial for discerning the functional relationships and structures of the system.

In summary, it has been shown to some extent that, in the context of different social survival strategies and different physical environments, cognition matters. It determines, for instance, which strategy and other social variables are appropriate under what cognitive conditions. Phenomena at the social level may be related to, or affected by, cognitive processes at the individual level. That is, they point to the micro-macro link between the social and the psychological (Alexander et al., 1987; Sawyer, 2003; Sun, 2001). Several hypotheses in this regard were generated through the simulation. Even though only very simple sociocultural processes were involved in this work, with the cognitive architecture used, some important interactions were nevertheless found.

7.3. Motivation and Survival

To further explore the effects of other psychological factors in addition to cognitive factors (in the narrow sense, as explored above), another set of simulations was conducted. This set of simulations, while exploring a different set of issues, was essentially based on the same task as described above. This set of simulations was originally described in Sun and Fleischer (2012).

7.3.1. Simulation Setup

In this set of simulations, the world was made up of a 200 × 200 grid, as opposed to the 100 × 100 grid in the previous simulation. Each of these 40,000 locations might contain (at most) one food item. At the beginning and every 40 cycles, the grid was replenished: randomly selected locations were restocked with food items, until the grid had 2,400 food items. A more benign condition, in which 3,600 locations contained one food item each, and a harsher condition, in which 1,200 locations contained one food item each, were also tested.

As in the previous simulation, a food item contained 50 energy units. Each agent began with 60 units of energy, and consumed one unit of energy per cycle. There were initially 120 agents to begin with, and the number of agents fluctuated due to birth and death, within the bound of a maximum of 120 agents.

As in the previous simulation, at each moment, each agent was located in a square on the grid. It faced a certain direction (north, south, east, or west). Each agent received inputs regarding the location of the nearest food, relative to its current position and its current direction. Its perception was divided into four pie-slice-shaped quadrants. Each agent could generate an action output: either (1) turn 90 degrees right, (2) turn 90 degrees left, (3) move forward, (4) pick up food and contribute a portion, (5) pick up food and keep all of it, or (6) reproduce (which is different from the previous simulation). Each agent lived for a maximum of 350 cycles, but might die early due to lack of food.

As in the previous simulation, procreation was asexual. Procreation occurred if an agent had reached 120 energy units or more, and there were fewer than the maximum number of agents in the world. The new agent was placed in a random location. The parent handed out 60 energy

units to the child upon birth. The child inherited its parent's internal makeup, although there was a 10% chance of mutation, as in the previous simulation.

In case that a central store was involved, an agent was required to contribute 20 energy units to the central store when it picked up a food item (50 energy units). At each cycle, agents with 10 or less energy units might receive 5 energy units each from the central store. Up to a maximum of 10% of the agent population might get energy from the central store at each cycle.

Each individual, when picking up a piece of food, decided whether to contribute to the central store or not. There were three variations on cheater detection and punishment. In the first variation, individuals might freely choose to contribute or not with no chance of being caught or punished. The second variation was the default setting used for most simulation runs, in which there was a 30% chance of a cheater being caught and fined 40 units of energy (with the fine being added to the central store). The third variation entailed a 100% chance of catching cheaters and the fine was all 50 units of energy provided by the food item found.

Agents were based on Clarion, involving the ACS, the MS, and the MCS (Sun and Fleischer, 2012). Within the MS, as befitting the nature of this task, considerations might be limited to three drives: *Food*, *Reproduction*, and *Honor* (see Chapter 4). Let us look into relevant parameters for these drives (as required by the drive equations).

The *deficit* parameter of the *food* drive was inversely proportional to the amount of energy that an individual currently had: $deficit_{food} = 1 - \left(\frac{energy_a}{500}\right)$, where $energy_a$ was the amount of energy the agent had. The *deficit* of the *reproduction* drive was proportional to the amount of energy the agent had: $deficit_{reproduction} = \left(\frac{energy_a}{200}\right)$. The *deficit* of the *honor* drive was kept constant at 1.0. The *baseline* parameters of these drives were set at 0.

On the other hand, the *stimulus* parameters for the *food* and *reproduction* drives were kept constant at 1.0 (because, e.g., food was almost always in sight), while the *stimulus* for *honor* was based on what others were doing, equal to the total number of times any individual had contributed to the central store divided by the total number of times anyone had picked up a piece of food (whether or not it then contributed to the central store): $stimulus_{honor} = f_s / f_t$, where f_s was the total number of times

anyone had contributed to the central store, and f_t was the total number of times anyone had acquired a food item.[3]

An individual could have one of the three goals: *hoardfood*, *sharefood*, and *reproduce*. *Hoardfood* was highly associated with the *food* drive (*relevance* = 1.0; see the goal strength equations in Chapter 4). *Sharefood* was moderately associated with the *food* drive (*relevance* = 0.5) and highly associated with the *honor* drive (*relevance* = 1.0). *Reproduce* was highly associated with the *reproduction* drive (*relevance* = 1.0).

The reinforcement needed for reinforcement learning in the ACS was generated by the MCS. The feedback for the reproduction action was:[4]

$$feedback = 0.8\left(\frac{energy_a}{200}\right) + c$$

$$c = \begin{cases} 0.2, & \textit{if the agent's goal is ``reproduce''} \\ 0.0, & \textit{otherwise} \end{cases}$$

Note that in the equation above, a feedback "bonus" c was provided when one achieved the current goal. (This bonus might result from, for example, the positive sense of accomplishing a goal.)

The feedback for picking up a food item and keeping all of it (without being caught) was:

$$feedback = 0.8\left(\left(\frac{e_f - e_p}{e_f}\right) deficit_{food} - \left(\frac{e_s}{e_f} * \frac{f_s}{f_t}\right)\right) + c$$

$$c = \begin{cases} 0.2, & \textit{if the agent's goal is ``hoardfood''} \\ 0.0, & \textit{otherwise} \end{cases}$$

where e_f was the amount of energy in a food item (i.e., 50 in this case), e_p was set to *0* in this case, $deficit_{food}$ was the *deficit* of the *food* drive, e_s was the amount of energy that one was supposed to contribute to the central store (i.e., 20 in this case), and f_s and f_t were defined before. In the equation, $\left(\frac{e_s}{e_f} * \frac{f_s}{f_t}\right)$ represented the internal feeling of guilt resulting from not conforming to the social norm, and it was proportional to the general ratio of contribution to the central store (by all individuals).

3. If any of these equations produced a number less than 0 or greater than 1, it was set to 0 or 1.

4. If an individual did not have enough energy (i.e., had less than 120 energy units), no child was produced, and a feedback of 0.0 was given.

If an individual attempted to hoard food and was caught and penalized (fined), the feedback became:

$$feedback = 0.8\left(\left(\frac{e_f - e_p}{e_f}\right) deficit_{food} - \left(\frac{e_s}{e_f} * \frac{f_s}{f_t}\right)\right) + c$$

where e_p was the amount of energy taken as penalty, and c was equal to *0*. The "bonus" c was absent in this case because, even if the goal was *hoardfood*, it had not succeeded and therefore could not get the positive feedback associated with goal accomplishment.

If an individual chose to contribute to the central store, the feedback was:

$$feedback = 0.8\left(\left(\frac{e_f - e_s}{e_f}\right) deficit_{food} + \left(\frac{e_s}{e_f} * \frac{f_s}{f_t}\right)\right) + c$$

$$c = \begin{cases} 0.2, & \textit{if the agent's goal is "sharefood"} \\ 0.0, & \textit{otherwise} \end{cases}$$

Clearly, feedback as determined above can also be mathematically expressed in terms of the drive strengths of relevant drives. In other words, one may view feedback as determined (in part) by drive strengths (in addition to goals and other factors).

Internal parameters within Clarion were adjusted. A genetic algorithm was used to determine the default values for these parameters, which roughly corresponded to the prior evolutionary history. That is, a GA was used as a rough approximation of a long evolutionary history that shaped the parameters of individual psychological processes (see Sun & Fleischer, 2012 for details; cf. Cosmides & Tooby, 1994; Wynn, 2002).

Based on the results of the GA, some parameters were fixed at the values found by the GA throughout the simulations in order to simplify the simulations to make them manageable. The other parameters were varied around the default values found by the GA (see below for details) to represent individual differences.

Those parameters that were varied to ascertain their effects included: (1) the learning rate of the neural network at the bottom level of the ACS (α), (2) the probability of using the bottom level as opposed to the top level ($prob_{BL}$), (3) the generalization threshold ($threshold_{gen}$), and (4) the drive gain parameters ($gain_{food}$, $gain_{reproduction}$, and $gain_{honor}$). Also varied were (5) survival strategy and (6) food availability, as mentioned earlier.

Table 7.2. Tested values of cognitive, social, and environmental variables. Asterisks indicate default values.

Probability of using the bottom level:	
#1	0.1 *
#2	0.5
#3	0.8
Learning rate:	
#1	0.01
#2	0.10
#3	0.50
#4	2.00 *
Generalization threshold:	
#1	2.0
#2	4.0 *
#3	8.0
Food availability:	
#1	1200 (poor)
#2	2400 (average) *
#3	3600 (abundant)
Survival strategy (where *cs* = central store):	
#1	cs/strong enforcement
#2	cs/weak enforcement *
#3	cs/no enforcement
#4	no cs

Considering the complexity and the computational cost, it was necessary to focus on a small set of values for each of the parameters (usually 2-4). See tables 7.2 and 7.3.

In this simulation, similar to the previous simulation, a small set of important metrics was used: average energy acquisition per agent per cycle, average population size (average number of agents in a population), and average lifespan (average age at death).

In the previous simulation, parameters were varied simultaneously in a factorial design. Here, parameters were varied relative to a single baseline condition. Each condition was identical to the baseline condition, except for one or two parameters that were assigned different values.

Table 7.3. Tested values of the motivational variables. Asterisks indicate default values.

$Gain_{food}$:	
#1	1.0*
#2	0.5
$Gain_{reproduction}$:	
#1	1.0*
#2	0.5
$Gain_{honor}$:	
#1	1.0*
#2	0.5

No more than two parameters were varied from their default values in any given run.[5] There were a total of 159 combinations. Each unique combination of parameter values was run 20 times, and each run lasted for a maximum of 2000 cycles.

7.3.2. Simulation Results

7.3.2.1. Effects of Social and Environmental Factors

Let us look into the social and environmental variables to see how they affected the overall performance of the tribal society.

Statistical analysis showed that the survival strategy had a significant effect on lifespan. See Figure 7.7. Generally speaking, CS with strong enforcement was better than CS with weak enforcement; CS with weak enforcement was better than CS without enforcement; CS without enforcement was better than no CS. These were expected, and were consistent with the findings from the previous simulation (as described in the previous section).

Statistical analysis also showed that the survival strategy had a significant effect on population size. However, average population size was incidentally calculated based only on those periods when the population was not extinct. This calculation tended to cause the average population size in the conditions where extinction often occurred to be misleadingly large. In hindsight, the population size should have been averaged over all the cycles rather than only the ones where there were agents alive.

5. Such an approach did sacrifice some rigor, but allowed a greater range of parameters to be tested, avoiding the exponentially increasing time cost of a factorial design.

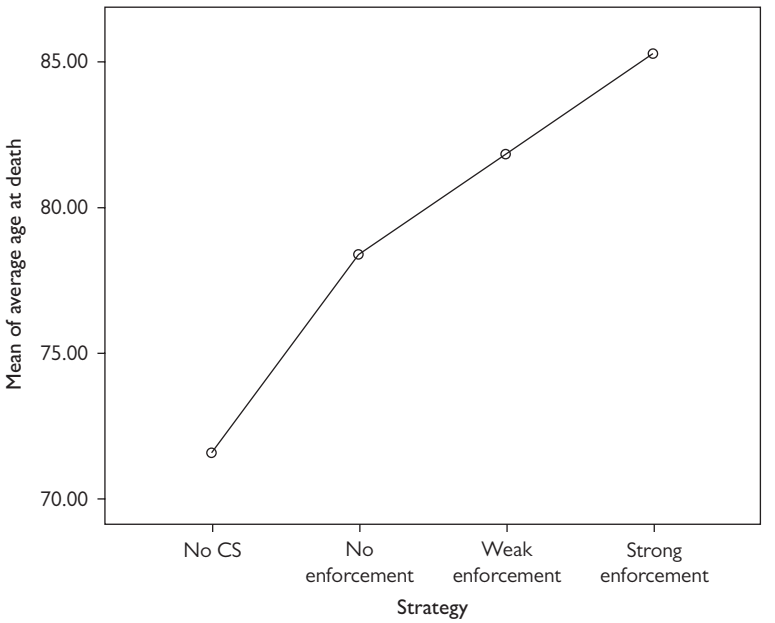

Figure 7.7. The effect of survival strategy on lifespan. The *y*-axis represents lifespan. The *x*-axis represents strategy.

Therefore, the results concerning population size should be looked at with this caveat in mind (Figure 7.8).

In terms of energy acquisition, the results were mixed: while in some analysis the difference was not statistically significant,[6] in some other analysis there was a significant difference as in the previous simulation. See Figure 7.9. The pattern in the results was generally consistent with the previous simulation.

7.3.2.2. Effects of Cognitive Factors

Turn now to cognitive factors within the ACS (such as learning rate and probability of using the bottom level), to see how they affected performance.

6. This was probably because in this case the existence of the central store did not make people significantly lazier or less competent; there were still sufficient incentives to obtain as much energy as possible (e.g., for the sake of reproduction).

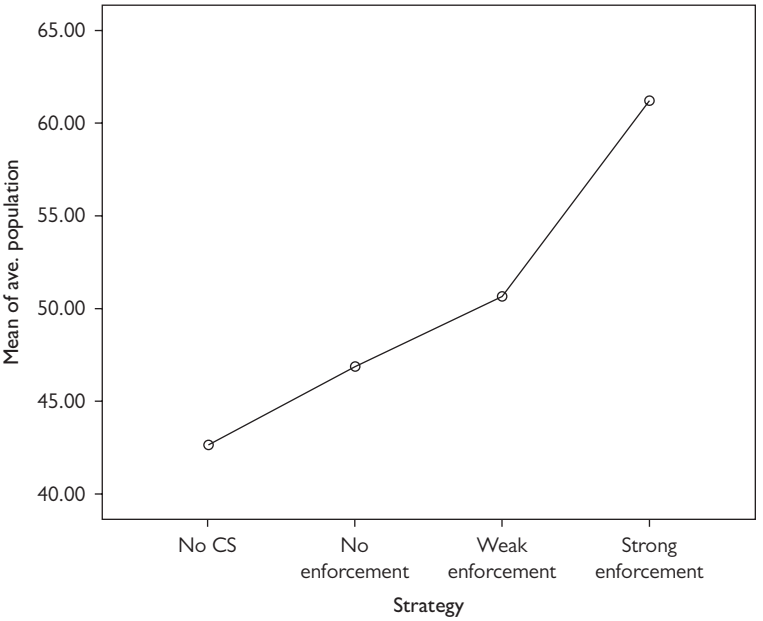

Figure 7.8. The effect of survival strategy on population size. The *y*-axis represents population size. The *x*-axis represents strategy.

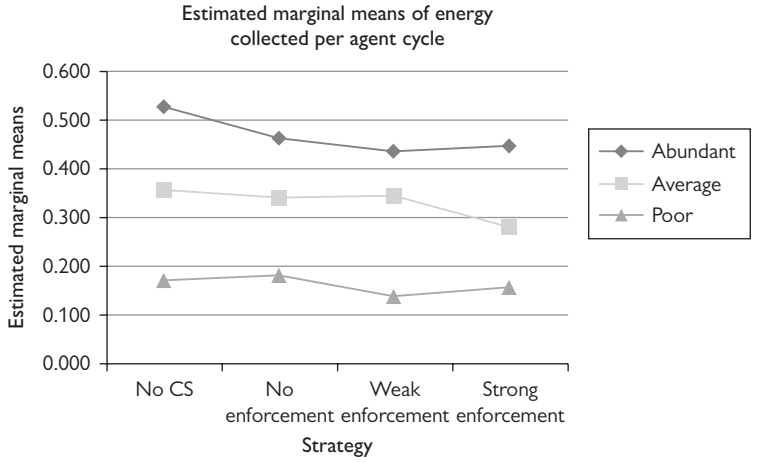

Figure 7.9. The effect of survival strategy on energy acquisition. The *y*-axis represents energy acquisition rate. The *x*-axis represents strategy. The different lines indicate different food availability.

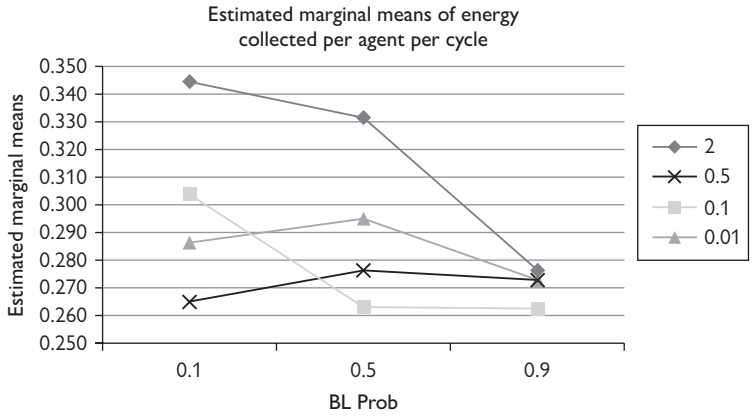

Figure 7.10. The effect of learning rate on energy acquisition. The *y*-axis represents energy acquisition rate. The *x*-axis represents probability of using the bottom level. The different lines indicate different learning rates.

Some analyses indicated that there were statistically significant effects of learning rate on energy acquisition and population size, while some other analyses failed to reach statistical significance. Generally speaking, higher learning rates worked better (for energy acquisition; while population size was not a reliable measure as indicated earlier), probably because they led to more learning and therefore better knowledge and skills, consistent with the findings from the previous simulations. See Figure 7.10.

Likewise, some analysis indicated that there were statistically significant effects of probability of using the bottom level on energy acquisition, population size, and lifespan, while some other analyses failed to reach statistical significance. Generally speaking, lower probabilities of the bottom level worked better (for energy acquisition and for lifespan; while population size was not a reliable measure as indicated earlier), probably because they led to more reliance on explicit processes, which were more precise, consistent with the result of the previous simulation. See Figure 7.11 and Figure 7.12.

Statistical analysis also showed that there was a significant interaction between probability of using the bottom level and environment (food availability) on population size. While, as indicated before, population size in this simulation could not be used as a measure indicative of actual performance, statistical differences with regard to this measure might still be pertinent. This result indicated that some environmental conditions required more reliance on explicit processes than others.

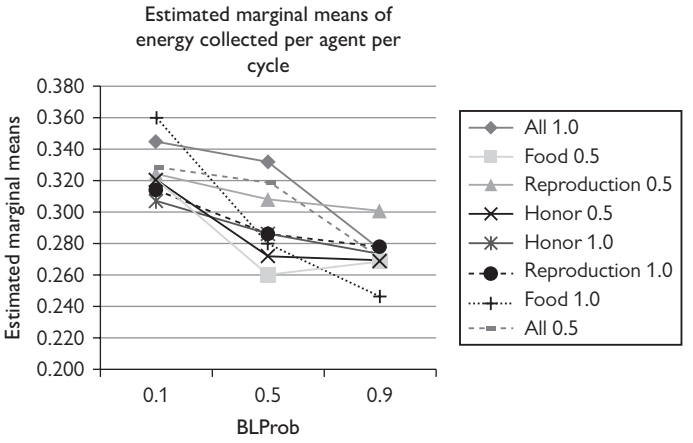

Figure 7.11. The effect of probability of using the bottom level on energy acquisition. The *y*-axis represents energy acquisition rate. The *x*-axis represents probability of using the bottom level. The different lines are for different drive gain settings (see the next subsection).

7.3.2.3. Effects of Motivational Factors

Now I turn to effects of motivational factors, which was not investigated in the previous simulation. As predicted, motivational factors had a significant effect on outcomes.

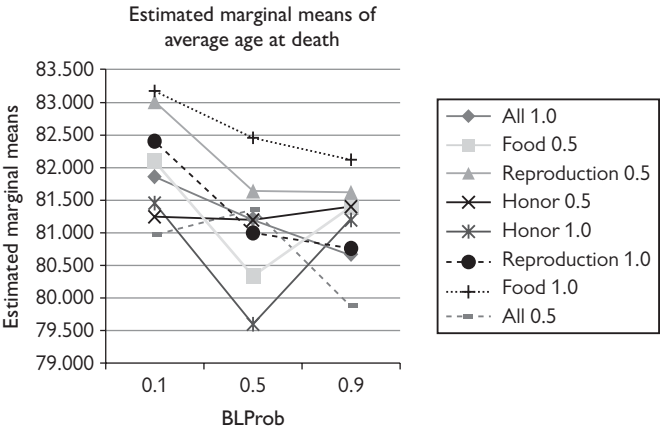

Figure 7.12. The effect of probability of using the bottom level on lifespan. The *y*-axis represents lifespan. The *x*-axis represents probability of using the bottom level. The different lines are for different drive gain settings (see the next subsection).

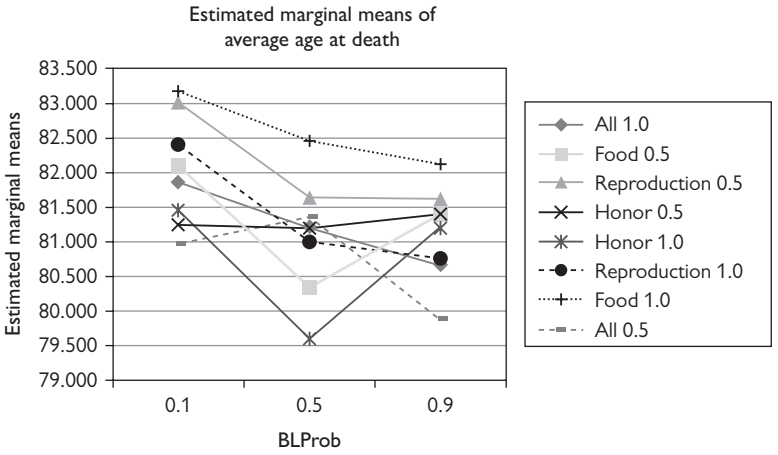

Figure 7.13. The effect of gains on lifespan. The *y*-axis represents lifespan. The *x*-axis represents probability of using the bottom level. The different lines represent different drive gain settings.

In this regard, "*gains*" was a variable created for the sole purpose of analysis. It consolidated the three drive gain parameters into one. There were eight values, ranging from "*All 0.5*" to "*All 1.0*", with "*All 1.0*" being the default. See Figure 7.13 for the complete list, where, for example, "*Honor 0.5*" meant that the *gain* parameter of the *honor* drive was 0.5 and all the other drive gains were 1.0.

There was a statistically significant effect of "*gains*" on lifespan from some analyses. Generally speaking, more emphasis on food (a higher drive gain for *food*) led to better performance (e.g., "*Food 1.0*" or "*Reproduction 0.5*"). Reduced emphasis on food generally led to worse performance (e.g. "*Food 0.5*" or "*Honor 1.0*"). This was no surprise. See Figure 7.13.

There was a significant interaction between "*gains*" and environment on lifespan. An interpretation of this result was as follows: in a more benign environment, less focus on *honor* (e.g., "*Honor 0.5*") helped survival, but in a harsh environment, drive focuses did not make much difference because one had to focus only on food in order to survive. See Figure 7.14 for the data.

There was a significant interaction between "*gains*" and generalization threshold for population size. When the *gain* parameter of the *honor* drive was low (0.5), a higher generalization threshold was better; when it was high (1.0), a lower generalization threshold was better. This was probably

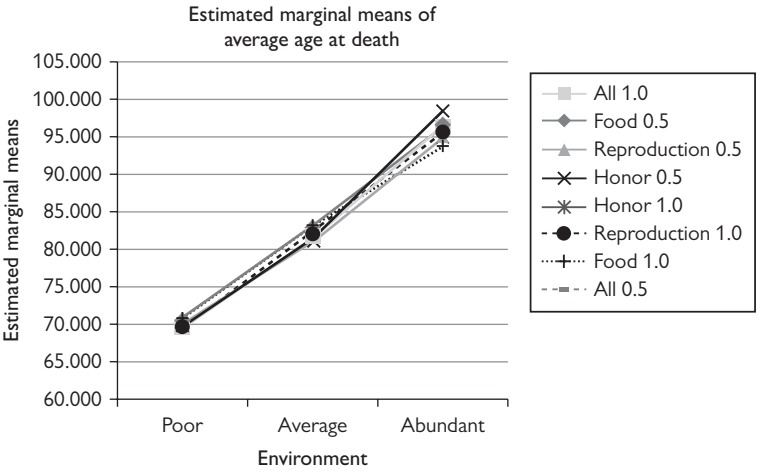

Figure 7.14. The interaction of gains and environment on lifespan. The *y*-axis represents lifespan. The *x*-axis represents environment. The different lines represent different drive gain settings.

because of the need to deal with the additional complexity of deciding whether to contribute to the central store or not: when the *gain* parameter of the *honor* drive was high (i.e., when one was more inclined to contribute), a more aggressive learning process (with a lower generalization threshold) might be better. Also, when the *gain* of the *reproduction* drive was low (0.5), a lower generalization threshold was better, probably because this drive setting led to a relatively high *gain* of the *honor* drive and therefore the interpretation above applied.

7.3.3. Discussion

The results of this set of simulations showed evidence for the interactions among cognitive, motivational, environmental, and social factors.

First, as discussed in the previous section, *cognitive-environmental dependency* suggests that what cognitive characteristics are the best for (and thus likely selected within) a certain population may be determined in part by its external, physical environmental conditions, as shown by, for example, the interaction between probability of using the bottom level and environment discussed earlier. Existent cognitive characteristics may have been selected (through evolution) to work within given physical environments (or, likewise, social conditions, motivational characteristics, and so on).

Second, *cognitive-motivational dependency* suggests that what cognitive characteristics are the best for (and thus likely selected within) a certain population may be determined in part by motivational characteristics, as shown by, for example, the interaction between generalization threshold and "gains" discussed earlier. Evolutionary explanations may apply to this case also.

Third, *motivational-environmental dependency* suggests that what motivational characteristics are the best for (and thus likely selected within) a certain population may be determined in part by physical environmental conditions, as shown by, for example, the interaction between "gains" and environment.

There are also many other types of dependency (e.g., as discussed in the previous section; see also Sun & Naveh, 2007 and Sun & Fleischer, 2012). Results as described above are generally consistent with the psychological and sociological literatures (such as the cognitive-motivational interaction, as in e.g. Markman & Maddox, 2005, or the social-environmental interaction, as in e.g. Doran et al., 1994).

One can compare the present simulations with that of Cecconi and Parisi (1998). The similarity is obvious (see the description earlier). However, significant differences exist. First, their model of individual agents was simple. Second, their model did not embody sufficiently realistic cognitive processes. Third, motivational processes were not present in their work. As a result of these three differences, the present work was able to explore significant effects of cognitive and motivational factors, while their model could not. Fourth, although they introduced social norms (such as contributing to central stores), different probabilities and rates of penalty for violating social norms were investigated in the present work. Fifth, detailed statistical analysis was conducted in the present work, which revealed some interesting points.

The work by Doran and associates (e.g., Doran et al., 1994) is also relevant here. In their model, each agent was a production system, made up of three parts. It had a working memory, in which "facts" were stored. Each agent also had a set of rules, based on which it made decisions. There was a mechanism that matched rules against the situation represented by the working memory. Agents could also send each other messages. Even though agents did not have any preconceived notion of groups, Doran et al. (1994) found that they formed groups that collectively carried out actions to obtain food. The simulation was used to explain the emergence of social complexity among prehistoric hunter-gatherers in

southwest France during the Upper Paleolithic period. The changes (e.g., larger bands with a clearly identified leader, status differences, rituals, and so on) emerged from local interactions among agents, given the deterioration of resource availability. The differences between their simulation and the present work include the fact that the Clarion-based model was more psychologically realistic, although their work involved more interagent relations, including communications.

NewTies (e.g., Gilbert et al., 2006) was a simulation project that investigated the emergence of social behavior to address environmental challenges analogous to those that human societies had been able to overcome. The simulation project could serve as a test bed for examining a wide range of social theories. Whereas one could not experiment on human societies, one could on such artificial societies. The goal of that project was consistent with the present work, although they did not address the cognitive and motivational issues addressed here.

The present simulations can be further enhanced. For instance, one could explore more realistic models of evolution. Not only psychological parameters, but also social institutions and culture can be modeled in detail and evolved. For another instance, asexual reproduction can be changed to a more realistic setting (which has already been explored in follow-up work). Finally, findings from the simulations should be more rigorously validated empirically. Other aspects of the simulations can also be enhanced.

7.4. Organizational Decision Making

Now I turn to organizational structures and cognition and examine how they affect collective decision making. The discussion below draws from Sun and Naveh (2004).

7.4.1. Organizational Decision Task

Classification is a typical task faced by organizations. In a classification task, individuals may gather information about problems, classify them, and then make further decisions based on the classification. For instance, a bank may classify a household as financially promising or unpromising, and on that basis decide to approve or reject a loan.

Within organizational research, Carley, Prietula, and Lin (1998) introduced a classification task involving different types of organizational structures and agents. By varying agent type and organizational structure, they studied how these factors interact with each other. Sun and Naveh (2004) built on that research to explore the interaction of cognition and organizational design, with a psychologically realistic agent model.

In the classification task, in each case, there is a single object in the airspace. The object has nine different attributes, each of which can take on one of three possible values (e.g., its speed can be low, medium, or high). An organization must determine the status of the observed object: whether it is friendly, neutral, or hostile. Hence, this is a ternary choice task. There are a total of 19,683 possible objects, and 100 problems are chosen randomly (without replacement) from this set.

In this task, the true status of an object can be determinable by adding up all nine attribute values. If the sum is less than 17, then it is friendly; if the sum is greater than 19, it is hostile; otherwise, it is neutral.

However, no one single agent has access to all the information necessary to make a decision. Collective decisions are made by integrating separate decisions made by different agents, each of which is based on a subset of information, through a specific organizational structure.

There are two types of organizational structures: (1) teams, in which agents make their individual decisions and the organizational decision is determined by the majority; and (2) hierarchies, in which decision recommendations are passed from subordinates to superiors, and the decision of a superior is solely based on the recommendations of his or her subordinates. In this task, a two-level hierarchy with nine subordinates and one superior is considered.

Organizations are also distinguished by information accessible to agents. There are two varieties: (1) distributed access, in which each agent sees a different subset of three attributes and no two agents see the same subset of three attributes, and (2) blocked access, in which three agents see exactly the same subset of attributes. In both cases, each attribute is accessible to three agents.

Carley et al. (1998) considered several agent models. Among them, CORP-ELM produced the most probable classification based on an agent's experience, CORP-P-ELM produced a classification stochastically in accordance with the estimate of the probability of each classification based on an agent's experience, CORP-SOP followed the organizationally

prescribed standard operating procedure (which involved summing up the values of the attributes available to an agent), and Radar-Soar was a model built in Soar based on explicit search through problem spaces (Rosenbloom et al., 1993).

In Carley et al. (1998), human experiments were done in a 2 × 2 fashion (organizational structure × information access). In addition, the human data were compared to the simulation results from the aforementioned four models. Their data appeared to show that agent model type interacted with organizational design (team versus hierarchy and blocked versus distributed information access). The human data and the simulation results were as shown in Table 7.4.

Their human data showed that, generally speaking, humans performed better in team situations. Moreover, distributed information access was generally better than blocked information access. The worst performance occurred when hierarchical organizational structure and blocked information access were used in conjunction.

Their human and simulation data also suggested that which organizational design (team versus hierarchy and blocked versus distributed information access) exhibited the highest performance depended on the type of agent. For example, human subjects performed best as a team with distributed information access, while Radar-Soar, CORP-ELM, and CORP-P-ELM performed best in a team with blocked information access. Relatedly, the adaptivity of agents tended to hinder the performance of hierarchal organization; with a nonadaptive agent such as CORP-SOP, there was no difference between the two organizational structures.

The results above brought up the issue of the interaction between organizational design and agent type (i.e., agent intelligence level). However, the agent models that were used there were simple. Therefore,

Table 7.4. Human and simulation data for the organization task from Carley et al. (1998). D indicates distributed information access; B indicates blocked information access. All numbers are percent correct.

	Team(B)	Team(D)	Hierarchy(B)	Hierarchy(D)
Human	50.0	56.7	46.7	55.0
Radar-Soar	73.3	63.3	63.3	53.3
CORP-P-ELM	78.3	71.7	40.0	36.7
CORP-ELM	88.3	85.0	45.0	50.0
CORP-SOP	81.7	85.0	81.7	85.0

the intelligence level in these models was generally low (including, to a large extent, the Soar model, which essentially encoded a set of simple rules). Learning in these models was also rudimentary: there was no sophisticated learning as one might observe in humans.

These shortcomings suggested that it was worthwhile to undertake a simulation that involved a more psychologically realistic agent model. Moreover, with the use of a more psychologically realistic model, the roles of different cognitive capacities, parameters, and other details in affecting organizational performance might be investigated.

7.4.2. Simulations and Results

Below, four simulations are described. In the first simulation, the aforementioned task was tackled with Clarion-based agents. The second simulation extended the duration of agent training. In the third simulation, a range of cognitive parameters of the agent model was varied in a factorial design. The point was to explore the interaction of cognitive factors with organizational design. In the fourth simulation, instead of using exactly the same agent model, organizations with different agents were investigated.

7.4.2.1. Simulation I: Matching Human Data

This simulation used the same setup as the original study described above (Carley et al., 1998), but substituted Clarion-based agents for the simpler agents used previously. The goal was to study the effects of organizational structure and information access on performance, as in the original study but in the context of the more psychologically realistic agent model.

The ACS of Clarion was used for capturing individual decision making. At the top level of the ACS, RER was used to extract rules. At the bottom level, there was a neural network. The network received feedback of 0 or 1 after each step, depending on whether the target was correctly classified. Due to the immediate feedback, simplified Q-learning was used. All agents ran under a single, uniform set of cognitive parameters (although these parameters were varied later).

The results of the simulation were as shown in Table 7.5. 4,000 training cycles (each corresponding to a single case with a decision by the organization) were used for each group. The results closely accorded with the patterns of the human data, with teams outperforming hierarchies,

Table 7.5. Simulation data for agents running for 4,000 cycles. Performance for Clarion is computed as percent correct over the last 1,000 cycles. The human data from Carley et al. (1998) are included for comparison.

.	Team(B)	Team(D)	Hierarchy(B)	Hierarchy(D)
Human	50.0	56.7	46.7	55.0
Clarion	53.2	59.3	45.0	49.4

and distributed access outperforming blocked access. Also, as in humans, performance was not grossly skewed but was roughly comparable across all conditions, unlike some of the simulations from Carley et al. (1998). The match with the human data was better than the simulations in the original study as described earlier.

Note that under the hierarchal conditions, performance was worse. In these conditions, two layers of agents were being trained, with the output of the upper layer depending on those of the lower layer. In addition, the higher input dimensionality of the supervisor (nine inputs as opposed to three inputs for a subordinate) increased the complexity, leading to slower learning. This was analogous to human learning, where input dimensionality was known to be one of the chief determinants of performance. Note also that in this simulation, distributed information access led to better performance than blocked access, probably because distributed access provided more diversified information sources.

7.4.2.2. Simulation II: Extending Simulation Temporally

Thus far agents trained for 4,000 cycles were considered, and the results were analogous to those of humans. However, it would be interesting to see what would happen if the length of training was extended. In particular, it would be interesting to see if the patterns observed above would be preserved in the long run.

As Figures 7.15–7.18 showed, learning could occur over 20,000 cycles (rather than 4,000 cycles). Previously, the best performing condition was team with distributed information access. This condition continued to improve slowly after the first 4,000 cycles, but was overtaken by team with blocked access. Thus, it seemed that while teams benefited from diversified information in the early phases of learning, a well-trained team with redundant information (i.e., blocked access)

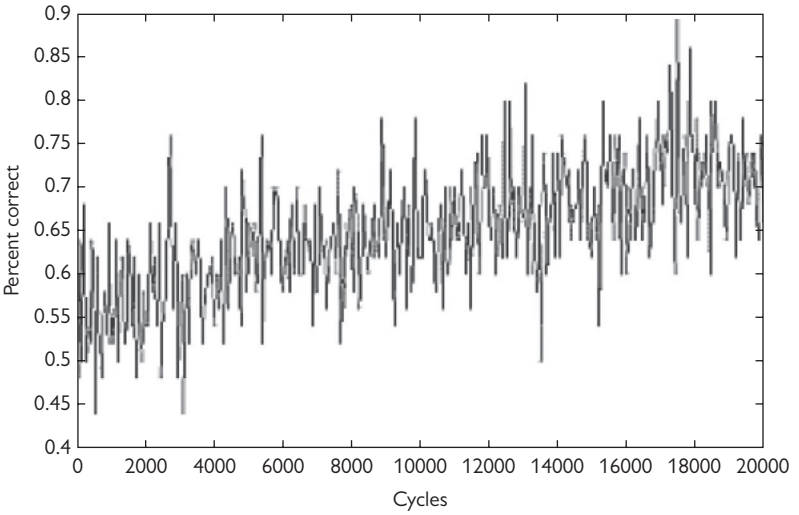

Figure 7.15. Training curve (team organization, distributed access).

performed better in the long run due to redundancy and thus less fluctuation.

Similarly, in hierarchal organizations, there was either a reversal or disappearance of the initial trends. Hierarchies with distributed information access produced the best and also the most stable performance. Likewise,

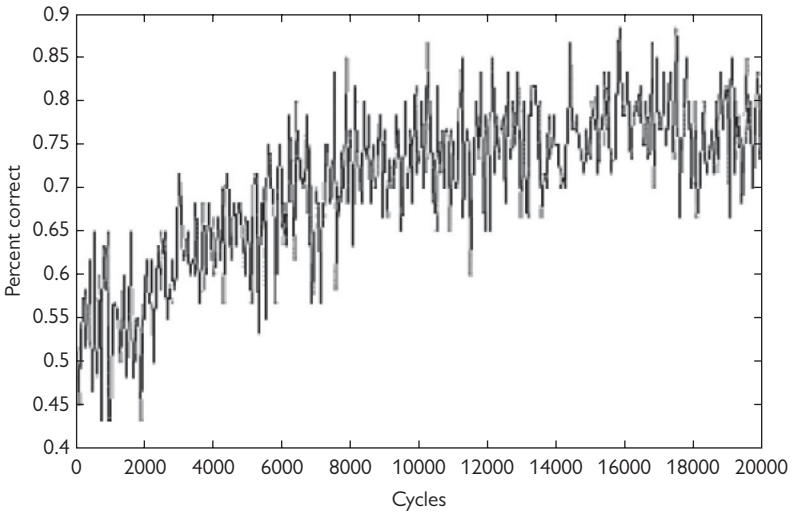

Figure 7.16. Training curve (team organization, blocked access).

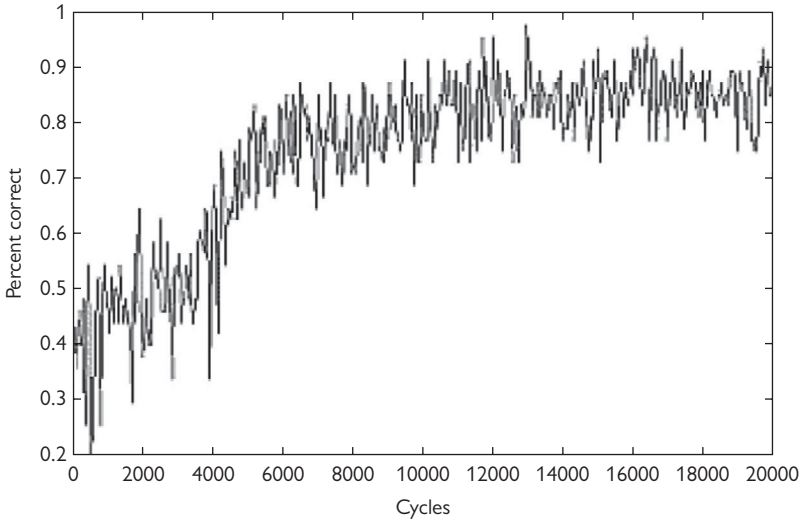

Figure 7.17. Training curve (hierarchical organization, distributed access).

a hierarchy with blocked access, although previously performed poorly, showed significant improvements. Whereas hierarchies took longer to train, their performance was superior in the long run. In a hierarchy, a well-trained supervisor was able to synthesize many data points in a more sophisticated way than a simple voting process. Likewise, the reduced

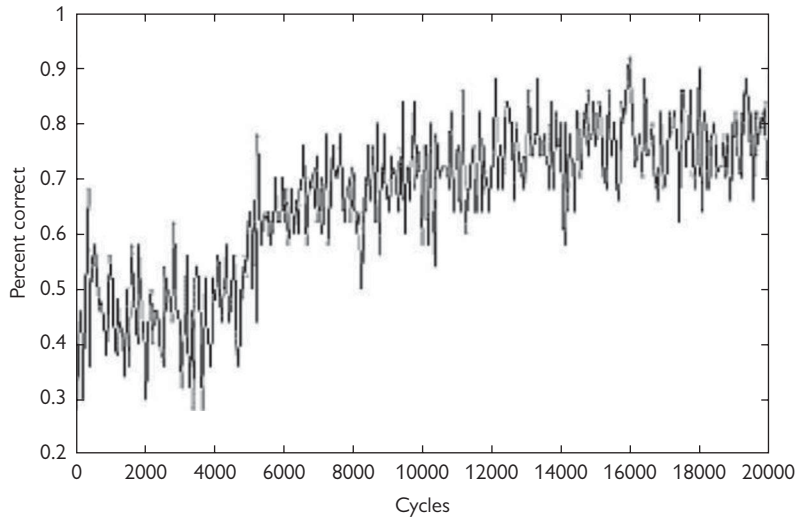

Figure 7.18. Training curve (hierarchical organization, blocked access).

individual variation in blocked access led to less fluctuation in performance in the long run.

There is a lesson here: limited data can only allow one to draw limited conclusions—only with regard to the specific condition under which the data were obtained. Researchers may overgeneralize their conclusions, which can only be remedied by more extensive investigations. Given the high cost of human experiments, simulation, especially social simulation with a cognitive architecture, has a significant role to play in exploring alternatives and discovering possibilities (Sun & Naveh, 2004, 2007).

7.4.2.3. Simulation III: Varying Cognitive Parameters

Clarion includes a wide range of cognitive-psychological mechanisms and processes, and its parameters are generic, not task-specific. Thus, one can study specific issues, such as organizational design, in the context of a general theory of cognition-psychology. In this simulation, cognitive parameters were varied within the ACS to examine their effects on collective performance. Analogous to varying training length earlier, varying cognitive parameters also allowed one to see the variability of results and thus avoid overgeneralization.

Parameters were varied in a factorial design, in order to examine both the effects of individual parameters and their interactions with each other. Two sets of parameters were separately varied to avoid the prohibitively high computational cost of varying all parameters simultaneously. The first set of parameters consisted of fundamental parameters of the ACS, including: (1) probability of using the bottom level, (2) learning rate of the neural network at the bottom level, (3) temperature (degree of randomness in action selection). The second set consisted of parameters related to rule extraction at the top level, including: (1) RER positivity threshold, (2) RER generalization threshold, and (3) RER rule density (which determined how often a rule must be invoked in order to be retained). (See Chapter 3 regarding details of these parameters.)

Each of the two sets, along with information access and organizational structure, was varied in a factorial design, so that all combinations of all values of the parameters in a set were considered. For each parameter, two or three different values were tested.

Performance was examined at two points of the learning curve—after some initial training, because results at that point corresponded closely to the human data described earlier, and after more extensive training.

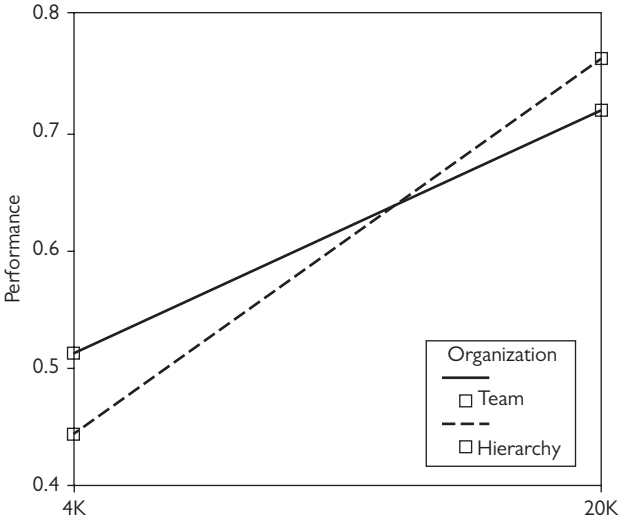

Figure 7.19. The effect of organization on performance over time. The *x*-axis represents training cycle. The *y*-axis represents performance in terms of percent correct. The two lines represent team and hierarchy respectively.

Therefore, performance was measured after 4,000 cycles and after 20,000 cycles.

Statistical analysis of the results confirmed the effects of organizational structure and information access to be significant. Moreover, the interactions of these two factors with length of training were significant. These interactions, as seen in Figures 7.19–7.20, reflected the trends discussed earlier: the superiority of teams and distributed information access at the early stage of the learning process, and either the disappearance or the reversal of these trends toward the end. This analysis showed that these trends persisted across a variety of settings of cognitive parameters and did not depend on any one setting.

The effect of probability of using the bottom level was likewise significant. More interestingly, its interaction with length of training was significant as well. As shown in Figure 7.21, explicit rule usage was very useful at the early stages of learning, when increased reliance on rules tended to boost performance. However, by the 20,000th cycle, this effect disappeared. This was because rules were crisp guidelines that provided a useful anchor at the uncertain early stages of learning. However, by the end of the learning process, they became too coarse grained to cover all possible contingencies and no more reliable than highly trained neural networks. This

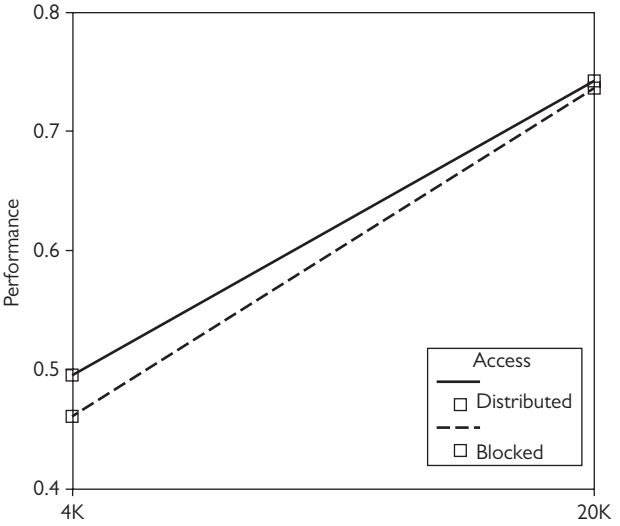

Figure 7.20. The effect of information access on performance over time. The *x*-axis represents training cycle. The *y*-axis represents performance in terms of percent correct. The two lines represent distributed and blocked access respectively.

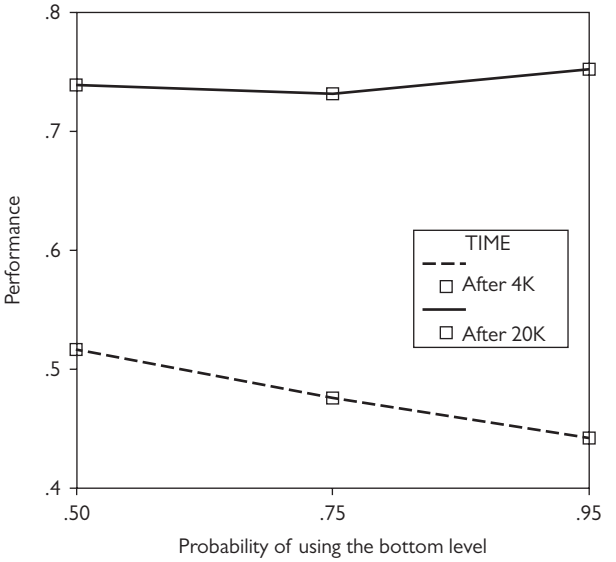

Figure 7.21. The effect of probability of using the bottom level on performance over time. The *x*-axis represents probability of using the bottom level. The *y*-axis represents performance in terms of percent correct. The two lines represent 4,000 and 20,000 training cycles respectively.

seemed to correspond to similar phenomena in human learning whereby explicit rule learning was widely used in the early stages of learning but was later replaced by implicit skills (Dreyfus & Dreyfus, 1987).

Predictably, the effect of learning rate was significant. However, there was no significant interaction between learning rate and organizational structure or information access, which suggested that the learning capability did not differentially benefit a hierarchy versus a team, or blocked versus distributed information access.

Now turn to the parameters related to rule extraction. As confirmed by statistical analysis, it was better to have a higher rule generalization threshold (up to a point). That is, if one restricted the generalization of rules to more successful ones, the result was a better rule set, which led to better performance.

Relatedly, whereas the effect of rule density was insignificant, the interaction between density and generalization threshold was significant. When rules were of relatively high quality (i.e., under a higher generalization threshold), it was better to have more of them available by lowering the density parameter. By contrast, when the quality of rules was lower (i.e., under a lower generalization threshold), it was advantageous to have a quicker forgetting process, as captured by a higher density parameter.

This simulation confirmed that which organizational structure (team versus hierarchy) or information access scheme (distributed versus blocked) was superior depended on the length of training. It also showed that some cognitive parameters (e.g., learning rate) had a monolithic effect, whereas in other cases, complex interactions of factors were at work. This illustrated, once again, the importance of limiting one's conclusions to the specific context in which data were obtained.

7.4.2.4. Simulation IV: Introducing Individual Differences

Thus far, only organizations with identical agents were considered. In the real world, however, organizations often consisted of individuals with widely varying cognitive capabilities. It would be interesting to further extend the simulation to capture this variability. One could observe how organizations varied in response to individual cognitive differences and determine whether certain organizations were better at dealing with such variations.

For instance, the case of a single slow learner was examined. Agents were organized in a hierarchy, and distributed information access was

used. All agents were identical, except for one agent who was a much slower learner than the others. At the end of the task, the supervisor's network (at the bottom level of its ACS) was analyzed by summing up the absolute values of the weights corresponding to the inputs from each subordinate agent, which allowed one to compare the relative influences of different agents on the supervisor's decision. The summed weights for the slow-learning agent were shown to be much lower than the other sums (by a factor of at least 2). In other words, a supervisor learned to pay less attention to the recommendations of the slow learner. Additionally, overall performance of the hierarchy dropped by only 3% to 4%. Thus, hierarchies were robust enough to deal with a single weak performer, showing only a slight degradation in performance.

For another instance, a situation with variable learning rates was examined. In this simulation, each agent had a different learning rate (instead of having just one agent that differed from the rest). The results of the simulation followed the same trends as reported earlier, with hierarchies outperforming teams (after 20,000 training cycles). However, here the margin by which teams were outperformed was significantly greater than when all agents were identical. This was because the decision-making process of a team—the majority vote—was less capable of taking individual differences into account. By contrast, a supervisor could rely more on one subordinate than on another, based on the past successes of their recommendations. A more extensive discussion of the impact of weak learners in an organization can be found elsewhere (e.g., Carley, Prietula, & Lin 1998; Sun & Naveh, 2004).

7.4.3. Discussion

In this task, a cognitively more realistic simulation with Clarion captured human data in organizational decision making. The Clarion-based model performed well across a variety of conditions, consistent with the human data. After a certain amount of training, the trends observed closely matched the human data. Thus, cognitive realism in social simulation could closely capture human results, even though social simulations tended to be at a higher level.

Moreover, by using Clarion, deeper explanations were formulated. For instance, the poorer performance of hierarchies early on might be attributed, at least in part, to the longer training time needed to deal with

higher dimensionality of information by the supervisor. Such explanations were only possible when the model was cognitively-psychologically realistic.

In addition to offering deeper explanations, cognitive-psychological realism can lead to greater predictive power for social simulation. In Clarion, one can vary cognitive parameters and test their effects on performance. In this way, Clarion may be used to predict organizational performance, and furthermore to help performance by prescribing optimal or near-optimal cognitive abilities for specific tasks and organizational structures.

Some prescriptions generated by Clarion may help to assign agents to organizational roles based on their cognitive capabilities. For instance, a hierarchy's performance hinges crucially on having a quick-learning supervisor. Furthermore, some other results generated by Clarion may help to formulate organizational policies. For instance, the importance of rule learning at the beginning of the learning process was observed. Based on this, an organization may emphasize rules (e.g., standard procedures) when training new personnel, but emphasize case studies when training experienced employees. Such prescriptions result from the cognitive-psychological realism of the model employed.

With greater cognitive-psychological realism, social simulation may be able to generate findings more meaningful for organizational design. It may happen that seemingly minor differences in cognition-psychology may make significant differences in terms of organizational performance. Conversely, seemingly significant differences in cognition-psychology may turn out to have little impact on collective performance. In many cases, there is no a priori way of predicting the effects of individual cognitive parameters, and therefore simulation may be useful. By varying cognitive (and other) parameters in this study, meaningful results were found. This was only possible with sufficient cognitive-psychological realism.

7.5. Academic Publishing

I now turn to examine the simulation of the development of academic science. The discussion draws from Naveh and Sun (2006).

7.5.1. Academic Science

Science may develop in a certain way following a certain pattern. In particular, it has been observed that the number of authors contributing a

certain number of articles to a scientific journal follows a highly skewed distribution (an inverse power curve). This distribution, known as a Zipf distribution, is common to a number of other phenomena in information science, such as the frequency of spoken words or of links on the World Wide Web. In the case of scientific publication, the tendency of authorship to follow a Zipf distribution was observed by Lotka (1926) and has been known as Lotka's law.

Simon (1957) proposed a simple stochastic process for capturing Lotka's law. One of his assumptions was that the probability that a paper was published by an author who had published i articles before was equal to a/i^k, where a was a constant of proportionality.

Gilbert (1997) modeled this phenomenon through social simulation. His simulation was based on the assumption that the system randomly selected a focal paper first, which was represented as a point in a two-dimensional space of ideas, and then it randomly selected a number of other papers, each of which occupied a different point and pulled the original point in its direction. The resulting paper would be located on that two-dimensional space based on the factors above. Papers were randomly assigned authors based on a stochastic process. To capture the constraint that academic papers must be original, a newly published paper must be at least m units away from any other existing paper (where m is a constant). Another assumption was that the number of papers produced in a given time period was determined by the number of papers in existence during the previous time period, by specifying a small probability of each existing paper acting as the seed for a new paper. Thus, papers spawned more papers, with authors serving only an ancillary role.

This model led to an idea space divided into clusters, which were assumed to correspond to different scientific areas. Each cluster originated in a few seminal papers and accumulated additional papers at an increasing rate over time. This model yielded publication trends consistent with human data, including Lotka's law described earlier. Although Gilbert's model captured to some extent the growth of academic science, it was not cognitively-psychologically realistic. It did not include many processes that were known to be important for scientific inquiry (e.g., learning, creativity, and so on).

However, by using a more cognitively realistic model, one could avoid many artificial assumptions (such as the assumption that papers automatically spawned more papers, or that researchers were randomly assigned authorship of specific papers). In this way, there would be more

distances between assumptions and outcomes, thereby generating deeper explanations.

7.5.2. Simulation Setup

In this alternative simulation, authors were not mere placeholders, but Clarion-based agents whose success or failure depended on their cognitive abilities. Successful authors would go on to publish many more papers, whereas unsuccessful authors would be removed from the system.

As in Gilbert's simulation, the scientific world consisted of papers, each of which proposed an idea, and of authors, who generated new papers through combining previous ideas (papers). In the present simulation, to publish a paper, an agent adopted a focal idea (represented by an existing paper), in accordance with some cognitive processes, implicit or explicit. The agent then used other ideas (other published papers), which pulled the original idea in different directions. In addition, the agent also performed local search to "optimize" the resulting idea. This reflected the fact that authors did not merely cobble together ideas from existing sources, but they also tried to refine the final product.

Possibility of failure to publish existed in this simulation, just as human authors could produce papers that were not publishable. This was in contrast to Gilbert's simulation, in which ideas were largely undifferentiated in terms of quality. Instead, in the present simulation, each agent had a set of evaluation functions that determined the quality of ideas. These functions corresponded to the important considerations in evaluating a scientific idea (e.g., clarity, insightfulness, empirical evidence, theoretical results, and application potential). However, just as researchers in the real world could not predict precisely when the results of their research would meet with approval and interest, agents' individual valuations of these functions might differ from the community valuation.

An agent, made up of the ACS, selected a focal idea and then a number of pull ideas. It learned through reinforcement learning (at the bottom level of the ACS). It naturally captured sequences of actions (i.e., selecting the focal idea, then the first pull idea, and so on). The feedback to agents was based on paper acceptance or failure (0 or 1). In addition, agents were provided with partial feedback at each major step of the paper generation process, equal to a fraction (one-third) of the unfinished paper's evaluation (as determined by an agent's own evaluation

function). This reflected the fact that agents were guided to a certain extent by their own experience.

At the same time, an agent might use RER to extract rules at the top level that determined how to choose focal ideas and how to choose pull ideas. These rules were used in conjunction with other rules concerning local search, which represented a priori knowledge (FRs).

In the simulation, an "episode" corresponded to a single attempt by an agent to publish a paper, whether successfully or not. There was a maximum of 10 agents in the system at any given time. Agents were pre-trained (for 10 episodes) before entering the system. Reflecting a "publish or perish" academic environment, agents were evaluated periodically (every five episodes) based on their publication record (i.e., success rate). If an agent fell below a minimum expected standard (40% success), the agent was removed from the academic world. If the agent passed all the evaluations, it retired upon reaching the maximum allowable age (60 episodes). Whenever an agent retired or was removed, a new agent took its place. This was somewhat analogous to the real-life academic world.

Below, I sketch some technical details to substantiate some ideas outlined above (which, however, may be skipped by any reader uninterested in technical details). In the simulation, a paper was a multidimensional vector. Without loss of generality, the vector had 12 dimensions, with each dimension having a value that was a real number between 0 and 16.

"Pulling" was accomplished by moving the original point in the idea space toward the second point (the "pull" point), by a certain fraction of the distance between them. Gilbert's formula was adopted:

$$d_i := d_i + (d'_i - d_i)(1 - m)/2$$

where i ranged over all dimensions, d_i was the value of dimension i of the original focal point, d'_i was the value of the same dimension of the "pull" point, and $m \in [0, 1]$ was a constant that was incremented by 0.1 after each "pull" (to gradually reduce the amount of pulling).

Agents were restricted to two "pull" ideas per focal idea. After idea selection and "pulling," an agent performed limited local search. The search was done within a radius of two from the modified idea, using the hill-climbing algorithm (which had been shown to capture human reasoning in some cases).

To represent communal evaluation functions, five polynomial functions (randomly generated) were used (with a maximum degree of three). Each agent had its own weights (randomly generated) for these functions that they used to compute a weighted average of the five functions. While agents knew the functions, they did not know the "true" weights (i.e., the weights used by a journal in determining acceptance or rejection of an article) and therefore their learning consisted, in part, of overcoming their initial bias in this respect.

Paper acceptance was determined by (1) being above a threshold of 0.5 when evaluated using the five evaluation functions with predetermined weights, and (2) having a minimum distance of 1 between the new paper and any existing papers. The former requirement provided a minimum standard for paper quality, forcing the agents to learn and adapt. The latter requirement represented a criterion for paper originality.

7.5.3. Simulation Results

The results of the simulation were as shown in Tables 7.6–7.7. They were compared with the actual data from *Chemical Abstracts* and *Econometrica*, and the results obtained from previous simulations by Simon (1957) and Gilbert (1997). The Clarion results were the averages of 300 runs, ensuring the representativeness of the results.

Table 7.6. Number of authors contributing to *Chemical Abstracts*. The figures in the table indicate number of authors contributing to the journal, by number of papers each author has published.

# of Papers	Actual	Simon's	Gilbert's	Clarion
1	3,991	4,050	4,066	3,803
2	1,059	1,160	1,175	1,228
3	493	522	526	637
4	287	288	302	436
5	184	179	176	245
6	131	120	122	200
7	113	86	93	154
8	85	64	63	163
9	64	49	50	55
10	65	38	45	18
>11	419	335	273	145

Table 7.7. Number of authors contributing to *Econometrica*. The figures in the table indicate number of authors contributing to the journal, by number of papers each author has published.

# of Papers	Actual	Simon's	Gilbert's	Clarion
1	436	453	458	418
2	107	119	120	135
3	61	51	51	70
4	40	27	27	48
5	14	16	17	27
6	23	11	9	22
7	6	7	7	17
8	11	5	6	18
9	1	4	4	6
10	0	3	2	2
>11	22	25	18	16

First, note that the Clarion simulation results for the two journals were fit to the power curve $f(i) = a/i^k$, resulting in an excellent match, similar to the prior simulations.

However, in the present simulation, the number of papers published by an author reflected the cognitive ability of the author, as opposed to being based on auxiliary assumptions (such as those made by Gilbert, 1997). This explains, in part, the slightly greater divergence of the present results from the human data: whereas Gilbert's simulation consisted of equations selected to match the human data, the Clarion approach relied on more detailed, lower-level mechanisms—namely, a cognitive architecture that was generic rather than task-specific. The result of the Clarion-based simulation was therefore emergent, not a result of specific and direct attempts to match the human data. This simulation put more distance between mechanisms and outcomes, which made it harder to obtain a match with the human data. Thus, the fact that a good match with the human data was found showed the potential of the Clarion-based approach.

As in the case of organizational decision making discussed earlier, there is more to cognitive social simulation than merely replicating and validating previous results. With Clarion, one can vary parameters that correspond to specific cognitive factors, and observe the effects on performance.

A number of cognitive parameters were varied. In the previous simulation of organizational performance, parameters were varied in

a factorial design, and this yielded a complex pattern of interactions among different variables. Here, due to the greater complexity of the task, each parameter was varied relative to a single baseline condition, rather than being compared to all combinations of all values of all parameters. A baseline condition was used, which consisted of the parameter values used in the simulation above. Each of the other conditions was identical to the baseline condition except for one parameter that was assigned a different value.

The results were as follows. As one would expect, cognitive parameters of individual agents were important in determining the rate of scientific progress. By varying these parameters, one could come up with scientific communities that produced different numbers of papers.

Apart from this aggregate measure of scientific productivity, it would also be interesting to see if the patterns of individual contribution observed earlier would be preserved under different cognitive parameter settings. In particular, it would be interesting to see if the power curve was obtained under different cognitive parameter settings. As seen in Figures 7.22–7.23, different settings of density and generalization threshold led to larger or smaller numbers of papers in aggregate, but they did not fundamentally change the curve, which followed an inverse power distribution. Similar results were obtained for other (though not all) ranges of cognitive parameters.

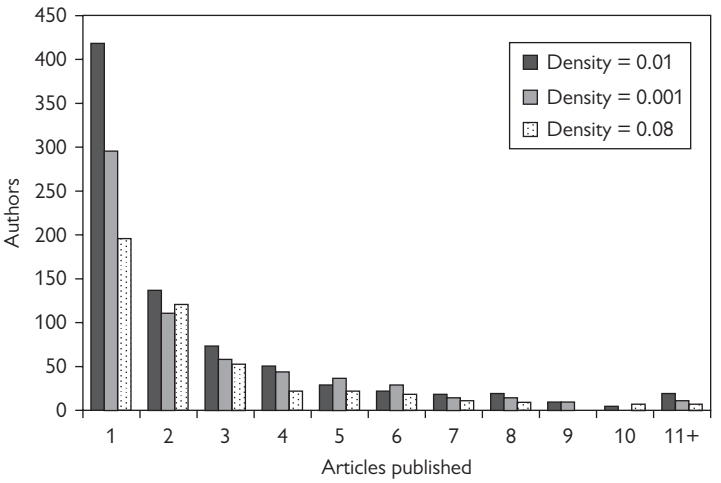

Figure 7.22. Numbers of authors contributing different numbers of articles, given different settings of density.

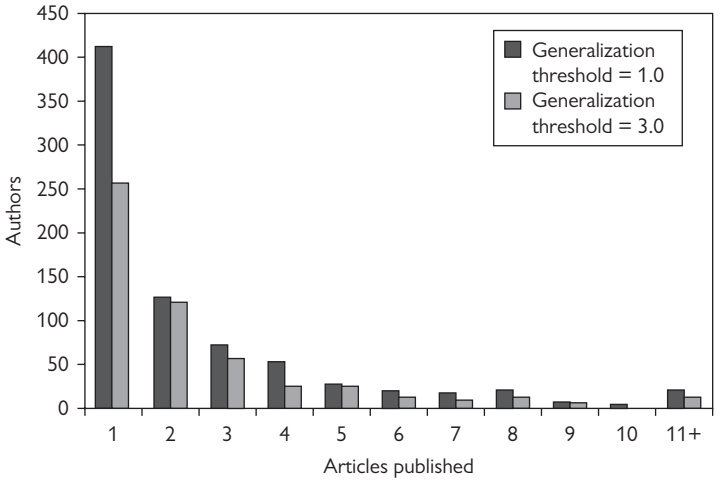

Figure 7.23. Numbers of authors contributing different numbers of articles, given different settings of generalization threshold.

This result, which I termed "cognitive-social invariance," was important. It showed that some regularities that characterized societies were to some extent invariant with respect to individual cognition (within a reasonable range, of course). This reduced the likelihood that the patterns observed were a by-product of a particular set of cognitive parameters. As a comparison, in the simulation of organizational decision making earlier, it was shown that some patterns were indeed directly related to the settings of cognitive parameters.

In addition, effects of individual cognitive parameters on collective performance were observed, similar to what was discussed in the organizational or the tribal simulations earlier. I will not repeat such an analysis here; the interested reader should see Naveh and Sun (2006).

However, for an instance of a new finding, look into the "temperature" (degree of randomness) of an agent's decision making, which modulated an agent's exploration of the idea space (see Chapter 3). As shown in Figure 7.24, agents were at their most prolific under a moderately high temperature setting—that is, when they showed a willingness to experiment (to pursue new leads) while still being mostly guided by their experiences. This observation is consistent with the notion of serendipity in scientific discovery. Many major scientific discoveries have been serendipitous in that they have been the result of seemingly unrelated

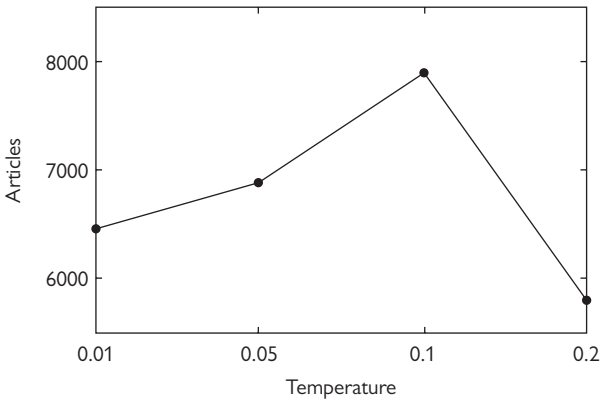

Figure 7.24. The effect of temperature on the total number of articles published.

investigations. Although such discoveries were often attributed to good fortune, some argue that serendipity is partly a cognitive faculty that can be nurtured and developed. The simulation captured this notion of serendipity through a modest degree of randomness in decision making.

7.5.4. Discussion

This study provided a corroboration of the earlier simulations of the same phenomenon, by showing some of their results to be independent of cognitive processes to some extent. Therefore, while sometimes cognitive details clearly cannot be abstracted away, other times they can. Along the way, we may discover important cognitive-social invariance.

Apart from corroboration and validation, cognitive-psychological realism in social simulation may lead to better representations of target phenomena. For instance, a possible way of capturing the role of serendipity in science was identified, as a researcher's willingness to explore apparently less than optimal ways. The ability to represent such aspects in terms integral to the cognitive architecture, rather than through auxiliary assumptions (e.g., by adding a "randomizing" function to the idea selection process in Gilbert's simulation), is an advantage of cognitively-psychologically realistic simulation.

Another advantage of cognitively-psychologically realistic simulation lies in the exploration of the relative roles that individual cognitive parameters play in the emergence of large-scale social phenomena.

Parameters of Clarion that corresponded to aspects of cognition were varied and their effects on outcomes were tested. As mentioned before, with Clarion, parameters being altered were presumably fundamental aspects of cognition, and thus observed differences in performance were more likely to stem from real differences in individual cognition.

Cognitively-psychologically realistic simulation can help to establish the constancy of some observed phenomena. For instance, the same power curve was observed under different cognitive parameter settings, even when overall scientific productivity varied. Such results lent support to theories of cognitive-social invariance. Alternatively, they might also suggest boundary conditions under which such phenomena began to break down. As demonstrated in the previous simulations of organizations and tribes, patterns of collective performance might change as a consequence of changes of individual cognitive processes.

7.6. General Discussion

Beyond what has been described so far, there have been various other efforts at exploring the grounding of social phenomena in cognitive-psychological mechanisms and processes. Some of the efforts were computationally motivated as has been discussed above (see also Sun, 2006). Some other efforts were more empirical or theoretical in nature (see Sun, 2012b).

For instance, war and revenge in tribal society have been simulated on the basis of Clarion, in which more realistic modeling of both social and cognitive-psychological processes is involved. Emergence of norms and moral codes has also been tackled. Various other projects are also under way.

7.6.1. Theoretical Issues in Cognitive Social Simulation

The integration of the social sciences and cognitive science may provide the social sciences with new approaches and novel frameworks, besides providing cognitive science with new data and new problems to address (Sun, 2006, 2012b). Understanding theoretical issues involved in the interaction of cognition-psychology and sociality requires, at least in part, computational modeling and simulation,

because of the complexity of such an undertaking, and also because of the expressive power of computational models. Unlike mathematical modeling, computational modeling is not limited by traditional mathematics. Hence it enjoys greater expressive power. Yet, compared with verbal theories, it is more precise. See Sun (2006, 2009b) for more discussions of this point.

In general, a mechanistic (e.g., computational) explanation of a phenomenon means specifying a mechanism that is capable of producing the phenomenon. It consists of describing the structures and processes related to a mechanism and showing how the mechanism leads to the phenomenon (Sun, 2009b). It is a description often in terms of lower-level entities and processes. In this regard, a cognitive architecture may lead to explanations of social phenomena based largely on underlying psychological factors, relying on entities and processes at a lower level (i.e., the psychological, as opposed to the sociological, level).

Although some have argued that lower-level explanations are uninformative and provide unnecessary details, lower-level mechanisms and processes are often an important part of the constitution of social processes. By using cognitively-psychologically realistic agents in social simulation, one might be able to provide explanations of (at least some) observed social phenomena based on individual psychological processes. This often allows one to do away with some assumptions that are not cognitively-psychologically grounded. Often, in social simulation, ad hoc assumptions were made because they were needed for generating simulation results that matched observed data. Assumptions should instead be made at a lower level. This approach puts more distance between assumptions and outcomes, and thereby provides deeper explanations (as argued more extensively in Sun, 2006).

Although nearly any higher-level process may be described in terms of lower-level entities, the actual higher-level processes that occur may depend on a particular combination of conditions. There is often no a priori way of determining, based solely on lower-level entities, which of the higher-level processes will actually occur. Thus, social processes may in this sense be "emergent." Based on that, one might argue that the approach described in this chapter is needlessly reductionist: causal relationships at the higher level may be a product of causal relationships at the lower level; nevertheless, it is possible to describe causal relationships at the higher level without referring to

relationships at the lower level. Why, then, is cognitive-psychological realism in social simulation needed? One answer is that a deep and convincing theory must be capable, in principle at least, of mapping a higher-level theory to lower-level theories. In this case, it means possibly mapping sociocultural phenomena to cognitive-psychological processes. The ability to accurately model higher-level phenomena through a higher-level description is a necessary, but not sufficient, condition for a deep theory. It is preferable to bridge higher and lower levels in theorizing. Besides, there are also a number of other (more practical) advantages as touched upon earlier (e.g., testing the effects of individual cognitive parameters, discovering cognitive-social interactions or invariance, and so on).

Such work may not only shed new theoretical light on social processes and phenomena, but may also connect individual-level cognitive-psychological analysis and collective-level social analysis (Sun, Coward, & Zenzen, 2005; Sawyer, 2003; Alexander et al., 1987) and thus build deeper theories linking the macro with the micro. In so doing, it also allows greater distance between assumptions and outcomes and therefore deeper explanations (as argued earlier). In addition, cognitive-psychological realism might also lead to theories of greater explanatory accuracy and greater predictive power (as shown earlier).

However, there are several considerations that may limit the applicability of a cognitive-psychological approach to social simulation. One is the consideration of complexity, which, for one thing, can make it difficult to interpret results in terms of their precise contributing factors. It is never an easy task to distinguish between simulation results that genuinely shed light on an issue and ones that are mere artifacts. More complex models may exacerbate this problem. One must be aware of "Bonini's Paradox" (Fum et al., 2007). That is, as models become more and more realistic, they may become more and more complex, and eventually they may be so complex that they are as difficult to understand, explain, and communicate as the system being modeled and simulated. This point seems to argue against cognitively-psychologically realistic social simulation. Furthermore, complexity also leads to high computational costs and hence raises the issue of scalability. In addition, there is the issue of proper choice of a theoretical framework in relation to cognition-psychology, which may hinge on particular conceptions or interpretations of social as well as psychological phenomena.

7.6.2. Challenges

The fundamental challenge in further developing (psychologically realistic) cognitive social simulation as described in this chapter is how to seamlessly integrate a detailed social simulation with a detailed cognitive architecture (serving as building blocks). One specific challenge in this regard is how to possibly enhance cognitive architectures for the purpose of accounting for sociality in individuals. For example, what are essential psychological features that should be taken into consideration in models of multi-agent interaction? In particular, what is needed in relation to social cognitive capabilities in individuals? What sociocultural representations are likely to reside in the individual mind (e.g., "norms," "obligations," "rights," etc.)? In what forms? And so on.

Consider the incorporation of various social cognitive mechanisms and processes. For one thing, the distinction of implicit versus explicit social cognition is important for explaining a variety of social psychological data, for example, Lambert et al. (2003), Norton, Vandello, and Darley (2004), Chen et al. (1996), Wegener and Petty (2001), and so on. There are also a variety of other social cognitive (social psychological) mechanisms: for example, moral judgment, person perception (impression formation), social categorization, and so on. With these social cognitive mechanisms, one may investigate the effects of these mechanisms on social processes (e.g., the effects of social cognitive attributes on organizational performance). Conversely, one may also pursue explanations of social phenomena from the underlying social cognitive-psychological mechanisms (and other cognitive-psychological mechanisms), for example, explaining Schelling's (1971) segregation model by grounding it in real human motives and psychologically realistic models of human social perception and social identity.

To meet these challenges above and to more fully reap the benefits of cognitive social simulation, a number of directions need to be explored and a number of different methods need to be utilized in synchrony, including all of the following:

- Incorporating in models of individuals learning, reasoning, decision making, problem solving, metacognition, motivation, emotion, personality, morality, and so on (i.e., going beyond the narrow definition of cognition).

- Developing social process models, especially large-scale ones, on top of detailed cognitive-psychological models of individuals
- Exploring empirical work (both laboratory experiments and field studies), for the sake of detailed cognitive-psychological modeling and large-scale social modeling and simulation on the basis of detailed cognitive-psychological models
- Exploring the relationship of social and individual processes, on a large scale possibly, on the basis of the synthesis of the models mentioned above
- Addressing a wide variety of issues based on the aforementioned explorations, for example, cultures, institutions, moral codes and beliefs, normative standards of reasoning, and so on.

7.6.3. Concluding Remarks

This chapter addresses cognitive social simulation (Sun, 2006), at the intersection of cognitive science (cognitive modeling in particular) and the social sciences (social simulation in particular). By combining cognitive models and social simulation models, cognitive social simulation is poised to address the interaction of the cognitive-psychological and the social, in addition to understanding cognitive-psychological and social phenomena separately. Cognitive social simulation may even find practical applications (e.g., as documented in Sun, 2006).

In particular, this chapter explored the integration of two modeling approaches, with the use of a cognitive architecture that includes cognitive, motivational, and other psychological processes. Such integration could lead to greater explanatory depth, through exploring the role of individual psychological processes in collective social phenomena. For instance, interactions among cognitive, motivational, environmental, and social factors have been shown. These interactions support the claim that social processes and phenomena may be related to individual cognitive, motivational, and other psychological processes—a form of the micro-macro link.

The results from the Clarion-based cognitive social simulation have been encouraging. They yielded findings consistent with the psychological and sociological literatures. Moreover, they also led to some novel insights into sociocultural processes, cognitive-psychological processes, and their interactions. Thus, the Clarion cognitive architecture

has shown some promise of serving as a foundation for cognitive social simulation.

Note that this is not to say that social processes are fully determined by individual cognition-psychology, but rather that they are (at least) manifested through individual cognition-psychology and thus there is a great deal to be gained from studying the social and the cognitive-psychological together and from exploring social processes and phenomena from a cognitive-psychological viewpoint (Sun, 2006; Sun, 2012b). Although some cognitive-psychological details may ultimately prove to be irrelevant, they often cannot be determined a priori. Thus, cognitive social simulation may be useful.

8

Some Important Questions and Their Short Answers

This chapter will be devoted to addressing a number of commonly raised or otherwise particularly pertinent questions about Clarion. These questions are roughly divided into the following generic categories: theoretical issues, computational issues, and biological connections.

8.1. Theoretical Questions

Overall, what is Clarion about? Is the goal of Clarion to draw inspiration from natural systems in order to build artificial systems or is it to explain natural phenomena?

There are significant differences between drawing inspiration from natural systems (for the sake of building comparable artificial systems) versus explaining and understanding natural phenomena (including mechanistic, process-based explanations of psychological phenomena with computational modeling). Clarion clearly belongs to the latter category. The former is engineering (or reverse engineering of nature), while the latter is not (for the most part at least). The goal of Clarion includes providing mechanistic, process-based explanations of psychological (and

maybe other) phenomena—that is, computational psychology (e.g., Sun, 2008) and beyond, as has been emphasized thus far in the preceding chapters.

Then, does this cognitive architecture constitute a cognitive-psychological theory?

A computational cognitive-psychological model is a formal description of some cognitive-psychological phenomena. The language of a computational model is, by itself, a distinct symbol system for formulating and expressing a formal description. Therefore, a computational model can constitute a theory by itself. There has long been a view that (almost) every computational model provides a theory of the phenomena that it models (e.g., Newell, 1990; Sun, 2009b). This position has been advocated or taken for granted by many in the cognitive science community.

This may be due to the fact that no verbal-conceptual theory completely specifies details of mechanisms involved in a phenomenon, let alone dynamic processes that may emerge from the mechanisms. Thus, computational models are necessary to describe these complex aspects, for example, in order to produce a runnable simulation, which at the same time also provides a more precise and more detailed theory than corresponding verbal-conceptual theories. The language of computational modeling is, in essence, just another language for presenting a theory, albeit at a more detailed (and somewhat less intelligible) level.

Equation-based mathematical theories are indeed often rigorous. However, their expressive power is often more limited. Computational models can more readily express contents that equations cannot express easily. Equations can usually be incorporated into computational models, while the reverse may not be true. Besides, equations may not clearly express dynamic processes that emerge from entities and their relations, even when equations can specify these entities and relations rigorously.

Like verbal-conceptual theories or mathematical theories, computational models can be used to generate predictions. In fact, they often can generate more detailed and more precise predictions that can potentially be more precisely tested.

But how does one justify or validate all these algorithmic details inevitably present in a computational model? It is worth noting that there is a well-argued position in philosophy of science, constructive empiricism (e.g., van Fraassen, 1980), which argues (roughly) that not all details

of a scientific theory need to be strictly derived from empirical data, which is impossible anyway. Only the empirically observable parts of the theory need to be mapped to empirical data. Some of the algorithmic details in a computational model are not empirically observable, and therefore validation of these details is not possible and not necessary, according to constructive empiricism.

Constructive empiricism may make a more sensible philosophical foundation for computational psychology—that is, computational cognitive modeling—than naive empiricist positions. See van Fraassen (1980) for a detailed account of this position. See also Sun (2009b) for its relevance to computational cognitive modeling.

In that case, how do computational theories, such as Clarion, relate to mathematical theories on the one hand and verbal-conceptual theories on the other?

The difference between a computational theory and a mathematical theory or between a computational theory and a verbal-conceptual theory is often a matter of descriptive medium, descriptive complexity, and descriptive style.

Mathematical equations and computational models are both instances of formal models. In this sense, they are not fundamentally different. For example, issues of validation, matching, and prediction are common to all formal models, whether mathematical or computational.

But they are different in some other ways. One difference is that of the languages on which they are based: mathematical equations versus computational algorithms (note that algorithms and program code may be viewed as being equivalent here). Another difference is that, due to the difference in language, mathematical models are often simpler to specify (in terms of length of description), while computational models often require longer descriptions to express. Yet another difference is that mathematical models are often in a closed form (i.e., with the relationship between input and output variables apparent), while computational models are often in an open form.

The notion of descriptive complexity may be used for comparing various types of theories. Depending on domains, theories vary in terms of explanatory succinctness. In some cases, a small and rigorous set of equations is able to express the regularity of a domain to a sufficient extent, approximating it with an acceptable level of accuracy. For example, in physics,

Newtonian classical mechanics is such a theory. However, in some other domains, no succinct set of equations is found that can express domain regularities to a satisfactory extent. In such cases, a more complex form of theory is necessary. Computational models are a possible type of complex theory for these domains. Understanding the human mind is one domain in which no simpler form of theory is viable, at least up to this point.[1]

In constructing a comprehensive computational model such as a cognitive architecture, one may sometimes take specific contents of verbal-conceptual theories and formalize them into algorithms. Or one may take equations from mathematical theories and fit them into the model. Moreover, a computational model may combine various theories, or various aspects of different theories, regardless of whether they are verbal-conceptual, mathematical, or computational. It therefore integrates different elements as "subsystems," "modules," "components," or "mechanisms" (Sun, 2009b). Thus, fragments of theories may cooperate and compete with each other in a more comprehensive theory/model, and they may also cooperate and compete with each other in explaining simulation results. One can clearly see this aspect at work in the descriptions of Clarion up to this point.

Finally, different types of theories may have different roles to play. Computational theories often focus more on questions of "how," while other types often focus more on "what" or "why." That is, computational theories are often more process-based and more mechanistic than other types. From all of the above, different types of theories can be complementary to each other (Sun, 2009b; see also the answer to the previous question).

Explaining natural phenomena is difficult because these phenomena may be ambiguous, there may be many ways of explaining them, and so on. So how can one possibly tackle them with computational modeling?

A field often progresses through looking for many possible ways to shed light on relevant phenomena and issues and then hopefully converging on

1. *Kolmogorov complexity* measures the minimum length of the description of an algorithm (Li and Vitanyi, 1997). It may be a foundation upon which one may compare different theories. A key difference between different types of theories (verbal-conceptual, mathematical, or computational) may be captured in terms of the description length (Kolmogorov complexity) of a theory and, by implication and extension, the numbers of entities and relationships required by the theory (Sun, Coward, & Zenzen, 2005; Sun, 2009b).

the best theory. A field is often focused on disambiguating ambiguous or otherwise difficult and complex phenomena. In this effort, computational modeling is but one possible approach—a highly pertinent approach, but it needs to be supplemented by other approaches, especially empirical work in various ways.

Scientific work often consists of a continuous search for explanations of phenomena and improvements of adequacy and quality of existing explanations (e.g., in terms of empirical coverage, explanatory succinctness, and so on). Computational models may serve as candidate theories (albeit in a computational form) in this search as previously argued (see Sun, 2009 b). Often, definitive "proof" cannot be found (at least not easily), for computational theories or theories of other forms alike. But gradually, evidence accumulates and hopefully converges, and eventually one may arrive at a reasonably solid theory of a set of related phenomena. But one is always ready to probe deeper or more widely, and in the process, ready to improve or revise existing theories, computational or otherwise (Kuhn, 1970).

Therefore, there is really no significant difference between computational theories and other forms of theories in this regard. They all have to deal with ambiguity and other difficulties inevitably encountered. The answer to the earlier question regarding computational models constituting theories (Sun, 2009b) is pertinent here also.

As a theory, can Clarion be disproved?

I like Kuhn's (1970) and Lakatos' (1970) ideas regarding scientific work. Consequently, I do not believe in the simplistic notion of proving or disproving ("falsification") broad theories.

According to Kuhn (1970), on the assumption that a current theory is correct, observations are collected and fitted within the current theory. In the process, unexpected phenomena may be uncovered and may lead to refinement or revision of the theory. Specifically, in cognitive modeling with cognitive architectures, architectural assumptions and other commitments constitute an initial theory, which undergoes testing and validation through matching and explaining data. Revision and refinement are carried out when inconsistencies and incorrect predictions are discovered or when the model is incapable of predicting something important. However, when given a sufficiently high degree of mismatch between the data and the current model (i.e., when revision and refinement are no longer able to accommodate problems that arise), a crisis

may develop, which leads to a new "paradigm": that is, a new cognitive architecture or even a new approach to building cognitive architectures.

According to Lakatos (1970), scientific growth should be assessed in terms of progressive or degenerating research programs. What is important is coming up with conjectures that have more empirical content than their predecessors. A research program is degenerating only if it does not generate new hypotheses that have more empirical content. So as long as a research program (such as a cognitive architecture) is making progress toward more and more empirical coverage, it is likely to be on the right track (Cooper, 2007).

Can a computational-mathematical model be replaced by its corresponding verbal explanations, which inevitably accompany a model?

It is true that a computational-mathematical model is often accompanied by verbal explanations (such as in this work), and one may often pay more attention to the verbal explanations rather than the model itself.

However, translating a computational-mathematical model into a verbal-conceptual theory precisely and completely is often difficult, if not impossible. The definitions, assumptions, mechanisms, processes, and parameters of a computational-mathematical model have to be explained using natural language. Due to reliance on natural language, this version may be less precise, and also open potentially to contradictory interpretations due to ambiguity or imprecision of natural language (in contrast to the precision of computational-mathematical models). Computational-mathematical models have to be defined with rigor and cannot include any mathematical or computational ambiguity, although they may often need to be interpreted at a higher level of abstraction. Any high-level conclusion drawn from modeling and simulation studies will have to be treated with such caveats in mind (Sun, 2009b).

Considering the reasons above, it is not likely that computational-mathematical models can be replaced by verbal explanations (or verbal-conceptual theories).

How does Clarion account for individual differences?

As discussed before (e.g., Section 6.4), various forms of individual differences may be translated into different parameters within Clarion.

In general, individual differences may result from many different aspects of an organism, psychological, biological, social, and so on. Within the Clarion framework, the following factors, among others, may be hypothesized:

- differences in drive activation
- differences in goal setting
- differences in action decision making
- differences in the capacity for implicit learning (at the bottom level)
- differences in the capacity for explicit learning (at the top level, including bottom-up learning)
- differences in the capability of, and the inclination for, explicit reasoning (including working memory capacities, logical reasoning capabilities, and so on; e.g., at the top level of the NACS)
- differences in the capability of, and the inclination for, intuitive thinking (e.g., at the bottom level of the NACS)
- differences in sensory-motor processes

and so on.

In Section 6.4, I discussed in detail how some fundamental individual differences (e.g., personality traits) might be accounted for (in part) by drive-related parameters. Specifically, within the drive strength equation, *deficit*, *gain*, *stimulus*, and other parameters may be used (in part) to account for various effects of the motivation-cognition interaction and some personality traits.

Note that individual differences are not necessarily unalterable in many of the aforementioned aspects. Of course, some aspects are more entrenched than others (e.g., primary drives may be the most stable and the least transient).

What distinguishes humans from other primates according to the Clarion framework?

The difference should be more quantitative than qualitative. The scale of "intelligence" should be (more or less) continuous. For this reason, there should not be radical differences in accounts of psychological processes of animals and humans. They should form a sort of continuum, going from simple physical reactions all the way to complex

psychological dynamics. At each stop, a few more capabilities are added (Braitenberg, 1984). Eventually, very sophisticated psychological beings arise as a result.

Of course, I am not implying that there is a strict hierarchy in this regard. Nor do I believe that there is no discontinuity of any kind. There have been arguments that cognitive discontinuity exists between humans and their close relatives in terms of enhanced hand-eye coordination, causal reasoning, executive control, social learning, social intelligence, and language use (e.g., Vaesen, 2012). However, such differences are relatively minor and quantitative when viewed in the bigger scheme of things.

From this perspective, a number of distinguishing psychological features between humans and other animals may be hypothesized within the Clarion framework: (1) in humans, compared with other primates, there are more extensive explicit representations in various subsystems; (2) in humans, there are more extensive explicit reasoning capabilities, especially in the non-action-centered subsystem; (3) in humans, there are better developed metacognitive capabilities (especially in the metacognitive subsystem); (4) in humans, there may be more complex and more sophisticated motivational processes (in the motivational subsystem), taking into consideration more complex and less rigid social situations that humans are likely to encounter. However, do note that these differences, as indicated earlier, are generally more quantitative than qualitative.

Why does Clarion not adopt the framework of BDI?

The BDI framework (e.g., Rao & Georgeff, 1991) may be useful for those working in AI; that is, it may be a useful tool for building intelligent systems for practical applications. But there is no demonstrated psychological validity at a detailed level. Nor does it shed any significant new light on human cognition-psychology. The BDI framework is, more or less, folk psychology.

For one thing, the BDI framework typically does not capture fine-grained details (mechanisms and processes) of human cognition, motivation, and their relations to action. Perhaps as a result of that, it has had no significant impact on cognitive science or psychology. There has been a long history of psychology of motivation, which provides in-depth explorations and theories of human motivation and its relation to cognition and action (Weiner, 1992).

However, the following rough correspondences may be identified between the BDI framework and the Clarion framework (which is, of course, more detailed and more psychologically oriented):

1. desires ≈ drives
2. intentions ≈ goals
3. beliefs ≈ knowledge in the ACS and the NACS (both implicit and explicit knowledge in the two subsystems).

That is, in Clarion, drives lead to goals, which in turn lead to actions on the basis of existing knowledge, in rough correspondence with BDI.

How does Clarion relate to other dual-process theories (two-system views)?

There have been a number of dual-process theories or two-system views. The distinction between System 1 and System 2 (or between "intuitive" and "reflective" thinking) has been one of the most important distinctions to emerge recently in cognitive science. It seems to have captured the current popular imagination.

However, although the distinction itself is evidently important, these terms used to describe it in the literature have been somewhat ambiguous. Not much finer-grained analysis has been carried out, especially not in a precise, mechanistic, process-based way. In Clarion, I adopted the terms of implicit and explicit processes and presented a more nuanced view. The use of the Clarion cognitive architecture led to formulating a more fine-grained interpretation.

In order to see this, some historical background should be briefly reviewed here. There have been some early ideas concerning duality of the mind that dated back before the inception of cognitive science. For instance, Martin Heidegger's distinction—the preontological versus the ontological—is an abstract version of such a duality (Heidegger, 1927). His view was roughly that because the essential way of being is existence in the world, an individual always embodies an understanding of its being through such existence. This embodied understanding consists of skills, reactions, and know-hows, without an explicit "ontology", and is thus preontological (implicit). On that basis, an individual may also achieve an explicit (ontological) understanding, especially through making the implicit understanding explicit, that is, through turning preontological understanding into ontological understanding (Heidegger, 1927; Dreyfus,

1992). This progression from the concrete to the abstract is a fundamental part of Clarion.

I should also mention William James's distinction between "empirical thinking" and "true reasoning." According to James (1890), empirical thinking is associative, made up of sequences of "images" that are suggested by one another. It is "reproductive" because it replicates past experience in some way instead of producing new ideas. Empirical thinking relies on overall comparisons and similarity among various concrete situations, and therefore may lose sight of critical information. On the other hand, "true reasoning" is achieved by abstracting attributes. It is "productive" because it is capable of producing novel ideas through abstraction. "True reasoning" breaks up the direct link between thought and action, and provides means for reasoning about consequences of an action without actually performing it. Some of these characteristics (such as overall similarity versus abstraction) are evident in Clarion.

There are a few theories or arguments for dual processes (two systems) from within cognitive science. In particular, Sun (1994, 1995), Sloman (1996), Kahneman (2003), and Evans (2003) are relevant here. One view was described in Sun (1994), in which the two systems were characterized as follows:

> It is assumed in this work that cognitive processes are carried out in two distinct "levels" with qualitatively different mechanisms. Each level encodes a fairly complete set of knowledge for its processing, and the coverage of the two sets of knowledge encoded by the two levels overlaps substantially (Sun, 1994).

That is, the two "levels" (i.e., two systems, two modules, or two components) encode somewhat similar or overlapping content. But they encode their content in different ways: symbolic versus subsymbolic representation were used, respectively. Therefore, different processes and mechanisms are involved at these two levels. As a result, one level contains explicit processes and the other implicit processes. It was hypothesized in Sun (1994) that these two different levels can potentially work together synergistically, complementing and supplementing each other, which is, at least in part, the reason why there are these two levels (evolutionarily speaking).

A more recent dual-process theory was proposed by Kahneman (2003). The gist of his ideas was as follows: "intuition" (or System 1) is typically based on associative reasoning, fast and automatic, involving

strong emotional bonds, based on formed habits, not conscious, and difficult to change or manipulate. "Reasoning" (or System 2) is slower, effortful, and subject to conscious judgment and control. Evans (2003) espoused a similar view. According to him, System 1 is "rapid, parallel and automatic in nature: only their final product is posted in consciousness," and its learning is "domain-specific." System 2 is "slow and sequential in nature and makes use of the central working memory system," and it "permits abstract hypothetical thinking that cannot be achieved by System 1." Moreover, in terms of the relationship between the two systems, he argued for a "default-interventionist" view. According to him, System 1 is the default system that operates all the time, while System 2 may intervene occasionally when feasible and called for.

According to Clarion, however, some such claims may be painting a picture in overly broad strokes, when examined against the empirical literature. For one thing, intuition can be very slow (e.g., as demonstrated by Helie & Sun, 2010; Bowers et al., 1990). For another, implicit processes can be subject to conscious control and manipulation; that is, it may not be entirely "automatic" (Berry, 1991; Curran & Keele, 1993; Stadler, 1995). Furthermore, implicit decisions can be subject to conscious "judgment" (Libet, 1985; Gathercole, 2003). In terms of the relationship between the two systems, implicit and explicit processes may be parallel and mutually interactive in more complex ways than what was described by the default-interventionist view (Sun, 2002). There are many such detailed issues and questions that one may need to explore with regard to the characteristics of the two systems, which Clarion can help to clarify (see, e.g., Sun, 2014).

How does Clarion relate, in particular, to the division of fast and slow processes, which is a crucial feature in some dual-process theories?

Instead of simply claiming, as in some existing dual-process theories, that implicit processes are fast and explicit processes are slow, one needs to take a more detailed look. To come up with more nuanced characterization, it is important to ask some pertinent questions first. Specifically, in relation to the relative speeds of implicit and explicit processes, the following questions should be asked with regard to each type of process:

- How deep is the processing (in terms of precision, certainty, and so on)?

- How broad is the processing (e.g., how much information is involved)?
- How incomplete, inconsistent, or uncertain is the information available?
- How many processing cycles are needed considering the factors above?

The twin dichotomies in Clarion, procedural versus declarative and implicit versus explicit, have implications for identifying slow versus fast processes. In accordance with the Clarion framework, instead of simply assuming the seemingly obvious, one may question conventional wisdom on a number of fronts in this regard:

- In terms of the division between procedural and declarative processes, can fast procedural versus slow declarative processes be posited?
- In terms of the division between implicit and explicit procedural processes, can fast implicit versus slow explicit processes be posited?
- In terms of the division between implicit and explicit declarative processes, can fast implicit versus slow explicit processes be likewise posited?
- What about relative speeds if we consider the four-way division together?

And so on.

The conjectures implied by the questions above may be true to some extent but not exactly accurate (Sun, 2014). In this regard, one may view existing models and simulations of these types of processes as a form of theoretical interpretation concerning their time courses. In that case, there are the following potential answers to these questions according to Clarion:

- Fast procedural versus slow declarative processes: This hypothesized speed difference is generally true if we examine many existing models and simulations (e.g., Sun, Zhang, & Mathews, 2009; Sun & Zhang, 2006; see also Anderson & Lebiere, 1998).
- Fast implicit versus slow explicit procedural processes: This hypothesized speed difference is, again, generally true,

using theoretical interpretations from simulations (e.g., Sun et al., 2005).

- Fast implicit versus slow explicit declarative processes: This, however, is generally not true. Implicit declarative processes (intuition) often take a long time, compared with explicit declarative processes. See, for example, Helie and Sun (2010) and Bowers et al. (1990).

Within the Clarion framework, many empirical and simulation studies have been conducted that shed light on these issues and substantiate the points made above. See, for example, Sun et al. (2009), Sun and Zhang (2006), Helie and Sun (2010), Sun and Mathews (2005), Sun (2012), and so on. The whole picture may not be as simple as conventional wisdom assumes. One needs to be careful in making sweeping generalizations—different types of processes need to be characterized in a more fine-grained fashion. Characteristics of different processes may also vary in relation to contextual factors, such as task demands.

Given the above, how does Clarion characterize implicit and explicit processes?

Based on empirical data and computational modeling, I can enumerate some main characteristics associated with the two "levels" of Clarion as in Table 8.1.

As mentioned earlier in Chapter 3, there are different types of differences between the two levels in Clarion (some of which have been listed in Table 8.1): (1) phenomenological differences (i.e., the distinctions between the conscious and the unconscious in a subjective sense); (2) psychological differences (the distinction as revealed by psychological experiments, including, for example, implicit versus explicit learning, implicit versus explicit memory, and other related psychological constructs and generalizations, as indicated in the table); (3) implementation-related differences. Among them, the implementation-related differences (in particular, the representational difference—symbolic-localist versus distributed representation) account for the phenomenological and the psychological differences (Sun, 1999; Sun, 2002; Sun, 2012). So in this sense, the implementation-related differences constitute the basis for accounting for dual processes (two "levels" or two "systems").

Table 8.1. Comparisons of the two levels of Clarion.

	Bottom level	Top level
Phenomenological differences	Unconscious	conscious potentially conscious
Psychological phenomena	implicit learning implicit memory implicit knowledge automatic processing intuition	explicit learning explicit memory explicit knowledge controlled processing explicit reasoning
Sources of knowledge	trial and error assimilation of explicit knowledge	external sources extraction from implicit knowledge
Operations	similarity based constraint based	symbol manipulation based rule based
Characteristics	more context sensitive fuzzier less selective more complex	more crisp more precise more selective simpler
Representations	distributed (micro)features	symbolic-localist conceptual units

Note that Clarion is unique in its utilization of the representational difference (i.e., symbolic-localist versus distributed representation) as the main computational substrate to account for the psychological and the phenomenological differences between implicit and explicit processes. Clarion thereby provides a framework to account for all forms of implicit and explicit processes. Of course, much more work is needed in pursuing this possibility.

When are implicit processes used, and when are explicit processes used?

The question is what determines the explicitness/implicitness of processing when a particular task is being dealt with—how a task is "assigned" to one level or the other (or both). I touched upon this issue before, appealing to a generic notion of complexity (Sun, 2002). I will further explicate this notion and draw a more detailed picture of the division of labor between the two levels.

It may be speculated that the following factors, among others, may determine complexity in this regard:

- Amount of information to be considered (e.g., numbers of inputs/outputs). The higher the amount is, the more likely that implicit processes are prominent.

- Stochasticity. The more stochastic a task is, the more likely that implicit processes are prominent.
- Sequentiality (e.g., how distant temporal dependency relations are). The more sequential a task is, the more likely that implicit processes are prominent.
- Instructions (given prior to or during task performance). Generally speaking, the more explicitly focused the instructions are, the more prominent explicit processes will be.
- Multiplicity of tasks. Generally speaking, under dual-task conditions, implicit processes become more prominent, compared with single-task conditions.

When dealing with learning, if the task to be learned is complex, explicit learning mechanisms may not work well, and therefore implicit learning becomes prominent. In contrast, when the task is simple, explicit learning mechanisms may work well, and therefore explicit learning becomes more noticeable. Empirical findings and simulation work with Clarion were consistent with these speculations.

Formally, complexity may be measured by, for example, (1) the minimum encoding length of knowledge necessary for performing a task, and (2) the learnability of such knowledge. Both are formal mathematical measures. The latter measures the complexity of learning and the former the complexity of the outcome of learning. For example, complexity may be determined by the size of a minimum rule set needed to perform a task and the difficulty of learning it.

When does bottom-up learning or top-down learning occur, respectively?

Bottom-up learning generally happens when a situation is such that it is easier to learn implicit knowledge than explicit knowledge (see the earlier discussion of factors determining this), and it is possible to learn explicit knowledge on the basis of implicit knowledge.

For example, in a complex (non-salient) process control task (see Chapter 5), one often develops implicit knowledge first, given that the situation makes directly learning explicit knowledge difficult. However, after a substantial amount of implicit knowledge accumulates, explicit knowledge often emerges on that basis (Stanley et al., 1989; Sun et al., 2001).

Top-down learning usually occurs when explicit knowledge is available from external sources, or when it is relatively easy to learn such knowledge (compared with learning corresponding implicit knowledge). Such knowledge, learned or directly received, may then be assimilated into an implicit form.

For example, in learning to play chess, one often first learns the basic rules of chess, and some essential guidelines as to what to do in prototypical situations. One may then develop more complex and more nuanced knowledge that is largely implicit (Dreyfus & Dreyfus, 1987).

Of course, preferences of learning directions may vary from individual to individual. Also, real-life learning scenarios may be more complex than the examples above and thus may lead to multiple directions of learning in an intermixed way.

Many of the tasks that were dealt with by Clarion are high-level cognitive tasks. Is there any evidence that such high-level cognitive tasks involve implicit processes at all?

In general, even high-level cognitive tasks may involve implicit processes. There have been indications that high-level cognitive tasks such as Tower of Hanoi, category learning, reasoning, and so on indeed involve implicit processes. For example, Gagne and Smith (1962) showed specifically that verbalization improved subjects' performance in learning Tower of Hanoi. Bower and King (1967) showed the same effect of verbalization in classification rule learning. Gick and Holyoak (1980) found that good problem solvers in high-level problem solving domains could better state rules that described their actions in problem solving. In all of these cases, it could be the explication of implicit knowledge that helped the performance (Sun, 2002).

Some more direct arguments may be found in Dreyfus and Dreyfus (1987). Based on detailed analysis, they argued that learning to play chess involved turning analytic (explicit) thinking into intuitive (implicit) thinking through extensive practice. Evans (2003) presented evidence and arguments that even deductive reasoning might be partially implicit.

For instance, mathematical theorem proving involves intuitive (implicit) thinking to a significant extent. In theorem proving, it is important to develop good intuition. The search space of different possibilities of constructing a proof is huge, and the cost of explicit exploration of the space is prohibitive. Therefore, intuition is crucial in guiding the search.

Such intuition is (usually) implicit, as shown experimentally by Lewicki (1986), Hasher and Zacks (1979), and so on. In a sense, good (implicit) intuition is what separates a good mathematician from a poor one.

It would be unwise to add lengthy discussions of these points, which would be needed if full justification of the implicit nature of these tasks was attempted.

How does Clarion relate to folk psychological notions such as "instinct," "intuition," and "creativity"?

Based on the Clarion framework, one may reinterpret many folk psychological (and other) notions, to give them more precision.

For instance, the notion of *instinct* may be made more precise in this way. Instinct involves mostly implicit processes and is mostly concerned with action. Within Clarion, instinct may be roughly equated with the following chain of activation: *stimuli* → *drives* → *goals* → *actions*. This chain goes from stimuli received to the MS, the MCS, and eventually the ACS. That is, stimuli activate drives (especially innate motives), drive activations lead to goal setting in a mostly implicit way (with mostly implicit or even innate processes), and based on the goal set, actions are selected in a mostly implicit way to achieve the goal. Instinct is mostly implicit, but it may become more explicit, especially with regard to the part of "*goals* → *actions*" (Sun et al., 2001).

For another instance, the notion of *intuition* can also be made more precise. Intuition, according to Clarion, is roughly the following chain: *stimuli* → *drives* → *goals* → *implicit reasoning*. This chain goes from stimuli received to the MS, the MCS, the ACS, and the NACS. As such, intuition involves mostly implicit declarative processes within the NACS, including the functionalities of associative memory retrieval, soft constraint satisfaction, and partial pattern completion (see Chapter 5 for details). Intuitive processes are often complementary to explicit reasoning, and the two are often used in conjunction (Sun & Zhang, 2006).

The notion of *creativity* can also be explained within the Clarion framework (Helie & Sun, 2010). According to Clarion, creativity may be achieved mainly through complex, multiphased interaction between implicit and explicit processes within the NACS; that is, through the interplay between intuition and explicit reasoning (the two types of declarative processes), on the basis of the motivational and metacognitive underpinnings. This interpretation has been developed into the EII

theory of creative problem solving—a theory derived from Clarion (Helie & Sun, 2010). It involves multiple phases, which include (1) the explicit phase: processing given information using mostly explicit declarative knowledge; (2) the implicit phase: developing intuition using mostly implicit declarative knowledge; finally the intuition emerges into explicit processes and therefore (3) the explicit phase: verifying and validating the result using mostly explicit declarative knowledge. See Helie and Sun (2010) for further details. This theory has been successful in accounting for a variety of empirical data.

Some other folk psychological notions may be reinterpreted and made more precise in a similar manner. For example, relevant notions of the BDI framework have been reinterpreted based on Clarion, as discussed earlier. Similarly, a possible reinterpretation of serendipity was discussed in Chapter 7. Explanations of anxiety and a variety of other emotions were discussed in Chapter 6. In addition, the important notion of consciousness has been reinterpreted based on the Clarion framework, as detailed in Sun (1999).

Why are primary drives (especially high-level primary drives) innate?

The main point is that the human mind is likely innately equipped for dealing with these aspects represented by these drives. The innate mechanisms for dealing with these aspects, to the extent they exist, result from the human evolutionary history. (For instance, evolutionary psychology argued for the innateness of some of these high-level primary drives, often in specific settings.) However, these drives, along with many other mechanisms and processes, may be fine tuned, to various extents, through experiences. Therefore they are not necessarily completely fixed.

Based on the relevant literatures, two points may be argued: (1) there is strong evidence that people do normally develop these drives and have these needs (McDougall, 1936; Murray, 1938; Maslow, 1943; Reiss, 2004), and (2) there is some evidence that these drives may be relatively invariant across cultures (although there may be some quantitative variations; Chirkov, Ryan, & Willness, 2005; McRae, 2002). On the basis of these points, it is reasonable, and indeed beneficial, to posit the innate existence of these drives.

8.2. Computational Questions

Computationally, Clarion seems a random collection of computational (AI) techniques. Is Clarion just a random collection of computational techniques?

Without delving into psychological details described by Clarion, one might wrongly conclude that Clarion is just an ad hoc collection of computational techniques and algorithms. Therefore, I should point out again that the present work is not about computational techniques or algorithms, but about cognitive-psychological mechanisms and processes. The computational techniques employed in Clarion were not randomly selected but carefully put together to account for a wide range of psychological data and phenomena. That is, they were selected for the sake of developing a comprehensive theory of the mind, not for AI applications.

The meta-principle guiding the development of Clarion as a psychological theory is (as noted in Chapter 1): minimum mechanism, maximum coverage, and optimal integration. So in a way, the development of Clarion was based on cost-benefit considerations whereby the cost being the complexity of the model and the benefit being the scope of data and phenomena that it is capable of capturing and explaining.

Even given the above, is Clarion just a collection of old computational (AI) techniques? That is, is there anything new there?

One may argue that Clarion is just a set of old or even outdated AI techniques, and as such there may not be anything new. To address this issue, I should note that the novelty of computational techniques (or the lack thereof) is not even a relevant issue here. Clarion, at the conceptual level, is certainly not about developing computational techniques, but about exploring and understanding cognitive-psychological processes. Even the Clarion computational cognitive architecture itself is, primarily, not about computational techniques (although there have been technical innovations), but about a proof-of-concept demonstration of the feasibility of describing cognition-psychology broadly in a computational form. One should not confuse the theory (and the resulting cognitive architecture) with the tools that it employs in expressing itself.

As stated before, Clarion is about selectively including a minimum set of mechanisms, structured in a parsimonious but effective way, to

account for and explain a maximum set of psychological data and phenomena. Furthermore, what is emphasized in the present work is the conceptual framework of Clarion and what it can account for psychologically, not computational techniques employed in doing so. There are ample reasons to believe that the framework of Clarion is generally valid, regardless of computational details used in implementation. Currently employed computational details constitute an existence proof of what one can accomplish with this framework, in a somewhat crude manner. More recent computational techniques, if proven significantly better performance-wise or in some other way, can be relatively easily inserted into the computational cognitive architecture to replace old techniques, without significant changes to the overall framework.[2]

Solely from cognitive modeling and simulation perspectives, why are there two "levels"?

Simply put, the presence of the two levels in Clarion provides a unified, succinct account of a variety of psychological data and phenomena, ranging from serial reaction time tasks to Tower of Hanoi. In particular, various synergy effects have been simulated and accounted for by Clarion computationally through the interaction of the two levels, which include, for example, improved performance through explicit search or verbalization (e.g., Sun, Slusarz, & Terry, 2005).

In addition, the computational differences (including the representational difference) between the two levels account for a variety of empirically derived psychological constructs concerning differences and dissociations exhibited in the empirical data (such as implicit versus explicit learning, implicit versus explicit memory, unconscious versus conscious perception, intuitive versus analytical reasoning, and so on), which has been mentioned in the answers to the earlier questions, and extensively discussed elsewhere (see, e.g., Sun, 2002).

2. Computational techniques used in Clarion were often not very recent. Some of the computational techniques used there were published by my collaborators and me in the 1990s and the 2000s in AI and other computational journals, ranging from the 1995 article in the journal *Artificial Intelligence* to the 2000 article in the journal *Adaptive Behavior*. However, some computational techniques used in Clarion were developed more recently.

Where does Clarion stand on the debate of connectionist versus symbolic models?

Let us examine briefly the debate of connectionist models versus symbolic models. First, there was the question of which paradigm should be adopted as a general framework for cognitive modeling. This was an issue of great controversy among theoretically minded cognitive scientists. Many claims and counterclaims were made. Clarion sidesteps this stalemate through incorporating both paradigms, in a principled way, into its framework. I have shown that the two can be combined to generate synergy of various kinds (e.g., Sun, Slusarz, & Terry, 2005; Helie & Sun, 2010). Clarion is one of many so-called hybrid models that started in the 1990s (Sun, 1994; Sun & Bookman, 1994) and have been receiving increasing attention since then.

In relation to this issue, there was also the more specific issue of the ability (or the inability) of one type or the other in accounting for implicit processes. It has been claimed, on the connectionist side, that a vast majority of human activities, including "perception, motor behavior, fluent linguistic behavior, intuition in problem solving and game playing—in short, practically all skilled performance" (Smolensky, 1988, p. 5), should be modeled by subsymbolic computation with connectionist models, and symbolic models can give only an imprecise and approximate explanation of these processes. On the other hand, it has been claimed on the symbolicist side that symbolic models can be responsible for conscious and unconscious processes alike, or even that implicit processes are better modeled by symbolic models.

In terms of matching data of any specific task, any Turing-equivalent computational model should be able to do so. Thus, the matching of some empirical data by itself does not prove whether a particular model is a suitable one. Other considerations need to be brought in. I suggest that one such consideration is phenomenological and computational accessibility discussed earlier. While symbolic models of implicit processes lead to symbolic representation of implicit knowledge that is supposedly inaccessible phenomenologically but evidently easily accessible computationally (without any add-on assumptions regarding the representation), connectionist models lead to subsymbolic representation of implicit knowledge that is inherently less accessible computationally (such as in the bottom level of Clarion) in closer accordance with its phenomenological characteristics. Thus, connectionist models have a clear advantage

here: being able to match human data (at least) as well as symbolic models, they also account computationally for the phenomenological inaccessibility of implicit processes. In this sense, they are better models.

However, symbolic-localist models have their roles to play as well. They are better at capturing explicit processes. The phenomenological characteristics of explicit processes are closely matched by the computational characteristics of symbolic-localist models.

This contrast between connectionist and symbolic models lends support to the belief that because connectionist models are good for implicit processes and symbolic models for explicit processes, the combination or integration of the two types of models should be emphasized in modeling human cognition-psychology (Sun, 1994, 2002).

In the bottom level of Clarion, where is subsymbolic (distributed) representation exactly? Is it just an implementation of symbolic processes?

Subsymbolic representation in the bottom level of Clarion refers mainly to internal distributed representation, not necessarily inputs or outputs, of the bottom level (because inputs/outputs might be localist in a particular simulation). In general, localist representation is just a special case of distributed (subsymbolic) representation. There is no way that Clarion itself can put any enforceable restriction on inputs/outputs, especially because inputs and outputs of a model have to be tailored to the task to be simulated. Inputs/outputs are ultimately decided by the modeler who constructs a particular simulation model and its input/output representation within Clarion.

On the other hand, in the hidden layer(s) of a Backpropagation (MLP) network (Rumelhart et al., 1986), the representation is indeed generally distributed (subsymbolic), as a result of Backpropagation learning happening in the network. Such representation involves distributed activation patterns that are sensitive to regularities embodied by training data. They are not mere implementations of symbolic representation (see, e.g., Rumelhart et al., 1986; Miikkulainen, 1993).

How does distributed representation emerge in the bottom level of Clarion?

There are a number of different ways in which distributed representation may emerge. For instance,

- It may emerge through supervised learning using, for example, the Backpropagation (MLP) algorithm (when relevant training data are available). Given proper training data, distributed representation emerges in the hidden layer(s) of a multilayer network as a result of iterative weight adjustments as dictated by the Backpropagation algorithm. Miikkulainen (1993), for example, showed that such learning could form meaningful distributed patterns sensitive to regularities underlying training data (thus having corresponding semantics). Work on deep learning is also consistent with this notion (Schmidhuber, 2014).
- It may also emerge through reinforcement learning (in a simulated environment or in the real world). In this case, learning may be carried out by, for example, Backpropagation based on the error signals generated by a reinforcement learning algorithm. Backpropagation learning can form distributed representation in the hidden layer(s) of a multilayer network as indicated above (Sun, 2002).
- Distributed representation may be randomly generated (in the form of random activation vectors), as an approximation of some natural process, when the semantics of the representation is not an issue (e.g., Helie & Sun, 2010). For instance, this may serve as an approximation of distributed representation formed by these two methods above, or as an approximation of possibly innate (or a priori) distributed representation.
- It may result from gleaning representational information from other sources, for example, from statistical correlations found within large datasets, or from representations acquired or learned in other models (see, e.g., Miikkulainen, 1993; Sun, 1994).

How can symbolic-localist representation (used in the top level of Clarion) be justified? Has localist representation been discredited?

In the connectionist literature, localist representation (roughly, representing any conceptual entity with a dedicated unit) is not "discredited" in any sense, although controversies do exist (as mentioned earlier). Symbolic-localist representation, as employed in the top level of Clarion, can be fully justified.

Computationally, symbolic-localist representation is justifiable, because of the useful characteristics of symbolic-localist representation. As discussed before, in computational modeling, explicit knowledge may be better captured by symbolic-localist representation, in which each node is more easily interpretable and has a clearer conceptual meaning when compared with distributed representation. That is, symbolic-localist representation is more accessible computationally. This computational characteristic of symbolic-localist representation captures the corresponding characteristic of explicit knowledge being more accessible in a subjective, phenomenological sense (Sun, 2002).

In addition, symbolic-localist representation is often more efficient. For instance, it minimizes the effort required to interpret activation patterns. The read-out of symbolic-localist representation is made easier by the outputs of dedicated, individually meaningful nodes (Sun, 1992, 2002).

Neurobiologically, localist representation may be justified as well. Koch (2011) argued that some neurons selectively responded to very specific persons or objects. This leads to "concept neurons," which may specifically encode family members, friends, coworkers, one's car, one's laptop, and so on. Every time one encounters a particular person or object, while a pattern of activation of neurons is generated in higher-order cortical regions, the networks in the medial temporal lobe may dedicate specific neurons to them. See also Bowers (2009).

However, note that even when localist representation is in place, a single node may not be responsible for all the knowledge related to a concept. Other nodes may be activated to various extents, bringing in related knowledge and participating in deciding on final outcomes. Another possibility is that localist representation may be replicated; that is, multiple localist nodes may be used to redundantly indicate one conceptual entity (for fault tolerance or for other purposes). In fact, a whole spectrum of representational possibilities exists, ranging from fully distributed to fully localist (Sun, 1992). They vary in terms of specificity, redundancy, and amount of activation overlap. Each form may have some distinct computational properties and thus may be useful in some circumstances.

But does a more uniform and more constrained model provide deeper explanations than those with a pool of highly specialized mechanisms?

It is true that in many cases, a more uniform and/or more succinct (i.e., more "constrained") model provides deeper explanations than those with

a large pool of highly specialized mechanisms, given that the empirical coverage (as well as other relevant aspects) of the models is roughly comparable. However, in the case of some existing cognitive architectures, one has to take account of the fact that there are more severe limitations in those that are more uniform and more "constrained" in terms of the range of cognitive-psychological phenomena that they can capture.

If a model fails to capture the breadth of cognitive-psychological phenomena, then there is very little to be gained in being "uniform" or "constrained" because "deep" explanations are not likely to come out of it when it fails to explain many relevant phenomena. The mind/brain is inherently heterogeneous; uniformity cannot be forced upon it beyond a certain natural limit (see also Minsky, 1985).

Even if the above is true, why was a unified implementation (e.g., a purely connectionist implementation or a purely symbolic implementation) not attempted?

It is certainly desirable to have a unified implementation of a theory, a model, or a cognitive architecture, even a very complex one. But the key question here is whether connectionism is the best approach for computational modeling of all parts of the mind, rather than just some parts, or whether symbolicism is the best approach for computational modeling of all parts of the mind.

Naturally, one may expect to be able to implement virtually all symbolic processes in connectionist models, and vice versa. A unified implementation of Clarion was indeed produced (Sun, 2002). But what does this kind of "implementationalism" buy us? Why not just include both types of processes and save all the unnecessary trouble of "implementation"?

One would naturally prefer to use the most suitable tool for each component of a job, not just one tool for all. Likewise, I would prefer to use the best medium for implementing each component of my model, not one medium (connectionist or symbolic) for all.

At first glance, it may not seem "elegant" to use hybrid models involving both symbolic and connectionist processes. However, looking through the literature on this, most connectionist implementations of sufficiently complex symbolic processes are not "elegant" in any sense, and vice versa. Therefore, the use of hybrid models did not actually introduce any additional "inelegance" but only what is necessary for capturing the

complexity of the human mind (which is known to be highly complex, and includes both symbolic and subsymbolic processes).

Are there too many free parameters in Clarion?

At first glance, the Clarion computational cognitive architecture may seem to have too many parameters. However, upon closer examination, one may quickly realize that the number of its free parameters in any subsystem is not significantly higher than usual computational models such as Backpropagation (MLP) networks.

Let us look specifically into the action-centered subsystem as an example, which has been important in simulating a wide variety of tasks. In addition to parameters of Backpropagation (MLP) networks (as in the bottom level of the ACS), at the top level of the ACS, there are only three important parameters concerning rule extraction and revision when using RER. That is to say that the ACS of Clarion is approximately comparable to Backpropagation networks in terms of complexity.

Furthermore, although values of all parameters affect performance, most of them were not changed throughout the simulation of various conditions of a particular task (e.g., the process control task; see Sun et al., 2007), and thus they should be treated as the fixed part of model specification (for the model created to simulate the task). In this sense, they are not free parameters: that is, they do not contribute to the degree of freedom that one has to match the change of human performance across different conditions in a particular task (such as what one sees in the process control task; Sun et al., 2007).

There are in fact three different types of parameters in Clarion: (1) domain-independent parameters (e.g., the drive deficit parameters for primary drives, the momentum parameter in Backpropagation networks, and so on), (2) domain-specific parameters (e.g., the number of input units or the number of output units), and (3) free parameters. While the first two types may be viewed as part of the fixed model specification for a particular task, free parameters are those that are changed for capturing different conditions of a task (such as those different experimental conditions in the process control task as described in Sun et al., 2007). In many past simulations involving Clarion, the actual number of free parameters was usually only one or two (e.g., a rule learning threshold at the top level). See, for example, Sun, Slusarz, and Terry (2005).

Why were there often components removed from Clarion in simulations? How do you justify such seemingly arbitrary removal of components (e.g., subsystems, levels, modules, or mechanisms)?

There are a number of different aspects to this question. First, sometimes, in a simulation only the ACS is used but not the other subsystems. This may occur under some circumstances. For example, when motivational inputs are constant or insignificant to action decision making in a situation, the MS is not needed for the simulation. In that case, the MS may be removed from the simulation for the sake of simplicity. The same may be said about the MCS or the NACS.

Second, in terms of the two levels (within any subsystem), Clarion assumes that their relative contributions can be adjusted (e.g., by the MCS), depending on many contextual factors. For instance, in incubation, processing is mostly done implicitly. For another instance, in verbal reasoning, processing can be very explicit. Therefore, it is justifiable to use different configuations of the two levels, including disregarding the contribution of one of the two levels (effectively removing it).

Third, within a level of a subsystem, sometimes some of the available mechanisms are disengaged during a simulation. For example, RER or IRL may be included or excluded from a simulation. The exclusion of a mechanism may be justifiable if in a particular domain the mechanism has a negligible effect and thus does not contribute much to the outcome of the simualtion.

Why were recurrent neural networks not used in the ACS?

Indeed, recurrent neural networks, such as various types of recurrent Backpropagation networks, may be used in the bottom level of the ACS, as well as a number of other places. This point has been mentioned in Chapter 3. However, feedforward networks are generally preferred, especially for the ACS.

There are a number of reasons for this (slight) preference. First, theoretically speaking, Clarion embodies the belief in, and therefore it focuses on, direct situation-to-action mappings, that is, situated action in a general sense (as argued in Chapter 2; see also Sun, 2002). In particular, the bottom level of the ACS relies on rather direct situation-to-action mappings to capture rapid, reactive action decision making and learning during interaction with the world. Second, recurrent neural networks may not

be needed for reactive action decision making and related trial-and-error learning. In particular, the temporal credit assignment mechanisms embodied in many reinforcement learning algorithms that may be used in the ACS (e.g., Q-learning) can deal with temporal dependencies. They do so through value "backup" (updating the value of the current state from values of future states; see Sun & Peterson, 1998; Bertsekas & Tsitsiklis, 1996; Sutton & Barto, 1998).

However, if recurrent networks are needed for any reason in any particular context, they can be easily used in place of feedforward networks.

Can communication be included in Clarion?

From the perspective of Clarion, communication is a type of "external" action. Therefore, Clarion should be able to include it, in particular, through the use of the action-centered subsystem (along with the non-action-centered subsystem).

However, currently, there is no fully developed model in Clarion for addressing language-specific issues such as syntactic parsing. Although they are not currently specifically included, natural language comprehension and production can certainly be carried out within Clarion.

Can sensory-motor processes be included in Clarion?

Currently, detailed sensory-motor processes are not included in Clarion. However, they can certainly be added into the cognitive architecture, at some level of abstraction. In fact, there has been in the past some implementation of sensory-motor processes in Clarion based on EPIC (Meyer and Kieras, 1997), although they are not currently included.

What real-world tasks can Clarion tackle computationally?

Clarion, including both the theoretical framework and the computational cognitive architecture, has been developed for the purpose of computational psychology—developing computational models of psychological processes underlying human performance and therefore detailed (mechanistic, process-based) explanations of psychological phenomena and data. I do not know if this fits in with anyone's personal definition of "real-world" tasks. But from my perspective, such tasks are indeed real, intellectually interesting, and useful to explore.

Besides, Clarion has found some (preliminary) applications recently in social simulation, organizational research, anthropological modeling, interactive story telling, music agents, and so on.

If one is more interested in engineering applications, robotics, data mining, or the like, there is not much that is directly relevant here. But engineering applications and the like are not the only "real-world" tasks.

Does Clarion produce anything like human-level general intelligence?

There have been many simulations based on Clarion, but many of them deal with small laboratory tasks. So, naturally, one may ask whether or not Clarion can address tasks that are generally considered difficult for humans. Indeed, Clarion has not been applied to the modeling of some tasks that have been tackled by some other cognitive architectures, such as learning algebra or flying UAVs. However, Clarion has been addressing other, probably equally difficult tasks. As pointed out before, such tasks include minefield navigation, analogical reasoning, metaphor, moral reasoning, social simulation, anthropological modeling, game playing, interactive story telling, and so on. The focus of Clarion is somewhat different from some other cognitive architectures, and thus the tasks addressed are also different as a result.

Clarion has the necessary ingredients for producing general human-level intelligence beyond these tasks tackled so far. As discussed earlier, Clarion takes into account necessary desiderata concerning human cognition-psychology. On that basis, an essential theoretical framework was developed, which led to computational implementations. Even though the computational details may change somehow over time, the essential framework may prove to be of fundamental importance in capturing general human-level intelligence.

Why has social simulation been emphasized in work on Clarion?

Social simulation has been an important development in the social sciences. It has been heralded as the new approach to the social sciences, offering a number of significant advantages over the other approaches (some of which have been discussed earlier).

Besides its relevance to the social sciences, social simulation is also relevant to cognitive science. Instead of simulating psychological functioning of an isolated individual alone, social simulation allows the inclusion

of social processes as well. In so doing, one may use cognitive architectures as building blocks. This approach puts an individual in a social context, which results in more complete models of individuals—hence its significance to cognitive science and to cognitive architectures.

Why has tribal simulation been emphasized in social simulation with Clarion?

One has to start from somewhere. It is natural to start from the simplest. Simple tribal society has been simulated by a number of other researchers; it is thus a good domain to begin with.

This approach tends to address the most basic forms of social processes (because of the focus on simple societies), and thus likely fundamental principles of social processes. In a way, it may be viewed as pursuing cognitive social sciences from the ground up (Sun, 2012b).

From there, we may move on to increasingly complex forms of societies and explore increasingly complex social processes. The current tribal simulation is just a step in this progression from the simplest to the most complex.

8.3. Biological Connections

Is it true that Clarion has no clear biological connections?

This work on Clarion is at the psychological level, as is common for most work in cognitive science, not the neurobiological level (as in cognitive neuroscience). Therefore, it does not deal much with neurobiological details directly. In other words, it tends to be at a higher level of abstraction (at least currently).

Level of abstraction is a crucial and proven notion. For example, we need thermal dynamics, classical mechanics, as well as quantum mechanics. We do not abandon all higher-level theories in favor of quantum physics. Likewise, we need sociological-anthropological theories, psychological theories, as well as biological theories. We do not abandon all higher-level theories in favor of biology. This is because each different level of abstraction may shed some unique light.

It is possible and indeed preferable to develop psychology (including computational psychology) in a way that is more abstract than

neuroscience. Given the current state of neuroscience and psychology, this type of work is needed: it sheds different light and provides different insights on issues of brain, mind, and behavior, and possibly on the social sciences as well (see, e.g., Sun, 2009b and Sun 2012b).

Are biological models better models of the mind?

The answer is: not necessarily. One pertinent question in this regard is the following: what principled understanding of the human mind does a particular biological model provide? Sometimes (but certainly not always), biological models (or biologically motivated models) of the mind do not add much beyond psychological theories or models; sometimes they may offer less than psychological theories or models. But they carry with them the added burden of justifying and validating (often minute) biological details.

By presenting a biological (or biologically motivated) model, the burden of validation is multiplied as a result of making biological claims, because

- in biological models, each element within a model is a biological claim that needs to be validated in a biological way, which is sometimes difficult or impossible
- in biological models, internal structures connecting various elements need to be detailed in a biological way and validated, which is often a rather difficult or impossible task, and is often not needed for a principled understanding.

Given the difficulties above, some (but not all) biological models are highly speculative and would be better off if their overly speculative biological claims were dropped. What is left is often basically a cognitive-psychological model, which might sometimes be more illuminating and more justifiable.

Will all psychological models eventually be replaced by biological models?

As argued above, psychological models, in general, are useful in their own right. They are often complementary, or even sometimes superior, to neurobiological models, and thus not easily replaceable. For instance, they offer the following advantages:

- In case not enough is known about the biological substrate of a particular psychological function or phenomenon, psychological models (especially computational psychological models) are what one may propose to explore the function or phenomenon.
- Psychological models often entail fewer claims (especially fewer claims pertaining to biological substrates) and therefore require less tedious validation (especially biological validation), while still providing a principled understanding.
- Psychological models often enjoy the benefit of simplicity and compactness, compared with biological models, and thus Occam's razor would lead to preferring such models under some circumstances (e.g., when biological details are not needed).
- Psychological models may lead to new hypotheses regarding biological substrates; that is, psychological models may lead to (e.g., inspire) biological models and/or related empirical work (i.e. going from higher levels to lower levels in an abstraction hierarchy).

Can Clarion embody biological constraints?

Clarion can certainly incorporate biological/physiological constraints in many ways. For instance, it may incorporate such constraints by providing only biologically realistic sensory information to a simulated organism (i.e., embodying perceptual constraints). For another example, it may incorporate biological/physiological constraints by allowing only a biologically realistic range of physical action choices in models (embodying physical action constraints). Clarion can also embody what is biologically possible in terms of cognitive capabilities. Clarion can also incorporate time constraints of biological organisms, in relation to perception time, memory retrieval time, reasoning time, physical action time, and so on. In addition, Clarion can embody realistic motivations of biological organisms (to the extent that they are understood).

9

General Discussions and Conclusions

At this point, two major questions concerning the work described thus far come to mind: Where are we now? Where should we go from here? The first question will be dealt with (partially) in sections 1 and 2 below. Sections 3, 4, and 5 further link Clarion to some existing ideas, approaches, and models, thus also addressing the first question above. The second question will be addressed in the last section of this chapter.

9.1. A Summary of the Cognitive Architecture

Clarion is distinguished by the combination of a number of characteristics. Those characteristics, as discussed before, include the following:

- Clarion is more comprehensive in terms of functionalities than most existing cognitive architectures, while capturing fine-grained mechanistic and process details.
- It centers on a dual-process, dual-representation framework that is theoretically well justified.
- It is hierarchically structured and modular, but highly interactive (both internally and externally). It addresses the question of how pieces fit together in the overall architecture.

- Its basic principles, assumptions, and desiderata have been well justified in relation to a variety of different types of evidence (empirical and theoretical).
- It has been validated through simulating and explaining a wide variety of psychological tasks, data, and phenomena, capturing their subtleties to a significant extent.
- It has led to a number of new theories in different domains.
- It has also taken into account higher levels, for example, concerning social processes and phenomena, as well as lower levels.

Take a look at one of these characteristics above in particular. Clarion embraces modularity and incorporates various relatively independent components. For instance, at the highest level, it is divided into four major subsystems: the action-centered subsystem, the non-action-centered subsystem, the motivational subsystem, and the metacognitive subsystem, each responsible for a relatively isolatable functionality. For another instance, there are a variety of memory stores in Clarion: semantic memory in both implicit and explicit forms (in the NACS), procedural memory in both implicit and explicit forms (in the ACS), episodic memory (in the NACS), working memory (in the ACS), goal structures (in the ACS), and so on, with a relatively clear division of labor among them.

While being modular, Clarion does not follow a strictly modular approach. It accentuates complex dynamic interactions of various processes within or across various modules in a context-sensitive way. Interactions may include those among various psychological functionalities: perception, categorization, memory, decision making, reasoning, action, learning, motivation, metacognition, and so on. Interactions may also include those between the psychological functionalities and the external or internal world, geared toward surviving and functioning in the world.

One possible argument against Clarion is that simpler models (e.g., with fewer modules) may provide better explanations, which may be true in general. However, as argued before, there are severe limitations in simpler models in terms of the range of phenomena that they can account for. Without capturing a broad range of psychological phenomena, there is little to be gained in being simple. It is unlikely that deep explanations come out of a model when it cannot capture many relevant psychological phenomena. As mentioned before, cognitive architectures need to

incorporate all functionalities, including perception, categorization and concepts, memory, reasoning, decision making, planning, problem solving, action control, learning, motivation, emotion, metacognition, and others (while recognizing that certain other functionalities, such as natural language processing or sensory-motor processing, may be somewhat separate). This point has been raised early on in the history of cognitive science (e.g., Newell, 1990) but is still a major issue.

Complex models, however, have always invoked suspicion in psychology. Miller et al. (1960) cited an argument against generic models: "A good scientist can draw an elephant with three parameters, and with four he can tie a knot in its tail. There must be hundred of parameters floating around in this kind of theory and nobody will ever be able to untangle them." Counter-arguments to such objections have been advanced on the basis of the necessity of having complex models in understanding the mind, due to the inherent complexity of the human mind, for example, as argued by Miller et al. (1960), Minsky (1985), Newell (1990), Sun (2002), and so on. Nevertheless, it has been clearly recognized in the work on Clarion that over-generality, beyond what is minimally necessary, is always a danger and thus should be strenuously avoided.

In all, Clarion is grounded in empirical research, is reasonably compact (given its broad scope), and accounts for a wide range of empirical data (as has been discussed thus far and beyond).

9.2. A Discussion of the Methodologies

This work on Clarion takes an integrative approach. So far, a preliminary version of a comprehensive and integrative cognitive architecture has been described. This cognitive architecture, aimed to be comprehensive and integrative, encompasses different representations, mechanisms, and processes, different cognitive-psychological functionalities, and many different tasks. I have been trying to achieve full functionalities as much as possible (while recognizing that certain functionalities may be somewhat separate and that more work needs to be done).

The work around Clarion combines strands of cognitive modeling (computational psychology), cognitive psychology, personality psychology, social psychology, psychology of motivation, moral psychology, as well as sociological, anthropological, and political sciences, among others.

Indeed, advancing the research on cognitive architectures beyond the narrowly defined notion of "cognition" has been a focal point of the work. Progress has been made in this regard. It is achieved through incorporating motivation, metacognition, emotion, personality, creativity, and many other aspects into a cognitive architecture, notably not as an add-on or an auxiliary, but as an integral part. Given the breadth of this work, there are many possibilities for extension, and a great deal of work remains to be done.

To constrain the cognitive architecture, two meta-principles have been adopted, as discussed earlier: (a) completeness of functionalities, but also (b) parsimony of mechanisms. As discussed earlier, the ultimate goal is to come up with a cognitive architecture with as few mechanisms and parameters as possible, while accounting for as wide a range of empirical phenomena as possible, in as wide a variety of domains as possible.

Coming up with a well-constrained cognitive architecture with few mechanisms and parameters while accounting for a great deal of empirical data, as has been attempted in the work on Clarion, is clearly difficult. The difficulty has been addressed to some extent, through adopting a broad perspective (philosophical, psychological, biological, as well as computational), and through adopting a multilevel framework (involving biological, componential, psychological, and sociological levels, as argued in detail in Sun, Coward, & Zenzen, 2005b).

However, some have argued that one cannot understand cognition based on cognitive architectures—this approach is wrongheaded because of the variability of empirical data and phenomena. For instance, as some have claimed, the distinction between implicit versus explicit processes, which Clarion emphasizes, has a "questionable" empirical pedigree and cannot be relied upon: such dichotomies often generate interest initially, but after a while they are no longer "trusted" because of the complexity of empirical findings.

Contrary to what these critics claimed, this phenomenon illustrates exactly why cognitive architectures are needed in cognitive science and psychology. The need for them arises because of the complexity and variability of experimental results in experimental psychology and the difficulty with their interpretations. There are too many contextual factors and too many minute variations in experimental settings, for example. Therefore, it might be futile to try to understand the mind purely through behavioral experiments (e.g., see Sun, Coward, & Zenzen, 2005b

and Sun, 2009b for discussions of this point; see also Hintzman, 1990). To address this problem, cognitive architectures may be used to provide a framework in which empirical phenomena and their variations are coherently interpreted.

For instance, regarding the implicit-explicit distinction specifically, Clarion has indeed provided some clarity. Sun, Slusarz, and Terry (2005) presented detailed discussions of how experimental subtleties and variabilities, and even apparently contradictory empirical findings, were accounted for within Clarion. Sun (2002) also provided extensive discussions in clarifying this distinction and its subtleties based on Clarion. See also the relevant discussion in Chapter 3 regarding the whole spectrum of conscious and unconscious phenomena between the two ends: the purely implicit and the purely explicit.

As pointed out before, despite some successes so far, many kinds of additions to and a great deal of refinements of the Clarion cognitive architecture are still very much needed. These additions and refinements, following the methodologies outlined above, need to be accomplished in future work, as will be discussed later in Section 6.

9.3. Relations to Some Important Notions

I will now discuss a number of important or otherwise popular theoretical notions and their relationships to the Clarion framework.

First, the implicit-explicit distinction, which is central to Clarion, closely relates to the notion of consciousness because this distinction involves, in its core, the issue of awareness (or conscious "access" in a phenomenological sense), which is the key to the notion of consciousness. The exploration of the implicit-explicit distinction as carried out within the Clarion framework (see Sun, 2002) may help to understand issues concerning consciousness by identifying computational mechanisms and processes underlying consciousness. That is, Clarion may help with identifying computational correlates or substrates of consciousness (Sun, 1997; Sun, 1999).

Clarion may shed some light on the question of what constitutes consciousness. My central conjecture in this regard has been that direct computational accessibility resulting from explicit (symbolic-localist) representation, along with direct access or manipulation on such

representation, leads to awareness and thus constitutes the essence of consciousness (Sun 1997, 1999). Clarion naturally captures the difference between phenomenological accessibility and inaccessibility through explicit and implicit computational processes, with the use of symbolic-localist and distributed representation respectively, which provide a plausible grounding for the notion of phenomenological accessibility and hence the phenomenological notions of awareness and consciousness.

Thus, in Clarion, being "conscious" implies direct accessibility and direct access computationally (e.g., activation and use of explicit representation), while being "unconscious" implies inaccessibility (or indirect accessibility). That is, explicit representations in Clarion are either conscious or potentially conscious, depending on computationally whether they are being activated and used or not; implicit representations are not conscious at all (but they may pass information to explicit processes and thereby become conscious in an indirect way).

Although there are a variety of views concerning consciousness, each based on a different physical substrate, Sun (1999) argued that the distinction between symbolic-localist and distributed representation provided a better alternative. There are of course also dualistic views that rely on the assumption of existence of nonphysical entities or properties, which I did not deal with. All things considered, Clarion has significant bearings on theorizing on consciousness. The reader is referred to Sun (1999) for details.

Second, Clarion has something to say about the notion of automaticity in psychology. In the past, the notion of automaticity has been variously associated with the following phenomena:

- the absence of competition for limited resources (attention) and thus the lack of performance degradation in multitask settings
- the absence of conscious control/intervention/intention in performing a task
- the general inaccessibility of processes
- the general speedup of skill performance

Clarion is consistent with these characteristics of automaticity. The top level (e.g., of the ACS) accounts for controlled processes (with the opposites of these characteristics identified above), and the bottom level

(e.g., of the ACS) has the potential of accounting for all the aforementioned characteristics of automatic processes, as has been shown in various task settings.

In various previous simulations, these characteristics have been covered separately: the speedup of skill performance, the direct inaccessibility of knowledge and processes at the bottom level (including running without conscious intervention), and the lack of resource competition (due to the coexistence of multiple modules that run in parallel at the bottom level). For details, see Sun (2002). Thus, in Clarion, automaticity serves as an umbrella term that describes a set of characteristics of the implicit processes in the bottom level.

On a related note, what is commonly referred to as automatic processing in the literature is often the result of top-down learning (implicitation or automatization), while within the Clarion framework implicit processes may sometimes (though not always) be the beginning of bottom-up learning (explicitation; Sun, 2002). So, in this sense (and in some other senses), the Clarion notion of implicit process is broader than the notion of automatic process.

Third, attention is an important notion in psychology. How does Clarion account for attention as a psychological construct? The computational correlate of attention in Clarion includes the following:

- the activation of a particular representation
- the use of that representation (e.g., in reasoning or in speech production)

Thus, computationally, attention is (in part) based on activation of representation. But, is it at the explicit (top) level of Clarion only, or can it be at either level? Or must it be at both levels? The answer depends on the very definition of attention—whether consciousness is assumed in attention. If consciousness is not required, then the computational correlate of attention specified above may suffice. If it is required, explicitness of representation (direct computational accessibility) needs to be added as a third condition. This stricter notion of attention covers a smaller set of phenomena in Clarion, compared with the looser notion. (In the literature, I have seen both types of "attention.")

Fourth, executive control has been an important notion in psychology, and thus it needs to be addressed here. Evidently, this notion is loaded (as is the case with many notions in psychology). Its theoretical status

is less than completely clear. Having said that, there is indeed some "executive control" in Clarion. However, this notion is somewhat generalized in Clarion, linked up with some other control functions and more distributed.

For example, what the ACS of Clarion controls includes not only actions that affect the external world but also actions that affect some other subsystems (e.g., the NACS). That is, in Clarion, the control of internal processes (namely, "executive control") and the control of external processes may be similarly done; they may both result from the decisions of the ACS.

When the ACS directs reasoning occurring in the NACS, the decision process may be highly similar to the decision process for external action. For instance, the cognitive processes behind "Should I terminate the pursuit?" and behind "Should I think about this [using the NACS] before terminating the pursuit?" should be similar and carried out by the same mechanism, even though one is concerned only with external action and the other is also concerned with internal action. Instead of relying on something else for controlling reasoning within the NACS, the ACS may be used. Including internal action selection within the ACS, in addition to external action selection, is also beneficial to the coordination of internal and external action (as shown by the example above).

On top of that, there is metacognitive monitoring and regulation/control, in the MCS of Clarion. The MCS may alter the functioning of other subsystems. Some may regard some of the functions performed by the MCS as also belonging to "executive control." In that sense, executive control is distributed between the ACS and the MCS in Clarion.

The justification for this approach lies in the two meta-principles discussed earlier: completeness of functionalities and parsimony of mechanisms. Completeness of functionalities requires the inclusion of metacognitive mechanisms, the functional distinctiveness of which leads to a separate MCS; parsimony requires that the ACS be used for the control of internal as well as external action, when the control processes are similar. So the upshot is that executive control in Clarion is layered, distributed, and generalized. Note that as a result there is no homunculus in Clarion.

I should also explore the notion of working memory within Clarion. The specific component named "working memory" in Clarion is narrower in scope than more general notions of working memory as has

been variously used in the psychological literature. Because in the literature the notion of working memory has been used to denote a number of different phenomena and has often been vague, a re-definition is necessary for precisely specifying a cognitive architecture.

Despite this narrower definition, Clarion, as a whole, can account for many "working memory" phenomena, either through the working memory as defined in Clarion or through other components present in Clarion. For example, Clarion accounts for the limited working memory capacity, the need for refreshing working memory, the limited number of explicit hypotheses that can be entertained at the same time (during procedural or declarative learning or during reasoning), the limited ability to deal explicitly with long-range temporal dependencies, and so on, through various mechanisms and processes within Clarion.

Then why does Clarion have a component called working memory after all? Rather than the vague notion in the literature, in Clarion, working memory is used specifically for the following:

- to facilitate action decision making in the ACS by storing relevant information useful for action decision making
- to facilitate communication between the ACS and the NACS
- to account for empirical data related to the two functions above (i.e., it accounts for some "working memory" data, while other components within Clarion account for other "working memory" data)

Note also that the working memory in Clarion is neither solely implicit nor solely explicit. It includes both at the same time.

I will now turn to address instance-based theories. Logan (1988) showed that skill learning (automatization) could be captured by the acquisition of a domain-specific knowledge base that was composed of experienced instances represented in individuated forms. Shanks and St. John (1994) developed a theoretical perspective in which implicit learning was viewed as nothing more than learning instances (which, however, has been criticized for various failings).

At first glance, these theories may seem at odds with Clarion. However, upon a closer examination, it is clear that the networks used in the bottom level of Clarion can be either mostly exemplar-based (essentially storing instances) or mostly prototype-based (summarizing instances), often depending on the parameters and structures of the networks.

Similarity-based processes necessary for instance-based theories can be embodied in connectionist networks, which are known to excel in such processes (Sun 1994, 1995). Instance-based theories, however, generally do not account for the learning of generic, explicit knowledge, nor for bottom-up learning.

In addition, episodic memory in Clarion may carry out instance-based processes (for reasoning, learning, and so on). For instance, appraisal of current situations, in the process of action decision making or in emotion processing, may be accomplished through comparisons with past experiences (in the form of instances) stored in the episodic memory.

Finally, there is the question of how Clarion can account for many different kinds of priming found in the empirical literature. There have been discussions of various types of priming scattered in the literature, including lexical, semantic, associative, and other priming. Some of these types of priming are perceptual, and some others are conceptual. Their effects may be discerned through the speed, likelihood, or accuracy of occurrence of a certain type of action or reaction. These different types of priming are, of course, accounted for in different ways within Clarion. For instance, motivational priming is accounted for by Clarion through persistence of drive and goal activations within the motivational subsystem, as touched upon in Chapter 4 (see also Sun & Wilson, 2014b). Action priming is accounted for through action persistence as well as base-level activations within the action-centered subsystem of Clarion, as discussed in Chapter 3 (see also Sun & Wilson, 2014b). Semantic priming occurs through (micro)feature representations at the bottom level and similarity-based processes (as well as base-level activations), within the action-centered and the non-action-centered subsystem, as discussed in Chapter 3. Associative priming, on the other hand, may also involve associative rules and chunks (and their base-level activations) within the non-action-centered subsystem.

9.4. Relations to Some Existing Approaches

Clarion is related to the approach of situated/embodied cognition. Various situated/embodied cognition views claim that cognition (or psychological functioning in general) is closely coupled to the world and

reacts to the world as experienced (Clancey, 1997; Sun, 2002). The mind does not rely on a single general-purpose symbol processor, with overly complex symbolic representation. Instead, it contains a large number of specialized systems that work together to achieve various types of functionalities, which are functionally equivalent to a general-purpose symbol processor but closely coupled to and acting in response to situations as experienced.

Such an approach is in fact fully compatible with Clarion. It is actually the meta-theoretical foundation of Clarion, as has been argued before. It was implied by the framework of the ecological-functional approach alluded to in Chapter 1. Moreover, my earlier work on Clarion (Sun, 2002) provided detailed arguments in this regard.

Briefly put, Clarion does contain a set of specialized modules (e.g., subsystems, components within subsystems, and so on) interacting with each other. There is no central symbol processor in the traditional sense (as advocated by, e.g., Newell & Simon, 1976). The processing within Clarion is the result of the interaction of various components, which together give rise to cognitive-psychological phenomena. Clarion is closely coupled to (and acts in response to) situations as perceived, especially within the ACS. It avoids unnecessarily complex symbolic representation. Moreover, in Clarion, even perception is shaped by its interaction with the world, that is, by its actions and reactions in the world (see the discussion regarding concept learning in Chapter 3). Cognitive-psychological processes within Clarion are in general shaped by its interaction with the world.

However, Clarion goes beyond narrower conceptions of situated/embodied cognition. It does so in the following ways: (1) although no general-purpose, centralized symbol processor is posited, Clarion addresses the existence of symbolic processes in human cognition-psychology; (2) Clarion also addresses the emergence of symbolic processes from ongoing subsymbolic processes in interacting with the world (i.e., bottom-up learning); (3) Clarion, furthermore, addresses the grounding of symbolic representations in subsymbolic processes and in ongoing interactions with the world. In so doing, Clarion demonstrates the relevance of symbolic processes even in situated cognition, as discussed at length in Sun (2002).

Related to situated cognition, there is also the enactive AI approach. Clarion agrees with, and embodies to various extents, the following tenets of the enactive AI approach: (1) An individual situates in the

world, interacting with the world in a direct way, which is the basis of the individual's cognition-psychology (as discussed in chapters 1 and 2); (2) the individual learns and adapts in the process of interacting with the world (see Chapter 3); (3) the individual is embodied physically, and this physical embodiment has ramifications for cognition-psychology;[1] (4) the individual is self-sustained and self-reproduced ("autopioesis"); (5) the individual and the world are codetermined by each other (because the world is, in some sense, the projection of the individual, and the individual consists largely of the patterns of interactions with the world); (6) the behavior of the individual necessarily reflects an intrinsic teleology forged by a long evolutionary history (e.g., primary drives; see Chapter 4).

However, Clarion goes beyond those points identified above, and it also argues for the following points: (1) the (innate) distinction of implicit and explicit processes, (2) the dual-representation approach toward capturing this distinction, (3) the importance of symbolic processes in the resulting system, (4) the importance of bottom-up learning in the resulting system (i.e., the emergence of symbolic representation from subsymbolic representation in interaction with the world). See the earlier chapters (as well as Sun, 2002) for discussions regarding these points.

Finally, I will relate Clarion to the dynamic systems approach. Some have claimed that cognitive-psychological phenomena cannot be divided neatly among mental functions, processes, or representations. They can only be captured through interactions, among histories, stimuli, cognitive factors, task demands, culture, language, and so on. It has been claimed that the essence of cognition-psychology is such context sensitivity, and it is incompatible with the idea of developing cognitive architectures. In response, I would argue that Clarion does account for context-sensitive, interactive dynamics. It accounts for complex, context-sensitive, and sometimes seemingly contradictory empirical findings from the interaction of various mechanisms, processes, and representations—that is, from internal and external dynamics. The discussions in the present volume so far should be sufficient in terms of demonstrating this point (see also Sun, 2002).

1. Note, however, that points 3 and 4 are beyond the scope of the current work, in that they involve a lot of biological processes.

To enthusiasts of the dynamic systems approach, Clarion is indeed a dynamic system, with many interacting components. Perception, categorization, memory, decision making, reasoning, planning, problem solving, metacognition, communication, action, motivation, and so on all interact with each other, through various representations (in a broad sense) and learning of all sorts. Furthermore, patterns of interaction change with changing task demands, physical environments, cultural milieus, and other contextual factors. In Clarion, effects of all cognitive-psychological factors are in flux, so to speak, with respect to each other and with respect to the contexts in which they are embedded.

One may take this vast catalog of interactions at face value and make the claim that there is no "fixed frame of reference" (such as a cognitive architecture), and cognitive science should give up the misleading pursuit of a "fixed frame of reference." One may claim that one should instead pursue context-sensitive structures to avoid the "reductive logic" of a "fixed frame of reference." I disagree with such defeatist claims. A dynamic system may be attributed to its constituting elements—otherwise, the field of dynamic systems does not need to exist. Cognitive architectures do not have to (though they may at times) represent a "reductive logic" or a "fixed frame of reference," any more than any other possible implementation of dynamic systems. For example, neural networks and other learning algorithms in Clarion capture "context-sensitive structures" well, in fact. The pursuit of cognitive architectures is by no means "misleading." The exploration of cognitive architectures should be integrated with the dynamic systems approach, but not replaced by it.

9.5. Comparisons with Other Cognitive Architectures

Now let us look into a number of other cognitive architectures and compare them to Clarion.

First, one may explore how Clarion is different from ACT-R (Anderson & Lebiere, 1998; Anderson, 1983, 2007). There are a number of major differences between Clarion and ACT-R, due in no small part to the basic underlying philosophical differences between the two, and to the different eras in which they were first conceived. I will enumerate only a few major differences below.

A very central difference between the two is in regard to a principled distinction and separation of implicit and explicit processes (memory, knowledge, and representation). In ACT-R, there is no such distinction; that is, there is no principled explanation of the difference between implicit and explicit cognitive-psychological processes (e.g., based on representational substrates as in Clarion), although there are differences between symbolic and numerical representations in ACT-R (Anderson, 2007). Ad hoc assumptions have to be made in ACT-R regarding which component is explicit or implicit (as discussed in Chapter 2 earlier). As a result, ACT-R does not naturally capture the psychological processes of the implicit-explicit interaction (Sun, 2002). It provides no direct explanation of the effects resulting from the interaction as observed in empirical data (e.g., the synergy effects; Sun, Slusarz, & Terry, 2005).

ACT-R does capture the distinction between procedural and declarative knowledge, analogous to the differences between the ACS and the NACS in Clarion. It thus has two main memory modules: procedural and declarative memory. Short-term memory is captured by activation traces, without a dedicated working memory component (Anderson & Lebiere, 1998).

Another major difference is that ACT-R, from the beginning, was not meant for autonomous learning without a great deal of a priori (pre-given, hand-coded) knowledge to begin with. Similarly, it does not directly capture the psychological process of bottom-up learning, due to the absence of the implicit-explicit distinction and consequently the dual representational structure. Its learning typically goes from declarative knowledge to procedural knowledge (Anderson, 1983).

Furthermore, Clarion, as a result of its dual representational structure, is capable of effortless "automatic" similarity-based reasoning, whereas ACT-R has to rely on computationally costly pairwise similarity relations to carry out similarity-based reasoning, which do not seem cognitively-psychologically realistic (Sun, 1994).

In ACT-R, there is no sufficiently developed, psychologically realistic, built-in modeling and explanation of motivation beyond simple goals. As a result, in a sense, goals are externally imposed and hand coded. They do not adequately reflect the complexity, diversity, and flexibility of human motivation and behavior. Likewise, in ACT-R, traditionally, there was no sufficiently developed, psychologically realistic, built-in modeling and explanation of metacognition, although some metacognition has been added recently.

Clarion, due to the involvement of distributed representation (e.g., in the hidden layers of the neural networks at the bottom level), has a general function approximation capability, as has been shown mathematically in the neural network literature. It is not clear how ACT-R can address this issue.

On the other hand, ACT-R has some detailed sensory-motor modules that Clarion currently does not include (in the current release of the code library at least, although such capabilities were implemented before in Clarion).

Finally, although there have been some overlaps, Clarion and ACT-R often account for different tasks. ACT-R has been used to model many different cognitive tasks by many different researchers. Its main strengths lie in modeling skill acquisition in a direction that goes from declarative to procedural knowledge (Anderson & Lebiere, 1998; Anderson, 1983, 1993). It has been used to model human-machine interaction and to build tutoring systems. It has also been used to model natural language processing, multitasking performance, and other tasks. It has been less successful at tackling human reasoning, motivation, and social interaction.

Some of the aforementioned differences may stem from the difference in emphasis, while others are more substantive, reflecting fundamental philosophical differences.[2]

Next, Clarion is also different from Soar (e.g., Rosenbloom et al., 1993; Laird, 2012). Soar is based on state spaces and operators for searching state spaces. In Soar, when there is a goal on a goal stack, different production rules propose different operators. When a sequence of production rules leads to achieving a goal, "chunking" occurs, which creates a single production rule that summarizes the sequence. Different from Clarion (which is capable of autonomous and bottom-up learning), traditionally a large amount of initial (a priori) knowledge about states and operators is required by Soar and needs to be hand coded, although recently reinforcement learning and so on were added to alleviate this problem.

Soar is different from Clarion also because Soar makes no distinction between explicit and implicit processes. Although there are differences between symbolic and numerical representations, they do not sufficiently capture the psychological distinction between implicit and explicit

2. Some ACT-R ideas, such as the distinction between procedural and declarative processes, priming by base-level activation, and so on, have been incorporated into Clarion.

processes (Sun, 2002). Therefore, in Soar, there are no (built-in) modeling and explanation of the interaction and the synergy between implicit and explicit processes. It does not distinguish between procedural and declarative processes either. Therefore it does not have the correspondingly separate memory modules, although episodic memory has been added recently (Laird, 2012).

In Soar, there is no distinction between symbolic-localist and distributed representation. It does not naturally capture similarity-based reasoning (e.g., based on dual representation as in Clarion). Also, due to the absence of distributed representation, there is no demonstration of sufficient function approximation capability.

As indicated above, learning in Soar is mostly based on symbolic representation using specialization, that is, "chunking"—creating a single production rule that summarizes a sequence of steps, which is a form of explanation-based learning. Some recent additions to Soar include reinforcement learning and some other learning methods (Laird, 2012), which are similar to Clarion to some extent. There is no (built-in) modeling of bottom-up learning though.

In Soar, there is no sufficiently complex, psychologically realistic, built-in motivational process (beyond subgoaling). Nor is there sufficiently complex, psychologically realistic, built-in metacognitive process in Soar, although some mechanisms there might be loosely described as metacognitive.

Soar has been used to model and simulate some psychological tasks, including skill acquisition (e.g., power law of practice; Newell, 1990). In addition, some theories of human emotion were implemented in Soar, although they were not intrinsic to Soar (e.g., Marsella & Gratch, 2009; Marinier et al., 2009). Soar is evidently capable of tackling decision making, reasoning, and problem solving tasks (e.g., Ritter & Bibby, 2008), although it has not been used extensively to address psychological data (perhaps because its focus has been elsewhere). However, Soar has been used in large-scale military simulations and in addressing social interaction.

Additionally, one may look into Psi (Bach, 2009).[3] Psi addresses autonomous learning, as Clarion does, although its learning algorithms were less developed algorithmically. Psi does not include the implicit-explicit

3. For a long time, there was no comprehensive description of Psi in English beyond brief mentions in review papers. Bach (2009) is, relatively speaking, the most comprehensive source of information so far. This comparison is thus based on Bach (2009).

distinction, although the reactive-planning distinction came close. Psi does not address bottom-up learning and top-down learning. Psi has a number of memory modules, including a working memory and a long-term memory store.

Like Clarion, Psi addresses the regulation of behaviors. In Psi, each goal-directed action has its source in a motive that connects a goal to an "urge," which is related to a physiological, cognitive, or social "demand" (Bach, 2009). When a goal is reached, a demand may be (partially or completely) fulfilled, which creates a pleasure signal that is used for learning by strengthening the associations of the goal with the actions carried out and the situations that led to the fulfillment.

So Psi is driven by demands. Some demands are for external resources, whereas others are abstract cognitive demands (such as certainty and competence) and social demands. There is a threshold for each demand. A deviation from the threshold is signaled as an urge, which then gives rise to a motive (with a goal). There may be multiple motives at any given time but only one "ruling motive." Actions are produced according to the ruling motive (with its associated goal).

In handling a motive, Psi goes through three stages. First, it tries to recall an automatic reaction. If no such reaction exists, it attempts to construct a plan, utilizing existing knowledge. If both automatic and planning attempts fail, it resorts to exploration by trial and error. Whenever a demand is satisfied, links are strengthened so that relevant situations become associated to the demand and the sequence of events that lead to the satisfaction of the demand, which may be used for planning.

Although Psi is similar to Clarion, it does not have as much empirical grounding. It has not been used extensively for modeling psychological processes and phenomena in accordance with empirical data and thus has been less well-validated psychologically. So far it appears that Psi does not have the mathematical-computational sophistication that other cognitive architectures have. However, in terms of comprehensiveness, Psi appears the closest to Clarion among existing cognitive architectures.

There are also biologically inspired cognitive architectures, such as Grossberg (1982), O'Reilly and Munakata (2000), or Eliasmith (2013). These models attempt to develop cognitive-psychological functionalities from biological constructs. One may consider these approaches complementary to the present work. See Chapter 8 for a discussion regarding biological connections.

Ideally, one would want to compare Clarion with these models above and other models quantitatively through simulation, although in reality it is difficult. The difficulty may stem from the following causes: (1) many models focused on different issues at different levels, not directly relevant to Clarion; (2) some models were highly specialized, dealing only with one task or even one data set, and therefore it is difficult or inappropriate to compare them with Clarion, which is a generic cognitive architecture that is coarser by necessity (Anderson & Lebiere, 2003); (3) different models often dealt with different tasks and therefore are not directly comparable to each other. Nevertheless, in the previous work on Clarion, detailed comparisons of Clarion simulations with simulations conducted using other models were often included, when there was indeed overlapping coverage (see, e.g., Sun, 2002; Sun, Slusarz, & Terry, 2005; Sun et al., 2007, 2009).

For more comparisons of cognitive architectures, see Pew and Mavor (1998), Ritter et al. (2003), Sun (2006), Chong et al. (2007), Taatgen and Anderson (2008), Langley et al. (2009), Thórisson and Helgasson (2012), and Helie and Sun (2014b).

Finally, I should mention that Allen Newell proposed in the 1980s a set of criteria for a humanlike cognitive model (e.g., as summarized in Newell, 1990). These criteria include: (1) behaving as an arbitrary function of the environment; (2) operating in real time; (3) exhibiting rational, effective adaptive behavior; (4) using vast amounts of knowledge about the environment; (5) behaving robustly in the face of error, the unexpected, and the unknown; (6) integrating diverse knowledge; (7) using language; (8) exhibiting self-awareness and a sense of self; (9) learning from its environment; (10) acquiring capabilities through development; (11) arising through evolution; and (12) being realizable within the brain. Newell proposed these criteria as part of an effort to justify his work at that time in developing cognitive architectures (and Soar in particular). After more than 30 years, however, this set of criteria may seem rough, unspecific, and outdated. Many currently existing cognitive architectures, including Clarion, could claim to satisfy these criteria (or at least most of them), and there is often no objective way of determining the truth or falsity of these claims. Therefore, at this point, I would avoid using these criteria in comparing cognitive architectures or other cognitive models.[4]

4. Note that a detailed critique of these criteria would be out of place in this book and thus will have to be deferred to other work.

9.6. Future Directions

Finally, it would be pertinent to see what issues are still open at this point, and which of these are particularly suitable for approaching from within Clarion. It would also be useful to know what difficulties and challenges are still ahead in this regard.

9.6.1. Directions for Cognitive Social Simulation

I would like to address a number of specific directions with regard to cognitive social simulation. Among the specific topics tackled within the Clarion framework, cognitive social simulation is promising in terms of future development.

Cognitive social simulation, as discussed in Chapter 7, needs a great deal more work. It can be a precious source of new ideas and inspirations for further developing cognitive architectures. It helps to highlight the social aspects of human cognition-psychology; thus it may lead to better incorporation of these aspects into models of individual cognition-psychology and in particular into cognitive architectures. Traditional approaches to cognitive modeling have largely ignored the effects of social processes. By modeling individuals in a social context, one can learn more about the sociocultural processes that influence individual cognition-psychology (Zerubavel, 1997). Thus, integrating social simulation and cognitive modeling may well lead to better understanding of individual cognition-psychology.

On the other hand, a more realistic model of individual minds, incorporating realistic tendencies, inclinations, and capabilities, can serve as a more realistic basis for understanding social processes, even though currently most agent models in social simulation have been extremely simple. As has been argued before (Sun, 2001; Sun & Naveh, 2004; Castelfranchi, 2001), social processes ultimately rest on decisions and actions of individuals, and thus understanding the mechanisms and processes of individual cognition-psychology can lead to better theories describing aggregates of individuals. Compared with more specialized or narrowly scoped cognitive-psychological models, cognitive architectures certainly have some significant advantages in this regard due to their generality and comprehensiveness (Sun, 2006; Newell, 1990).

The most fundamental issue in this regard, as indicated before, is how to develop better ways of conducting detailed social simulation on the

basis of cognitive architectures as building blocks. This is not an easy task, although some initial work has been done (e.g., as discussed in Chapter 7; see also Sun, 2006). In this regard, one specific direction that has been mentioned before is enhancing cognitive architectures for the purpose of accounting for sociality in individuals. There are many questions to be asked in this regard: for example, what are the additional characteristics of a cognitive architecture for modeling the interaction of individuals (not just modeling individuals in isolation)? What additional representations, mechanisms, and processes within individuals are needed? Sun (2006) provided detailed discussions of these questions. Within the Clarion cognitive architecture, we need to further develop essential social-psychological capabilities necessary for cognitive social simulation, including motivation (especially socially oriented motivation), emotion (especially socially oriented emotion), personality (and its relation to social interaction), moral judgment (and its relation to social interaction), representation of self and others, social role, social identity, self categorization, and so on, as touched upon in previous chapters.

With Clarion, various sociocultural factors and their interactions with the cognitive-psychological factors need to be further explored. For example, one may explore cultural aspects through cognitive social simulation: exploring culture formation, propagation, and transformation (convergence, divergence, shifting boundaries, and so on) on the basis of cognition-psychology. With Clarion, how cognitive-psychological factors, along with sociocultural factors, physical environmental factors, and so on, lead to various forms of social institutions (and vice versa) may also be further explored for a better understanding of the interaction of sociality and cognition-psychology.

The role of social networks in social processes needs to be explored and related to cognitive-psychological factors. For instance, one may explore the relationship of social networks to motivational dynamics, personality formation and adaptation, moral belief formation, creative problem solving, and so on. One may conduct human experiments exploring effects of social networks on these aspects. Experiments may also explore effects on social networks of different individual motivations, personalities, moral beliefs, self-identities, and so on. Then these effects may be captured in computational forms (e.g., through Clarion) in cognitive social simulation.

Results of these research directions may be applied to a wide variety of psychological and social phenomena deepening our understanding

of these phenomena: for example, addiction of various types and forms (Sun, Wilson, & Mathews, 2011), personality and personality disorders (Sun & Wilson, 2014; Sun & Wilson, 2014b), social anxiety (Wilson, Sun, & Mathews, 2009), social stereotyping (Wilson et al., 2010), and so on.

However, there are also the issues of computational complexity and thus scalability of simulation that need to be addressed. Social simulation could involve a large number of individuals (often thousands of individuals). Computational complexity is thus already high, even without involving cognitive architectures as models of individuals. To incorporate cognitive architectures into social simulation, one has to deal with added complexity. Thus, scalability is a significant problem. Specialized, narrowly scoped cognitive models might be better at avoiding this problem, but they lack the generality and comprehensiveness that are attractive for cognitive social simulation in general. More work is needed in this regard.

Yet another important direction with regard to cognitive social simulation is exploring its theoretical potentials, for example, in terms of its role in coming up with cognitive-psychological explanations of social processes and phenomena and even the very notion of cognitive-psychological explanations of social processes and phenomena. In so doing, we need to address theoretical issues surrounding cognitive social simulation. These issues have been initially addressed in Sun (2012b), but much more work is needed.

9.6.2. Other Directions for Cognitive Architectures

I now turn to other directions of developing cognitive architectures. More comprehensive, more psychologically realistic, more algorithmically sophisticated cognitive architectures are to be developed, either through incremental improvements or through coming up with radical new ideas. Most likely, it will be an incremental, continuous process to improve upon the state of the art and to come up with cognitive architectures that better and better mirror the human mind and possibly serve a variety of application domains at the same time.

The understanding of the mind has always been driven by technological developments and in recent decades by the developments in computer science. To develop better cognitive architectures, better constituting computational methods and algorithms are needed, especially if one wants to scale up modeling and simulation. Relevant computational methods and algorithms are found in various subfields of computer

science and engineering (e.g., machine learning). It is important that computational researchers in these subfields come up with better methods and algorithms, for various functionalities such as perception, learning, memory, reasoning, decision making, problem solving, planning, language, and so on. On the basis of such developments, better cognitive architectures can be developed correspondingly.

In particular, better natural language processing capabilities, more extensive sensory-motor capabilities, better perception, more powerful learning algorithms, more efficient planning algorithms, and the like are needed. Each of these types of algorithms could potentially significantly improve cognitive architectures in terms of their psychological realism or their application potentials.

Better computational methods and algorithms for putting the pieces together to form better overall architectures should also be emphasized. Various pieces have been developed (e.g., neural networks, reinforcement learning, and so on) and are improving, so it is important to put them together to form a more coherent, better-integrated cognitive architecture that more accurately reflects human cognition-psychology. Better algorithms and computational methods are needed for this purpose.

Another direction that should be continued in future work, based on what has been done thus far, is to develop a cognitive architecture with as few parameters as possible while accounting for as large a variety of empirical tasks, observations, phenomena, and constructs as possible. In so doing, one needs to address the cost-benefit trade-off between complexity and capability. In the same vein, one should consider the issue that a model accounts for a large set of data, either because of its extreme generality or because it captures deep structures and regularities of the mind. Any cognitive model has to address these issues. More work is certainly needed in this regard.

Future work needs to address many more generic psychological "laws," beyond those discussed in Chapter 5. This direction is highly relevant to developing well-constrained but generic models balancing complexity and capability. Future work in accounting for psychological "laws" should expand the scope of the existing Clarion explanations of psychological "laws" to a larger set of psychological phenomena, although many phenomena have been accounted for (as sampled in Chapter 5, and as reported more extensively elsewhere, e.g., in Helie & Sun, 2014 and Sun & Helie, 2013). Deeper explorations of finer-grained details of psychological

"laws" should also be attempted. Such work should also be compared to other existing theories and models for these phenomena.

A difficult issue has been the validation of details of a computational model against empirical (e.g., psychological) data, especially for cognitive architectures. Painstakingly detailed work needs to be carried out before claims are made, especially generic claims about human cognition-psychology. The issue of validation poses a serious challenge for cognitive architectures, because of the myriad of mechanisms involved in a cognitive architecture and therefore the overall complexity. Detailed validation of cognitive architectures has been extremely difficult, much more so than the validation of simple, narrowly scoped models. Addressing this issue better is an important research direction, requiring much further work.

A related issue is the validation of cognitive social simulation. It has been pointed out that validation of complex social simulation models is extremely difficult (Sun, 2006). However, in this regard, adopting an existing cognitive architecture as part of a cognitive social simulation may be beneficial. If one adopts a well-established cognitive model, the prior validation of that cognitive model, to whatever extent it may exist, may be leveraged in validating the overall simulation.

9.6.3. Final Words on Future Directions

Look at work on cognitive architectures more broadly. Some have claimed that grand scientific theorizing has become a thing of the past. What remains to be done is the filling-in of minute details and the refining of relatively minor points. Fortunately, many others have believed otherwise. Researchers are often pursuing integrative approaches that attempt to explain data in multiple domains and functionalities. In cognitive science, as in many other scientific fields, significant advances may be made through hypothesizing, testing, and confirming deep-level principles that unify superficial explanations across multiple domains, in a way somewhat analogous to Einstein's theory that unified electromagnetic and gravitational forces or String Theory that provides even further unifications. Such unifying theories concerning the human mind (human cognition-psychology), on the basis of cognitive architectures, are what should be more strongly emphasized.

Comprehensive, integrative models, such as cognitive architectures, serve as antidotes to the increasing specialization of scientific research.

Cognitive architectures that integrate a broad range of functionalities go against this trend of increasing specialization and help to fit pieces together again. The trend of over-specialization could be harmful and a reversal of this trend may be a necessary step toward further advances of cognitive science and psychology (Sun, Honavar, & Oden 1999). Developing integrative cognitive architectures is thus a major challenge and a major opportunity.

It is of vital importance to continue to work toward the ultimate goal of fully understanding, explaining, and capturing integrated and functioning, biological, psychological, and social "personhood", which results from the sum total of the biological, psychological, social, cultural, and other factors, through the relationship between the biological being and the physical and social worlds—their interaction and co-evolvement, as accentuated by the ecological-functional perspective articulated in Chapter 1 (Sun, 2002; Sun, 2012). Although important steps toward this ultimate goal have been taken, there is still a very long way to go in this regard.

References

Ahlum-Heath, M. & DiVesta, F. (1986). The effect of conscious controlled verbalization of a cognitive strategy on transfer in problem solving. *Memory and Cognition, 14*, 281–285.

Alexander, J., Giesen, B., Munch, R., & Smelser, N. (Eds.). (1987). *The micro-macro link*. Berkeley, CA: University of California Press.

Anderson, J. R. (1983). *The architecture of cognition*. Cambridge, MA: Harvard University Press.

Anderson, J. R. (1993). *Rules of the mind*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.

Anderson, J. R. & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Lawrence Erlbaum Associates.

Anderson, J. R. & Lebiere, C. L. (2003). The Newell test for a theory of cognition. *Behavioral and Brain Science, 26*, 587–637.

Atran, S. & Norenzayan, A. (2004). Religion's evolutionary landscape: Counterintuition, commitment, compassion, communion. *Behavioral and Brain Sciences, 27*(6), 713–730.

Axelrod, R. (1984). *The evolution of cooperation*. New York: Basic Books.

Baddeley, A. (1986). *Working memory*. New York: Oxford University Press.

Bach, J. (2009). *Principles of synthetic intelligence PSI: An architecture of motivated cognition*. New York: Oxford University Press.

Bandura, A. (1997). *Self-efficacy: The exercise of control*. New York: Freeman.

Barkow, J., Cosmides, L., & Tooby, J. (1992). *The adapted mind: Evolutionary psychology and the generation of culture*. New York: Oxford University Press.

Baumeister, R. F. (1984). Choking under pressure: Self-consciousness and paradoxical effects of incentives on skillful performance. *Journal of Personality and Social Psychology*, *46*, 610–620.

Bechtel, W. (2003). Modules, brain parts, and evolutionary psychology. In S. J. Scher and F. Rauscher (Eds.), *Evolutionary psychology: Alternative approaches*. Dordrecht: Kluwer.

Beilock, S. & Carr, T. (2001). On the fragility of skilled performance: What governs choking under pressure? *Journal of Experimental Psychology: General*, *130*(4), 701–725.

Beilock, S., Kulp, C., Holt, L., & Carr, T. (2004), More on the fragility of performance: Choking under pressure in mathematical problem solving. *Journal of Experimental Psychology: General*, *133*(4), 584–600.

Bennis, W. M., Medin, D. L., & Bartels, D. M. (2010). The costs and benefits of calculation and moral rules. *Perspectives on Psychological Science*, *5*, 187–202.

Berry, D. (1983). Metacognitive experience and transfer of logical reasoning. *Quarterly Journal of Experimental Psychology*, *35*A, 39–49.

Berry, D. (1991). The role of action in implicit learning. *Quarterly Journal of Experimental Psychology*, *43*A, 881–906.

Berry, D. & Broadbent, D. (1984). On the relationship between task performance and associated verbalizable knowledge. *Quarterly Journal of Experimental Psychology*, *36*A, 209–231.

Berry, D. & Broadbent, D. (1988). Interactive tasks and the implicit-explicit distinction. *British Journal of Psychology*, *79*, 251–272.

Bertsekas, D. & Tsitsiklis, J. (1996). *Neuro-dynamic programming*. Belmont, MA: Athena Scientific.

Bickhard, M. (1993). Representational content in humans and machines. *Journal of Experimental and Theoretical Artificial Intelligence*, 285–333.

Bourdieu, P. & Wacquant, I. (1992). *An invitation to reflexive sociology*. Chicago: University of Chicago Press.

Bower, A. & King, W. (1967). The effect of number of irrelevant stimulus dimensions, verbalization, and sex on learning biconditional classification rules. *Psychonomic Science*, *8*(10), 453–454.

Bower, G. H. (1981). Mood and memory. *American Psychologist*, *36*, 129–148.

Bowers, J. S. (2009). On the biological plausibility of grandmother cells: Implications for neural network theories in psychology and neuroscience. *Psychological Review*, *116*, 220–251.

Bowers, K., Regehr, G., Balthazard, C., and Parker, K. (1990). Intuition in the context of discovery. *Cognitive Psychology*, *22*, 72–110.

Boyer, P. & Ramble, C. (2001). Cognitive templates for religious concepts: Cross-cultural evidence for recall of counter-intuitive representations. *Cognitive Science*, *25*, 535–564.

Braine, M. & O'Brien, D. (Eds.). (1998). *Mental logic*. Mahwah, NJ: Lawrence Erlbaum Associates.

Braitenberg, V. (1984). *Vehicles: Experiments in synthetic psychology*. Cambridge, MA: MIT Press.

Brooks, J. D., Wilson, N., & Sun, R. (2012). The effects of performance motivation: A computational exploration of a dynamic decision making task. *Proceedings of the First International Conference on Brain-Mind* (pp. 7–14). East Lansing, MI: BMI Press.

Brooks, R. (1991). Intelligence without representation. *Artificial Intelligence*, *47*, 139–160.

Bruner, J., Goodnow, J., & Austin, J. (1956). *A study of thinking*. New York: Wiley.

Buckner, R. L., Petersen, S. E., Ojemann, J. G., Miezin, F. M., Squire, L. R., & Raichle, M. E. (1995). Functional anatomical studies of explicit and implicit memory retrieval tasks. *Journal of Neuroscience*, *15*(1), 12–29.

Busemeyer, J. R. & Johnson, J. G. (2008). Micro-process models of decision making. In R. Sun (Ed.), *Cambridge handbook of computational psychology* (pp. 302–321). New York: Cambridge University Press.

Cacioppo, J. T., Gardner, W. L., & Berntson, G. G. (1999). The affect system has parallel and integrative processing components: Form follows function. *Journal of Personality and Social Psychology*, *76*, 839–855.

Caprara, G. V. and D. Cervone, (2000). *Personality: Determinants, Dynamics, and Potentials*. New York: Cambridge University Press.

Carley, K. M., M. J. Prietula, and Z. Lin, (1998). Design versus cognition: The interaction of agent cognition and organizational design on organizational performance. *Journal of Artificial Societies and Social Simulation*, *1*(3).

Carver, C. & Scheier, M. (1998). *On the self-regulation of behavior*. Cambridge, UK: Cambridge University Press.

Castelfranchi, C. (2001). The theory of social functions: challenges for computational social science and multi-agent learning. *Cognitive Systems Research*, *2*(1), 5–38.

Cecconi, F. and D. Parisi, (1998). Individual versus social survival strategies. *Journal of Artificial Societies and Social Simulation*, 1(2). http://www.soc.surrey.ac.uk/JASSS/1/2/1.html

Cervone, D. (2004). The architecture of personality. *Psychological Review*, *111*(1), 183–204.

Chaiken, S. & Trope, Y. (Eds.). (1999). *Dual process theories in social psychology*. New York: Guilford Press.

Chartier, S. and Proulx, R. (2005). NDRAM: A nonlinear dynamic recurrent associative memory for learning bipolar and nonbipolar correlated patterns. *IEEE Transactions on Neural Networks*, *16*, 1393–1400.

Chen, S., Shechter, D., & Chaiken, S. (1996). Getting at the truth or getting along: Accuracy- versus impression-motivated heuristic and systematic processing. *Journal of Personality and Social Psychology*, *71*(2), 262–275.

Chi, M., M. Bassok, M. Lewis, P. Reimann, and P. Glaser, (1989). Self-explanation: How students study and use examples in learning to solve problems. *Cognitive Science*, *13*, 145–182.

Chirkov, V. I., Ryan, R. M., & Willness, C. (2005). Cultural context and psychological needs in Canada and Brazil: Testing a self-determination approach to the internalization of cultural practices, identity, and well-being. *Journal of Cross-Cultural Psychology*, *36*, 423–443.

Chomsky, N. (1980). *Rules and representation*. New York: Columbia University Press.

Chong, H., Tan, A., & Ng, G. (2007). Integrated cognitive architectures: A survey. *Artificial Intelligence Rev*iew, *28*(2), 103–130.

Clancey, W. J. (1997). *Situated cognition: On human knowledge and computer representation*. New York: Cambridge University Press.

Clark, L. A., & Watson, D. (1999). Temperament: A new paradigm for trait psychology. In L. A. Pervin & O. P. John (Eds.), *Handbook of personality: Theory and research* (2d ed.) (pp. 399–423). New York: Guilford Press.

Cleeremans, A. (1997). Principles for implicit learning. In D. Berry (Ed.), *How implicit is implicit learning?* (pp. 195–234). Oxford: Oxford University Press.

Cleeremans, A., A. Destrebecqz & M. Boyer. (1998). Implicit learning: News from the front. *Trends in Cognitive Sciences*, *2*(10), 406–416.

Cleeremans, A. and J. McClelland, (1991). Learning the structure of event sequences. *Journal of Experimental Psychology: General*, *120*, 235–253.

Collins, A. (1978). Fragments of a theory of human plausible reasoning. In D. Waltz (Ed.), *Theoretical Issues in Natural Language Processing II*, 194–201. Norword, NJ: Ablex.

Collins, A. & Loftus, J. (1975). Spreading activation theory of semantic processing. *Psychological Review*, *82*, 407–428.

Collins, A. & Michalski, R. (1989). The logic of plausible reasoning. *Cognitive Science*, *13*(1), 1–49.

Collins, A. M., & Quillian, M. R. (1969). Retrieval time from semantic memory. *Journal of Verbal Behavior and Verbal Learning*, *8*, 432–438.

Cooper, R. P. (2007). The role of falsification in the development of cognitive architectures: Insights from a Lakatosian analysis. *Cognitive Science*, *31*, 509–533.

Cosmides, L. & Tooby, L. (1994). Beyond intuition and instinct blindness: Toward an evolutionarily rigorous cognitive science. *Cognition*, *50*, 41–77.

Curran, T. & Keele, S. (1993). Attention and structure in sequence learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *19*, 189–202.

Dai, D. Y. & Sun, R. (2012). Where is the unity of attention, representation, and performance? In S. Masmoudi, D. Y. Dai, & A. Naceur (Eds.), *Attention, representation, and human performance: Integration of cognition, emotion, and motivation* (pp. 217–233). London: Taylor & Francis.

Damasio, A. (1994). *Descartes' error: Emotion, reason and the human brain*. New York: Grosset/Putnam.

D'Andrade, R.G. & Strauss, C. (Eds.). (1992). *Human motives and cultural models*. Cambridge, UK: Cambridge University Press.

Deci, E. (1980). Intrinsic motivation and personality. In E. Staub (Ed.), *Personality: Basic issues and current research* (pp. 35–80). Englewood Cliffs, NJ: Prentice Hall.

Deci, E. L. & Ryan, R. M. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, *55*, 68–78.

de Quervain, D.J., Fischbacher, U., Treyer, V., Schellhammer, M., Schnyder, U., Buck, A., and Fehr, E. (2004). The neural basis of altruistic punishment. *Science*, *305*, 1254–1258.

Dewey, J. (1958). *Experience and nature*. New York: Dover.

Dienes, Z. and Fahey, R. (1995). Role of specific instances in controlling a dynamic system. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *21*(4), 848–862.

Digman, J. M. (1990). Personality structure: Emergence of the five-factor model. *Annual Review of Psychology*, *41*, 417–440.

Doerner, D. (2003). The mathematics of emotions. In F. Detje, D. Doerner, and H. Schaub (Eds.), *Proceedings of the Fifth International Conference on Cognitive Modeling* (pp. 75–79). Bamberg, Germany.

Domangue, T. J., Mathews, R. C., Sun, R., Roussel, L. G., & Guidry, C. E. (2004). Effects of model-based and memory-based processing on speed and accuracy of grammar string generation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *30*(5), 1002–1011.

Dominowski, R. (1972). How do people discover concepts? In R. L. Solso (Ed.), *Theories in cognitive psychology: The Loyola symposium* (pp. 257–288). Potomac, MD: Lawrence Erlbaum Associates.

Doran, J., Palmer, M., Gilbert, N., and Mellars, P. (1994). The EOS project: Modeling upper Paleolithic social change. In N. Gilbert and J. Doran (Eds.), *Simulating Societies* (pp. 195–221). London: UCL Press.

Dreyfus, H. (1992). *Being-in-the-world*. Cambridge, MA: MIT Press.

Dreyfus, H. & Dreyfus, S. (1987). *Mind over machine: The power of human intuition*. New York: Free Press.

Dunn, J. & Kirsner, K. (1988). Discovering functionally independent mental processes: the principle of reversed associations. *Psychological Review*, *95*, 21–101.

Durkheim, W. (1962). *The rules of the sociological method*. Glencoe, IL: Free Press. (Original work published in 1895.)

Dweck, C. S. (2008). Can personality be changed? The role of beliefs in personality and change. *Current Directions in Psychological Science, 17*(6), 391–394.

Ekman, P. (1999). Basic emotions. In Dalgleish, T. & Power, M. (Eds.), *Handbook of cognition and emotion*. Chichester, UK: John Wiley and Sons.

Eliasmith, C. (2013). *How to build a brain: A neural architecture for biological cognition*. New York: Oxford University Press.

Elliot, A. & Thrash, T. (2002). Approach-avoidance motivation in personality: Approach and avoidance temperaments and goals. *Journal of Personality and Social Psychology, 82*(5), 804–818.

Epstein, A. (1982). Instinct and motivation as explanations for complex behavior. In D. W. Pfaff (Ed.), *The physiological mechanisms of motivation*. Berlin: SpringerVerlag.

Erickson, M. & Kruschke, J. (1998). Rules and exemplars in category learning. *Journal of Experimental Psychology: General, 127*, 107–140.

Estes, W. (1986). Memory storage and retrieval processes in category learning. *Journal of Experimental Psychology: General, 115*, 155–174.

Evans, J. (2003). In two minds: Dual-process accounts of reasoning. *Trends in Cognitive Sciences, 7*(10), 454–459.

Evans, J. & Frankish, K. (Eds.). (2009). *In two minds: Dual processes and beyond*. Oxford: Oxford University Press.

Fehr, E. & Gintis, H. (2007). Human motivation and social cooperation: Experimental and analytical foundations. *Annual Review of Sociology, 33*, 43–64.

Flavell, J. (1976). Metacognitive aspects of problem solving. In B. Resnick (Ed.), *The nature of intelligence*. Hillsdale, NJ: Erlbaum Associates.

Fodor, J. (1983). *The modularity of mind*. Cambridge, MA: MIT Press.

Foot, P. (1967). The problem of abortion and the doctrine of double effect. *Oxford Review, 5*, 5–15.

Frijda, N. (1986). *The emotions*. New York: Cambridge University Press.

Fum, D., Del Missier, F., & Stocco, A. (2007). The cognitive modeling of human behavior: Why a model is (sometimes) better than 10,000 words. *Cognitive Systems Research, 8*, 135–142.

Fum, D. & Stocco, A. (2003). Instance vs. rule based learning in controlling a dynamic system. In *Proceeedings of the Fifth International Conference on Cognitive Modelling* (pp. 105–110). Bamberg, Germany.

Gagne, R. & Smith, E. (1962). A study of the effects of verbalization on problem solving. *Journal of Experimental Psychology, 63*, 12–18.

Garnham, A. & Oakhill, J.V. (1994). *Thinking and reasoning*. Oxford: Blackwell.

Gathercole, S. (2003). *Short-term and working memory*. London: Taylor and Francis.

Gentner, D., & Collins, A. (1981). Studies of inference from lack of knowledge. *Memory and Cognition*, *9*, 434–443.

Gick, M. & Holyoak, K. (1980). Analogical problem solving. *Cognitive Psychology*, *12*, 306–355.

Giddens, A. (1984). *The constitution of society*. Cambridge, UK: Polity Press.

Gigerenzer, G., Todd, P. M., & ABC Group. (1999). *Simple heuristics that make us smart*. New York: Oxford University Press.

Gilbert, N. (1997). A simulation of the structure of academic science. *Sociological Research Online*, *2*(2). http://www.socresonline.org.uk/socresonline/2/2/3.html.

Gilbert, N., den Besten, M., Bontovics, A., Craenen, B. G. W., Divina, F., Eiben, A. E. (2006). Emerging artificial societies through learning. *Journal of Artificial Societies and Social Simulation*, *9*(2). http://jasss.soc.surrey.ac.uk/9/2/9.html.

Gilbert, N. & Doran, J. (1994). *Simulating societies: The computer simulation of social phenomena*. London: UCL Press.

Glenberg, A., Wilkinson, A., and Epstein, W. (1982). The illusion of knowing: Failure in the self assessment of comprehension. *Memory and Cognition*, *10*, 597–602.

Goel, V., Bruchel, C., Frith C., & Dolan, R. (2000). Dissociation of mechanisms underlying syllogistic reasoning. *Neuroimage, 12*(5), 504–514.

Gratch, J., & Marsella, S. (2004). A domain-independent framework for modeling emotion. *Cognitive Systems Research*, *5*(4), 269–306.

Gray, J. A. (1987). Perspectives on anxiety and impulsivity: A commentary. *Journal of Research in Personality*, *21*(4), 493–509.

Gray, J. A. & McNaughton, N. (2000). *The neuropsychology of anxiety: An enquiry into the functions of the septo-hippocampal system* (2d ed.). New York: Oxford University Press.

Greene, J. D., Cushman, F. A., Stewart, L. E., Lowenberg, K., Nystrom, L. E., & Cohen, J. D. (2009). Pushing moral buttons: The interaction between personal force and intention in moral judgment. *Cognition*, *111*(3), 364–371.

Greene, J., Morelli, S., Lowenberg, K., Nystrom, L., & Cohen, J. (2008). Cognitive load selectively interferes with utilitarian moral judgment. *Cognition*, *107*(3), 1144–1154.

Gross, J. J. (Ed.). (2007). *Handbook of emotion regulation*. New York: Guilford Press.

Grossberg, S. (1976), Adaptive pattern classification and universal recoding: I. parallel development and coding of neural feature detectors. *Biological Cybernetics*, *23*, 121–134.

Grossberg, S. (1982). *Studies of mind and brain: Neural principles of learning, perception, development, cognition, and motor control*. Norwell, MA: Kluwer Academic Publishers.

Grossberg, S. (1988). Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks*, *1*, 17–61.

Gyurak, A., Gross, J. J., & A. Etkin. (2011). Explicit and implicit emotion regulation: A dual-process framework. *Cognition and Emotion*, *25*(3), 400–412.

Hardy, L. & Parfitt, G. (1991). A catastrophe model of anxiety and performance. *British Journal of Psychology*, *82*(2), 163–178.

Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear phenomena*, *42*, 335–346.

Hasher, J. & Zacks, J. (1979). Automatic and effortful processes in memory. *Journal of Experimental Psychology: General*, *108*, 356–358.

Heidegger, M. (1927). *Being and time*. New York: Harper and Row, 1962.

Heit, E. (2008). Models of inductive reasoning. In R. Sun (Ed.), *Cambridge handbook of computational psychology* (pp. 322–338). New York: Cambridge University Press.

Helie, S., Chartier, S., & Proulx, R. (2006). Are unsupervised neural networks ignorant? Sizing the effect of environmental distributions on unsupervised learning. *Cognitive Systems Research*, *7*, 357–371.

Helie, S., Roeder, J. L., & Ashby, F. G. (2010). Evidence for cortical automaticity in rule-based categorization. *Journal of Neuroscience*, *30*(42), 14225–14234.

Helie, S., Proulx, R., & Lefebvre, B. (2011). Bottom-up learning of explicit knowledge using a Bayesian algorithm and a new Hebbian learning rule. *Neural Networks, 24*(3), 219–232.

Helie, S. & Sun, R. (2010). Incubation, insight, and creative problem solving: A unified theory and a connectionist model. *Psychological Review*, *117*(3), 994–1024.

Helie, S. & Sun, R. (2010b). Creative problem solving: A Clarion theory. *Proceedings of the 2010 International Joint Conference on Neural Networks*, Barcelona, Spain. pp. 1460–1466. Piscataway, NJ: IEEE Press.

Helie, S. & Sun, R. (2014). An integrative account of memory and reasoning phenomena. *New Ideas in Psychology*, *35*, 36–52.

Helie, S. & Sun, R. (2014b). Autonomous learning in psychologically-oriented cognitive architectures: A survey. *New Ideas in Psychology*, *34*, 37–55.

Henrich, J., Heine, S. J., & Norenzayan, A. (2010). Beyond WEIRD: Towards a broad-based behavioral science. *Behavioral and Brain Sciences*, *33*(2–3), 111–135.

Higgins E. T. (1997). Beyond pleasure and pain. *American Psychologist, 52*(12), 1280–1300.

Hintzman, D. (1990). Human learning and memory: Connections and dissociations. *Annual Review of Psychology*, *41*, 109–139.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, *79*, 2554–2558.

Huang, J. & Bargh, J. (2014). The selfish goal: Autonomously operating motivational structures as the proximate cause of human judgment and behavior. *Behavioral and Brain Sciences*, *37*, 121–175.

Hull, C. (1943). *Principles of behavior: An introduction to behavior theory*. New York: D. Appleton-Century Company.

Hull, C. (1951). *Essentials of behavior*. New Haven, CT: Yale University Press.

Humphreys, M. S. & Revelle, W. (1984). Personality, motivation, and performance: A theory of the relationship between individual differences and information processing. *Psychological Review*, *91*(2), 153–184.

Jacoby, L. (1983). Perceptual enhancement: persistent effects of an experience. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *9*(1), 21–38.

James, W. (1890). *The principles of psychology*. New York: Dover.

John, O. P. & Srivastava, S., (1999). The Big Five trait taxonomy: history, measurement, and theoretical perspectives. In L. A. Pervin & O. P. John (Eds.), *Handbook of personality: Theory and research (2nd edition)* (pp. 102–138). New York: Guilford Press.

Johnson, M. (1987). *The body in the mind: The bodily basis of imagination, reason, and meaning*. Chicago: University of Chicago Press.

Johnson, T. (1998). Acquisition and transfer of declarative and procedural knowledge. *Proceedings of the European Conference on Cognitive Modeling* (pp. 15–22). Nottingham, UK: Nottingham University Press.

Johnson-Laird, P.N., and Yang, Y., (2008). Mental logic, mental models, and computer simulations of human reasoning. In R. Sun (Ed.), *Cambridge handbook of computational psychology*. New York: Cambridge University Press.

Kahneman, D. (2003). A perspective on judgment and choice: Mapping bounded rationality. *American Psychologist*, *58*(9), 697–720.

Kanfer, R. & Ackerman, P. L. (1989). Motivation and cognitive abilities: An integrative/aptitude-treatment interaction approach to skill acquisition. *Journal of Applied Psychology*, *74*(4), 657–690.

Karmiloff-Smith, A. (1986). From meta-processes to conscious access: Evidence from children's metalinguistic and repair data. *Cognition*, *23*, 95–147.

Keil, F. (1989). *Concepts, kinds, and cognitive development*. Cambridge, MA: MIT Press.

Kennedy, W. G., & Bugajska, M. (2010). Integrating fast and slow cognitive processes. In D.D. Salvucci & G. Gunzelmann (Eds.), *Proceedings of the International Conference on Cognitive Modeling* (ICCM 2010) (pp. 121–126). Philadelphia, PA: Drexel University.

Kent, G. H., & Rosanoff, A. J. (1910). A study of association in insanity. *American Journal of Psychiatry*, *67*, 317–390.

Kirsh, D. (1990). When is information explicitly represented. In P. Hanson (Ed.), *Information, language, and cognition*. Vancouver: University of British Columbia Press.

Klein, S., Cosmides, L., Tooby, J., and Chance, S. (2002). Decisions and the evolution of memory: Multiple systems, multiple functions. *Psychological Review*, *109*(2), 306–329.

Kluver, J., Schmidt, J., & Stoica, C. (2005). The emergence of social order by processes of typifying: A computational model. *Journal of Mathematical Sociology*, *29*, 155–176.

Koch, C. (2011 March/April). Being John Malkovich. *Scientific American Mind*, 18–19.

Kuhn, T. (1970). *Structure of scientific revolutions*. Chicago: University of Chicago Press.

Laird, J. (2012). *The Soar cognitive architecture*. Cambridge, MA: MIT Press.

Lakatos, I. (1970). Falsification and methodology of research programs. In I. Lakatos & A. Musgrave (Eds.), *Criticism and the growth of knowledge*. Cambridge, UK: Cambridge University Press.

Lambert, A., Payne, B., Jacoby, L., Shaffer, L., Chasteen, A., & Khan, S. (2003). Stereotypes as dominant responses: On the "social facilitation" of prejudice in anticipated public contexts. *Journal of Personality and Social Psychology*, *84*(2), 277–295.

Lane, S., Mathews, R., Sallas, B., Prattini, R., & Sun, R. (2008). Facilitative interactions of model- and experience-based processes: Implications for type and flexibility of representation. *Memory and Cognition*, *36*(1), 157–169.

Langley, P. A., Laird, J. E. B., & Rogers, S.A. (2009). Cognitive architectures: Research issues and challenges. *Cognitive Systems Research*, *10*(2), 141–160.

Lavrac, N. & Dzeroski, S. (1994). *Inductive logic programming*. New York: Ellis Horword.

Lazarus, R. S., & Folkman, S. (1984). *Stress, appraisal, and coping*. New York: Springer.

LeDoux, J. (1996). *The emotional brain*. New York: Simon and Schuster.

Leven, S. & Levine, D. S. (1996). Multi-attribute decision making in context: A dynamic neural network methodology. *Cognitive Science*, *20*, 271–299.

Levine, D. S. (2000). *Introduction to neural and cognitive modeling*. Mahwah, NJ: Erlbaum Associates.

Lewicki, P. (1986). Processing information about covariations that cannot be articulated. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *12*, 135–146.

Lewicki, P., Czyzewska, M., & Hoffman, H. (1987). Unconscious acquisition of complex procedural knowledge. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *13*(4), 523–530.

Lewis, B. & Linder, D. (1997). Thinking about choking: Attentional processes and paradoxical performance. *Personality and Social Psychology Bulletin*, 23, 937–944.

Li, M., & Vitanyi, P. (1997). *An introduction to Kolmogorov complexity and its applications*. Heidelberg, Germany: Springer.

Libet, B. (1985). Unconscious cerebral initiative and the role of conscious will in voluntary action. *Behavioral and Brain Sciences*, *8*, 529–566.

Licato, J., Sun, R., & Bringsjord, S. (2014). Using a hybrid cognitive architecture to model children's errors in an analogy task. *Proceedings of the Annual Conference of Cognitive Science Society*, Quebec City, Quebec. (pp. 857–862). Austin, TX: Cognitive Science Society.

Licato, J., Sun, R., & Bringsjord, S. (2014b). Structural representation and reasoning in a hybrid cognitive architecture. *Proceedings of the 2014 International Joint Conference on Neural Networks*. Piscataway, NJ: IEEE Press.

Licklider, J. C. R. (1960). Man-computer symbiosis. *IRE Transactions on Human Factors in Electronics*, HFE-1, 4–11.

Lieberman, M. D. (2009). What zombies can't do: A social cognitive neuroscience approach to the irreducibility of reflective consciousness. In J. St. B. T. Evans & K. Frankish (Eds.), *In two minds: Dual processes and beyond* (pp. 293–316). Oxford: Oxford University Press.

Locke, E. A. & Latham, G. P. (1990). *A theory of goal setting and task performance*. Englewood Cliffs, NJ: Prentice Hall,

Locke, E. A. & Latham, G. P. (2002). Building a practically useful theory of goal setting and task motivation: A 35-year odyssey. *American Psychologist*, *57*(9), 705–717.

Logan, G. (1988). Toward an instance theory of automatization. *Psychological Review*, *95*(4), 492–527.

López, F. J. & Shanks, D. R. (2008). Models of animal learning and their relations to human learning. In R. Sun (Ed.), *Cambridge handbook of computational psychology*. New York: Cambridge University Press.

Lotka, A. J. (1926). The frequency distribution of scientific productivity. *Journal of the Washington Academy of Sciences*, *16*, 317–323.

Luce, R. D. (1959). *Individual choice behavior: A theoretical analysis*. New York: Wiley.

Lustick, I. (2000). Agent-based modeling of collective identity: Testing constructivist theory. *Journal of Artificial Society and Social Simulation*, *3*(1). http://www.soc.surrey.ac.uk/JASSS/3/1/1.html

Mandler, J. (1992). How to build a baby. *Psychological Review*, *99*(4), 587–604.

Maner, J. K., Kenrick, D. T., Neuberg, S. L., Becker, D. V., Robertson, T., Hofer, B., … Schaller, M. (2005). Functional projection: How fundamental social motives can bias interpersonal perception. *Journal of Personality and Social Psychology*, *88*, 63–78.

Marcel, A. J. (1983). Conscious and unconscious perception: An approach to the relations between phenomenal experience and perceptual processes. *Cognitive Psychology*, *15*, 238–300.

Marinier, R. P., Laird, J. E., & Lewis, R. L. (2009). A computational unification of cognitive behavior and emotion. *Cognitive Systems Research*, *10*(1), 48–69.

Markman, A. B. & Maddox, W. T. (2005). The implications of advances in research on motivation for cognitive models. *Journal of Experimental and Theoretical Artificial Intelligence*, *17*, 371–384.

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information*. New York: W. H. Freeman.

Marsella, S. & Gratch, J. (2009). EMA: A process model of appraisal dynamics. *Cognitive Systems Research*, *10*(1), 70–90.

Maslow, A. (1943). A theory of human motivation. *Psychological Review*, *50*, 370–396.

Maslow, A. (1987). *Motivation and personality*. 3d ed. New York: Harper and Row.

Masmoudi, S., D. Y. Dai, & A. Naceur (Eds.). (2012). *Attention, representation, and human performance: Integration of cognition, emotion, and motivation*. London: Taylor & Francis.

Massaro, D. (1988). Some criticisms of connectionist models of human performance. *Journal of Memory and Language*, *27*, 213–234.

Mathews, R., Buss, R., Stanley, W., Blanchard-Fields, F., Cho, J., & Druhan, B. (1989). Role of implicit and explicit processes in learning from examples: A synergistic effect. *Journal of Experimental Psychology: Learning, Memory and Cognition*. 15, 1083–1100.

Mathews, R., Tall, J., Lane, S. M., & Sun, R. (2011). Getting it right generally, but not precisely: Learning the relation between multiple inputs and outputs. *Memory and Cognition*, *39*(6), 1133–1145.

Mayer, J. D. (2005). Tale of two visions: Can a new view of personality help integrate psychology? *American Psychologist*, *60*(4), 294–307.

Mazzoni, G. & T. Nelson (Eds.). (1998). *Metacognition and cognitive neuropsychology*. Mahwah, NJ: Erlbaum Associates.

McClelland, J., McNaughton, B., & O'Reilly, R. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, *102*(3), 419–457.

McCrae, R. R. (2002). Cross-cultural research on the five-factor model of personality. In W. J. Lonner, D. L. Dinnel, S. A. Hayes, & D. N. Sattler (Eds.), *Online readings in psychology and culture*. Bellingham, WA: Center for Cross-Cultural Research, Western Washington University.

McCrae, R. R., & Costa, P. T. Jr. (2010). *NEO inventories: Professional manual*. Lutz, FL: Psychological Assessment Resources.

McDougall, W. (1936). *An introduction to social psychology*. London: Methuen & Co.

McFarland, D. (1989). *Problems of animal behaviour*. New York: Longman.

Merikle, P. M., & Daneman, M. (1998). Psychological investigations of unconscious perception. *Journal of Consciousness Studies*, *5*, 5–18.

Merleau-Ponty, M. (1963). *The structure of behavior*. Boston: Beacon Press.

Metcalfe, J. (1986). Dynamic metacognitive monitoring during problem solving. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *12*, 623–634.

Meyer, D. & Kieras, D. (1997). A computational theory of executive cognitive processes and human multiple-task performance: Part 1, basic mechanisms. *Psychological Review*, *104*(1), 3–65.

Miikkulainen, R. (1993). *Subsymbolic natural language processing: An integrated model of scripts, lexicon, and memory*. Cambridge, MA: MIT Press.

Mikhail, J. (2007). Universal moral grammar: Theory, evidence and the future. *Trends in Cognitive Sciences*, *11*(4), 143–152.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, *63*, 81–97.

Miller, G. A., Galanter, E., & Pribram, K. H. (1960). *Plans and the structure of behavior*. New York: Holt.

Milner, D., and Goodale, M. (1995). *The visual brain in action*. Oxford: Oxford University Press.

Mineka, S. & Sutton, S. (1992). Cognitive biases and the emotional disorders. *Psychological Science*, *3*(1), 65–69.

Minsky, M. (1985). *The society of mind*. New York: Simon and Schuster.

Monroe, K. (2012). Cognition and moral choice. In R. Sun (Ed.), *Grounding social sciences in cognitive sciences* (pp.183–206). Cambridge, MA: MIT Press.

Montague, P. R. (1999). Review of reinforcement learning: An introduction. *Trends in Cognitive Science, 3*(9), 360–361.

Montgomery, K. J., Seeherman, K. R., & Haxby, J. V. (2009). The well-tempered social brain. *Psychological Science*, *20*(10), 1211–1213.

Moscovitch, M. & Umilta, C. (1991). Conscious and unconscious aspects of memory: A neuropsychological framework of modules and central systems. In R. Lister & H. Weingartner (Eds.), *Perspectives on cognitive neuroscience*. New York: Oxford University Press.

Moskowitz, D. S., Suh, E. J., and Desaulniers, J. (1994). Situational influences on gender differences in agency and communion. *Journal of Personality and Social Psychology*, *66*, 753–761.

Murray, H. (1938). *Explorations in personality*. New York: Oxford University Press.

Naveh, I. and Sun, R. (2006). A cognitively based simulation of academic science. *Computational and Mathematical Organization Theory*, *12*(4), 313–337.

Nelson, D., McKinney, V., Gee, N., and Janczura, G. (1998). Interpreting the influence of implicitly activated memories on recall and recognition. *Psychological Review*, *105*(2), 299–324.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

Newell, A., & Simon, H. A. (1976). Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, *19*(3), 113–126.

Nisbett, R. & Wilson, T. (1977). Telling more than we can know: Verbal reports on mental processes. *Psychological Review*, *84*(3), 231–259.

Nokes, T. J., & Ohlsson, S. (2001). How is abstract generative knowledge acquired? A comparison of three learning scenarios. In J. D. Moore and K. Stenning (Eds.), *Proceedings of the Twenty Third Annual Conference of the Cognitive Science Society* (pp. 710–715). Mahwah, NJ: Erlbaum Associates.

Norman, D. & Shallice, T. (1986). Attention to action: Willed and automatic control of behavior. In G. Schwartz & D. Shapiro (Eds.), *Consciousness and self regulation: Advances in research and theory* (pp. 1–18). New York: Plenum.

Norman, K., Detre, G., & Polyn, S. (2008). Computational models of episodic memory. In R. Sun (Ed.), *Cambridge handbook on computational psychology* (pp. 189–225). New York: Cambridge University Press.

Norton, M., Vandello, J., & Darley, J. (2004). Casuistry and social category bias. *Journal of Personality and Social Psychology*, *87*(6), 817–831.

O'Reilly, R. C. & Munakata, Y. (2000). *Computational explorations in cognitive neuroscience: Understanding the mind by simulating the brain*. Cambridge, MA: MIT Press.

Ortony, A., Clore, G., & Collins, A. (1988). *The cognitive structure of emotions*. Cambridge, UK: Cambridge University Press.

Osherson, D.N., Smith, E. E., Wilkie, O., Lopez, A., & Shafir, E. (1990). Category-based induction. *Psychological Review*, *97*, 185–200.

Osman, M. (2010). Controlling uncertainty: A review of human behavior in complex dynamic environments. *Psychological Bulletin*, *136*(1), 65–86.

Over, H. & Carpenter, M. (2009). Eighteen-month-old infants show increased helping following priming with affiliation. *Psychological Science*, *20*, 1189–1193.

Pew, R. W. & Mavor, A. S. (Eds.). (1998). *Modeling human and organizational behavior: Application to military simulations*. Washington, DC: National Academy Press.

Posner, M., DiGirolamo, G., & Fernandez-Duque, D. (1997). Brain mechanisms of cognitive skills. *Consciousness and Cognition*, *6*, 267–290.

Proctor, R. & Dutta, A. (1995). *Skill acquisition and human performance*. Thousand Oaks, CA: SAGE.

Quek, M., & Moskowitz, D. S. (2007). Testing neural network models of personality. *Journal of Research in Personality*, *41*, 700–706.

Quillian, M. R. (1968). Semantic memory. In M. Minsky (Ed.), *Semantic information processing* (pp. 227–270). Cambridge, MA: MIT Press.

Rabinowitz, M. & Goldberg, N. (1995). Evaluating the structure-process hypothesis. In F. Weinert & W. Schneider (Eds.), *Memory performance and competencies*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Rao, A. S. & Georgeff, M. P. (1991). Modeling rational agents within a BDI-architecture. In J. Allen, R. Fikes, & E. Sandewall (Eds.), *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning* (pp. 473–484). San Mateo: Morgan Kaufmann Publishers.

Read, S. J., Monroe, B. M., Brownstein, A. L., Yang, Y., Chopra, G., and Miller, L. C. (2010). Virtual personalities II: A neural network model of the structure and dynamics of human personality. *Psychological Review*, *117*, 61–92.

Reber, A. (1989). Implicit learning and tacit knowledge. *Journal of Experimental Psychology: General*, *118*(3), 219–235.

Reber, A. & Allen, R. (1978). Analogy and abstraction strategies in synthetic grammar learning: A functionalist interpretation. *Cognition*, *6*, 189–221.

Reber, A., Kassin, S., Lewis, S., & Cantor, G. (1980). On the relationship between implicit and explicit modes in the learning of a complex rule structure. *Journal of Experimental Psychology: Human Learning and Memory*, *6*, 492–502.

Reber, A. & Lewis, S. (1977). Toward a theory of implicit learning: The analysis of the form and structure of a body of tacit knowledge. *Cognition*, *5*, 333–361.

Reder, L. (Ed.). (1996). *Implicit memory and metacognition*. Mahwah, NJ: Erlbaum Associates.

Reder, L. & Schunn, C. (1996). Metacognition does not imply awareness: Strategy choice is governed by implicit learning and memory. In L. Reder (Ed.), *Implicit memory and metacognition*. Mahwah, NJ: Erlbaum Associates.

Reisenzein, R. (2009). Emotions as metarepresentational states of mind: Naturalizing the belief-desire theory of emotion. *Cognitive Systems Research*, *10*(1), 6–20.

Reiss, S. (2004). Multifaceted nature of intrinsic motivation: The theory of 16 basic desires. *Review of General Psychology*, *8*(3), 179–193.

Reiss, S. (2008). *The normal personality: A new way of thinking about people*. New York: Cambridge University Press.

Reynolds, R. (1994). Learning to co-operate using cultural algorithms. In N. Gilbert and J. Doran (Eds.), *Simulating societies: The computer simulation of social phenomena*. London: UCL Press.

Rips, L. J. (1975). Inductive judgments about mental categories. *Journal of Verbal Learning and Verbal Behavior*, *14*, 665–681.

Rips, L. (1994). *The psychology of proof*. Cambridge, MA: MIT Press.

Ritter, F. E. & Bibby, P. A. (2008). Modeling how, when, and what is learned in a simple fault-finding task. *Cognitive Science, 32*(5), 862–892.

Ritter, F., Shadbolt, N., Elliman, D., Young, R., Gobet, F., & Baxter, G. (2003). *Techniques for modeling human performance in synthetic environments: A supplementary review*. Dayton, OH: Human Systems Information Analysis Center, Wright-Patterson Air Force Base.

Roberts, S. & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review*, *107*(2), 358–367.

Roediger, H. (1990). Implicit memory: Retention without remembering. *American Psychologist*, *45*(9), 1043–1056.

Rogers, T. (2008). Computational models of semantic memory. In R. Sun (Ed.), *Cambridge handbook on computational psychology* (pp. 226–266). New York: Cambridge University Press.

Rosenbaum, D., Carlson, R., & Gilmore, R. (2001). Acquisition of intellectual and perceptual-motor skills. *Annual Review of Psychology*, *52*, 453–470.

Rosenbloom, P., Laird, J., & Newell, A. (1993). *The SOAR papers: Research on integrated intelligence*. Cambridge, MA: MIT Press.

Rumelhart, D., McClelland, J., & PDP Research Group. (1986). *Parallel distributed processing: Explorations in the microstructures of cognition*. Cambridge, MA: MIT Press.

Samuel, D.B., & Widiger, T.A. (2008). A meta-analytic review of the relationships between the five-factor model and DSM-IV-TR personality disorders: A facet level analysis. *Clinical Psychology Review*, *28*(8), 1326–1342.

Sawyer, K. (2003). Artificial societies: Multiagent systems and the micro-macro link in sociological theory. *Sociological Methods & Research*, *31*(3), 325–363.

Schacter, D. (1987). Implicit memory: History and current status. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *13*, 501–518.

Schacter, D. (1990). Toward a cognitive neuropsychology of awareness: Implicit knowledge and anosagnosia. *Journal of Clinical and Experimental Neuropsychology*, *12*(1), 155–178.

Schelling, T. C. (1971). Dynamic models of segregation. *Journal of Mathematical Sociology*, *1*, 143–186.

Schmidhuber, J. (2014). Deep learning in neural networks: An Overview. http://www.idsia.ch/~juergen/deep-learning-overview.html.

Schooler, J. W., Ohlsson, S., & Brooks, K. (1993). Thoughts beyond words: When language overshadows insight. *Journal of Experimental Psychology: General*, *122*, 166–183.

Schopenhauer, A. (1819). *The world as will and representation*. Translated by E. F. J. Payne. New York: Dover Publications, 1969.

Schutz, A. (1967). *The phenomenology of the social world*. Evanston, IL: Northwestern University Press.

Schwartz, S. (1994). Are there universal aspects of human values? *Journal of Social Issues*, *50*, 19–45.

Seger, C. (1994). Implicit learning. *Psychological Bulletin*, *115*(2), 163–196.

Shanks, D. & St. John, M. (1994). Characteristics of dissociable learning systems. *Behavioral and Brain Sciences*, *17*, 367–394.

Sheldon, K. M. (2011). Integrating behavioral-motive and experiential-requirement perspectives on psychological needs: A two process model. *Psychological Review, 118*(4), 552–569.

Shoda, Y., & Mischel, W. (1998). Personality as a stable cognitive–affective activationnetwork: Characteristic patterns of behavior variation emerge from a stable personality structure. In S. J. Read & L. C. Miller (Eds.), *Connectionist models of social reasoning and social behavior* (pp. 175–208). Mahwah, NJ: Lawrence Erlbaum Associates.

Shultz, T. R., & Sirois, S. (2008). Computational models of developmental psychology. In R. Sun (Ed.), *The Cambridge handbook of computational psychology* (pp. 451–476). New York: Cambridge University Press.

Siegler, R. & Stern, E. (1998). Conscious and unconscious strategy discovery: A microgenetic analysis. *Journal of Experimental Psychology: General, 127*(4), 377–397.

Simon, H.A. (1957). *Models of man, social and rational*. New York: Wiley.

Simon, H. A. (1967). Motivational and emotional controls of cognition. *Psychological Review*, *74*, 29–39.

Sloman, S. (1993). Feature based induction. *Cognitive Psychology*, *25*, 231–280.

Sloman, S. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin*, *119*, 3–22.

Sloman, S. (1998). Categorical inference is not a tree: The myth of inheritance hierarchies. *Cognitive Psychology*, *35*, 1–33.

Smillie, L. D., Pickering, A. D., & Jackson, C. J. (2006). The new Reinforcement Sensitivity Theory: Implications for personality measurement. *Personality and Social Psychology Review*, *10*, 320–335.

Smith, C. A., & Lazarus, R. (1990). Emotion and adaptation. In L. A. Pervin (Ed.), *Handbook of personality: Theory & research* (pp. 609–637). New York: Guilford Press.

Smith, E. & Medin, D. (1981). *Categories and concepts*. Cambridge, MA: Harvard University Press.

Smolensky. P. (1988). On the proper treatment of connectionism, *Behavioral and Brain Sciences*, *11*, 1–43.

Squire, L. (1987). *Memory and brain*. New York: Oxford University Press.

Squire, L., & Frambach, M. (1990). Cognitive skill learning in amnesia. *Psychobiology*, *18*, 109–117.

Stadler, M. A. (1995). Role of attention in implicit learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *21*, 674–685.

Stadler, M. & Frensch, P. (1998). *Handbook of implicit learning*. Thousand Oaks, CA: SAGE.

Stanley, W., Mathews, R., Buss, R., & Kotler-Cope, S. (1989). Insight without awareness: On the interaction of verbalization, instruction and practice in a simulated process control task. *Quarterly Journal of Experimental Psychology*. *41*A (3), 553–577.

Strack, F. & Deutsch, R. (2005). Reflection and impulse as determinants of conscious and unconscious motivation. In J. Forgas, K. Williams, and S. Laham (Eds.), *Social motivation: Conscious and unconscious processes*. New York: Cambridge University Press.

Suh, E. J., D. S.Moskowitz, M. Fournier, and D. C. Zuroff (2004). Gender and relationships: Influences on agentic and communal behaviors. *Personal Relationships*, *11*, 41–59.

Sun, R. (1991). Connectionist models of rule-based reasoning. *Proceedings of the 13th Cognitive Science Conference* (pp. 437–442). Hillsdale, NJ: Lawrence Erlbaum Associates.

Sun, R. (1993). An efficient feature-based connectionist inheritance scheme. *IEEE Transactions on System, Man, and Cybernetics*, *23*(1), 23–54.

Sun, R. (1994). *Integrating rules and connectionism for robust commonsense reasoning*. New York: Wiley.

Sun, R. (1995). Robust reasoning: Integrating rule-based and similarity-based reasoning. *Artificial Intelligence*, *75*(2), 241–296.

Sun, R. (1995b). A microfeature-based approach toward metaphor interpretation. *Proceedings of the International Joint Conference on Artificial Intelligence* (IJCAI-95) (pp. 424–430). San Francisco, CA: Morgan Kaufmann.

Sun, R. (1999). Accounting for the computational basis of consciousness: A connectionist approach. *Consciousness and Cognition*, *8*, 529–565.

Sun, R. (2000). Symbol grounding: A new look at an old issue. *Philosophical Psychology*, *13*(3), 403–418.

Sun, R. (2001). Cognitive science meets multi-agent systems: A prolegomenon. *Philosophical Psychology, 14*(1), 5–28.

Sun, R. (2002). *Duality of the mind*. Mahwah, NJ: Lawrence Erlbaum Associates.

Sun, R. (2003). *A Tutorial on Clarion 5.0*. Technical report, Cognitive Sciences Department, Rensselaer Polytechnic Institute, Troy, NY. http://www.cogsci.rpi.edu/~rsun/sun.tutorial.pdf

Sun, R. (2004). Desiderata for cognitive architectures. *Philosophical Psychology*, *17*(3), 341–373.

Sun, R. (Ed.). (2006). *Cognition and multi-agent interaction*. New York: Cambridge University Press.

Sun, R. (2007). The importance of cognitive architectures: An analysis based on Clarion. *Journal of Experimental and Theoretical Artificial Intelligence*, *19*(2), 159–193.

Sun, R. (2007b). The motivational and metacognitive control in Clarion. In W. Gray (Ed.), *Integrated models of cognitive systems* (pp. 63–75). New York: Oxford University Press.

Sun, R. (Ed.). (2008). *The Cambridge handbook of computational psychology*. New York: Cambridge University Press.

Sun, R. (2009). Motivational representations within a computational cognitive architecture. *Cognitive Computation*, *1*(1), 91–103.

Sun, R. (2009b). Theoretical status of computational cognitive modeling. *Cognitive Systems Research*, *10*(2), 124–140.

Sun, R. (2012). Memory systems within a cognitive architecture. *New Ideas in Psychology*, *30*, 227–240.

Sun, R. (Ed.). (2012b). *Grounding social sciences in cognitive sciences*. Cambridge, MA: MIT Press.

Sun, R. (2013). Moral judgment, human motivation, and neural networks. *Cognitive Computation*, *5*(4), 566–579.

Sun, R. (2013b). Autonomous generation of symbolic representations through subsymbolic activities. *Philosophical Psychology*, *26*(6), 888–912.

Sun, R. (2014). Interpreting psychological notions: A dual-process computational theory. *Journal of Applied Research in Memory and Cognition*, in press.

Sun, R. & L. Bookman (Eds.). (1994). *Computational architectures integrating neural and symbolic processes*. Needham, MA: Kluwer Academic Publishers.

Sun, R., Coward, L. A., & Zenzen, M. J. (2005b). On levels of cognitive modeling. *Philosophical Psychology*, *18*(5), 613–637.

Sun, R. & Fleischer, P. (2012). A cognitive social simulation of tribal survival strategies: The importance of cognitive and motivational factors. *Journal of Cognition and Culture*, *12*(3–4), 287–321.

Sun, R. & Helie, S. (2012). Reasoning with heuristics and induction. *Proceedings of the 2012 International Joint Conference on Neural Networks*, Brisbane, Australia (pp. 1359–1366). Piscataway, NJ: IEEE Press.

Sun, R. & Helie, S. (2013). Psychologically realistic cognitive agents: Taking human cognition seriously. *Journal of Experimental and Theoretical Artificial Intelligence*, *25*, 65–92.

Sun, R., Honavar, V., and Oden, G. (1999). Integration of cognitive systems across disciplinary boundaries. *Cognitive Systems Research*, *1*(1), 1–3.

Sun, R. & Mathews, R. (2005). *Exploring the interaction of implicit and explicit processes to facilitate individual skill learning*. Technical Report TR-1162, Army Research Institute for the Social and Behavioral Sciences, Arlington, VA.

Sun, R. & R. Mathews. (2012). Implicit cognition, emotion, and meta-cognitive control. *Mind and Society*, *11*(1), 107–119.

Sun, R., Merrill, E., & Peterson, T. (2001). From implicit skills to explicit knowledge: A bottom-up model of skill learning. *Cognitive Science*, *25*, 203–244.

Sun, R. & Naveh, I. (2004). Simulating organizational decision-making using a cognitively realistic agent model. *Journal of Artificial Societies and Social Simulation*, *7*(3). http://jasss.soc.surrey.ac.uk/7/3/5.html

Sun, R. & Naveh, I. (2007). Social institution, cognition, and survival: A cognitive-social simulation. *Mind and Society*, *6*(2), 115–142.

Sun, R. & Peterson, T. (1998). Autonomous learning of sequential tasks: Experiments and analyses. *IEEE Transactions on Neural Networks*, *9*(6), 1217–1234.

Sun, R., Slusarz, P., & Terry, C. (2005). The interaction of the explicit and the implicit in skill learning: A dual-process approach. *Psychological Review*, *112*(1), 159–192.

Sun, R. & Wilson, N. (2011). Motivational processes within the perception-action cycle. In V. Cutsuridis, A. Hussain, and J. G. Taylor (Eds.), *Perception-action cycle: Models, architectures and hardware* (pp. 449–472). Berlin: Springer.

Sun, R. & Wilson, N. (2014). Roles of implicit processes: Instinct, intuition, and personality. *Mind and Society*, *13*(1), 109–134.

Sun, R. & Wilson, N. (2014b). A model of personality should be a cognitive architecture itself. *Cognitive Systems Research*, *29*–30, 1–30.

Sun, R., Wilson, N., & Mathews, R. (2011). Accounting for certain mental disorders within a comprehensive cognitive architecture. *Cognitive Computation*, *3*(2), 341–359.

Sun, R. & Zhang, X. (2004). Top-down versus bottom-up learning in cognitive skill acquisition. *Cognitive Systems Research*, *5*(1), 63–89.

Sun, R. & Zhang, X. (2006). Accounting for a variety of reasoning data within a cognitive architecture. *Journal of Experimental and Theoretical Artificial Intelligence*, *18*(2), 169–191.

Sun, R., X. Zhang, and R. Mathews, (2006). Modeling meta-cognition in a cognitive architecture. *Cognitive Systems Research*, *7*(4), 327–338.

Sun, R., Zhang, X., & Mathews, R. (2009). Capturing human data in a letter counting task: Accessibility and action-centeredness in representing cognitive skills. *Neural Networks*, *22*, 15–29.

Sun, R., Zhang, X., Slusarz, P., & Mathews, R. (2007). The interaction of implicit learning, explicit hypothesis testing learning, and implicit-to-explicit knowledge extraction. *Neural Networks*, *20*(1), 34–47.

Sutton, R. & Barto, A. (1998). *Reinforcement learning*. Cambridge, MA: MIT Press.

Taatgen, N. & Anderson, J. (2008). Constraints in cognitive architectures. In R. Sun (Ed.), *The Cambridge handbook of computational psychology* (pp. 170–185). New York: Cambridge University Press.

Taatgen, N., & Wallach, D. (2002). Whether skill acquisition is rule or instance based is determined by the structure of the task. *Cognitive Science Quarterly*, *2*(2), 163–204.

Tetlock, P. & Lebow, R.N. (2001). Poking counterfactual holes in covering laws: Cognitive styles and historical reasoning. *American Political Science Review*, 95, 829–843.

Thagard, P. (1996). *Mind: Introduction to Cognitive Science.* Cambridge, MA: MIT Press.

Thomson, J. J. (1985). The trolley problem. *Yale Law Journal*, *94*, 1395–1415.

Thórisson, K. R. & Helgasson, H. P. (2012). Cognitive architectures and autonomy: A comparative review. *Journal of Artificial General Intelligence*, *3*(2), 1–30.

Thorndike, E. (1911). *Animal intelligence*. Darien, CT: Hafner,

Timberlake, W. & Lucas, G. (1989). Behavior systems and learning: From misbehavior to general principles. In S. B. Klein & R. R. Mowrer (Eds.), *Contemporary learning theories: Instrumental conditioning theory and the impact of biological constraints on learning* (pp. 237–275). Hillsdale, NJ: Lawrence Erlbaum Associates,

Tinbergen, N. (1951). *The study of instinct*. London: Oxford University Press.

Toates, F. (1986). *Motivational systems*. Cambridge, UK: Cambridge University Press.

Tolman, E. C. (1932). *Purposive behavior in animals and men*. New York: Century.

Toth, J., Reingold, E., & Jacoby, L. (1994). Toward a redefinition of implicit memory: Process dissociations following elaborative processing and self-generation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *20*(2), 290–303.

Tsunoda, K., Yamane, Y., Nishizaki, M., & Tanifuji, M. (2001). Complex objects are represented in macaque inferotemporal cortex by the combination of feature columns. *Nature Neuroscience*, *4*(8), 832–838.

Tulving, E. (1972). Episodic and semantic memory. In E. Tulving and W. Donaldson (Eds.), *Organization of memory* (pp. 381–403). New York: Academic Press.

Tulving, E. (1983). *Elements of episodic memory*. Oxford: Clarendon Press.

Tulving, E. (1985). How many memory systems are there? *American Psychologist*, *40*, 385–398.

Tulving, E. and D. Schacter, (1990). Priming and human memory systems. *Science*, *247*, 301–305.

Tversky, A. (1977). Features of similarity. *Psychological Review*, *84*(4), 327–352.

Tversky, A. & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, *185*, 1124–1131.

Tversky, A. & Kahneman, D. (1983). Extensional versus intuitive reasoning: The conjunction fallacy in probability judgment. *Psychological Review, 90*, 439–450.

Tyrell, T. (1993). *Computational mechanisms for action selection*. PhD thesis, Oxford University, Oxford, UK.

Vaesen, K. (2012). The cognitive bases of human tool use. *Behavioral and Brain Sciences*, *35*(4), 203–262.

van de Vliert, E. (2013). Climato-economic habitats support patterns of human needs, stresses, and freedoms. *Behavioral and Brain Sciences*, *36*(5), 465–480.

van Fraassen, B. (1980). *The scientific image*. Oxford: Oxford University Press.

Vygotsky, L. (1962). *Thought and language*. Cambridge, MA: MIT Press.

Warrington, E. & Weiskrantz, L. (1970). Amnesic syndrome: Consolidation or retrieval? *Nature*, *228*, 628–630.

Watkins, C. (1989). *Learning with delayed rewards*. PhD thesis, Cambridge University, Cambridge, UK.

Weber, M. (1991). *Weber: Selections in translation*. Cambridge, UK: Cambridge University Press.

Wegener, D. T. & Petty, R. E. (2001). On the use of naive theories of bias to remove or avoid bias: the flexible correction model. In M. C. Gilly and J. Meyers-Levy (Eds.), *Advances in consumer research* (pp. 378–383). Valdosta, GA: Association for Consumer Research.

Wegner, D. M., & Bargh, J. A. (1998). Control and automaticity in social life. In D. Gilbert, S. T. Fiske, & G. Lindzey (Eds.), *Handbook of social psychology* (4th ed.) (pp. 446–496). New York: McGraw-Hill.

Wegner, D. M., & Wheatley, T. P. (1999). Apparent mental causation: Sources of the experience of will. *American Psychologist*, *54*, 480–492.

Weiner, B. (1992). *Human motivation: Metaphors, theories, and research*. Newbury Park, CA: SAGE.

Weinstein, N., Przybylski, A. K., & Ryan, R. M. (2013). The integrative process: New research and future directions. *Current Directions in Psychological Science*, *22*, 69–74.

White, J. (2010). Understanding and augmenting human morality: An introduction to the ACTWith model of conscience. In L. Magnani, W. Carnielli, and C. Pizzi (Eds.), *Model-based reasoning in science & technology* (pp. 607–621). Berlin: Springer.

Willingham, D. (1998). A neuropsychological theory of motor skill learning. *Psychological Review*, *105*(3), 558–584.

Willingham, D.,Nissen, M., & Bullemer, P. (1989). On the development of procedural knowledge. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *15*, 1047–1060.

Wilson, N. (2012). *Towards a psychologically plausible comprehensive computational theory of emotion*. PhD thesis, Rensselaer Polytechnic Institute, Troy, NY.

Wilson, N., Sun, R., & Mathews, R. (2009). Performance degradation under pressure. *Neural Networks*, *22*, 502–508.

Wilson, N., Sun, R., & Mathews, R. (2010). A motivationally based computational interpretation of social anxiety induced stereotype bias. *Proceedings of the Annual Conference of the Cognitive Science Society*, Portland, Oregon (pp. 1750–1755). Austin, TX: Cognitive Science Society.

Wilson, N. & Sun, R. (2014). Coping with bullying: A computational emotion-theoretic account. In P. Bello et al. (Eds.), *Proceedings of the Annual Conference of Cognitive Science Society*, Quebec City, Quebec, Canada (pp. 3119–3124). Austin, TX: Cognitive Science Society.

Wilson, N. & Sun, R. (in preparation). Modeling personality disorders.

Wilson, N., Sun, R., & Mathews, R. (in preparation). A detailed computational explanation of anxiety induced performance degradation.

Wine, J. (1971). Test anxiety and direction of attention. *Psychological Bulletin*, *76*, 92–104.

Winter, D. G., John, O. P., Stewart, A. J., Klohnen, E. C., & Duncan, L. E. (1998). Traits and motives: Toward an integration of two traditions in personality research. *Psychological Review*, *105*(2), 230–250.

Woike, B. (1995). Most memorable experiences: Evidence for a link between implicit and explicit motives and social cognitive processes in everyday life. *Journal of Personality and Social Psychology*, *68*, 1081–1091.

Wood, W. & Quinn, J. (2005). Habits and the structure of motivation in everyday life. In J. Forgas, K. Williams, and S. Laham (Eds.), *Social motivation: Conscious and unconscious processes*. New York: Cambridge University Press.

Wright, I. P., & Sloman, A. (1997). *MINDER1: An implementation of a proto-emotional agent architecture*. Technical Report CSRP-97-1, School of Computer Science, University of Birmingham, Birmingham, UK.

Wynn, T. (2002). Archaeology and cognitive evolution. *Brain and Behavioral Sciences, 25*(3), 389–438.

Yerkes, R. & Dodson, J. (1908). The relation of strength of stimulus to rapidity of habit-formation. *Journal of Comparative Neurology and Psychology*, *18*(5), 459–482.

Zajonc, R. B. (1980). Feeling and thinking: Preferences need no inferences. *American Psychologist*, *35*(2), 151–175.

Zerubavel, E. (1997). *Social mindscapes: An invitation to cognitive sociology*. Cambridge, MA: Harvard University Press.

Index

Note: Page numbers followed by "*f*" and "*t*" denote figures and tables, respectively.