SpringerWienNewYork

Nasrullah Memon · Jonathan D. Farley
David L. Hicks · Torben Rosenorn
Editors

Mathematical Methods in Counterterrorism

SpringerWienNewYork

Editors
Nasrullah Memon, Ph.D.
The Maersk Mc-Kinney Moller Institute
University of Southern Denmark
Campusvej 55
5230 Odense M
Denmark
memon@mmmi.sdu.dk

David L. Hicks, Ph.D.
Department of Computer Science
and Engineering
Aalborg University Esbjerg
Niels Bohrs Vej 8
6700 Esbjerg
Denmark
hicks@cs.aaue.dk

Jonathan D. Farley, D.Phil. (Oxon.)
Institut für Algebra
Johannes Kepler Universität Linz
4040 Linz
Austria
lattice.theory@gmail.com

Torben Rosenorn
Esbjerg Institute of Technology
Aalborg University Esbjerg
Niels Bohrs Vej 8
6700 Esbjerg
Denmark
tur@aaue.dk

Printed in Germany
SpringerWienNewYork is a part of Springer Science + Business Media
springer.at

Printed by: Books on Demand, Norderstedt, Germany
Printed on acid-free paper
SPIN 12197798

With 124 Figures

Library of Congress Control Number: 2009933187

ISBN 978-3-211-09441-9 SpringerWienNewYork

Foreword

Terrorism is one of the serious threats to international peace and security that we face in this decade. No nation can consider itself immune from the dangers it poses, and no society can remain disengaged from the efforts to combat it.

The term *counterterrorism* refers to the techniques, strategies, and tactics used in the fight against terrorism. Counterterrorism efforts involve many segments of society, especially governmental agencies including the police, military, and intelligence agencies (both domestic and international). The goal of counterterrorism efforts is to not only detect and prevent potential future acts but also to assist in the response to events that have already occurred.

A terrorist cell usually forms very quietly and then grows in a pattern – spanning international borders, oceans, and hemispheres. Surprising to many, an effective "weapon", just as quiet – mathematics – can serve as a powerful tool to combat terrorism, providing the ability to connect the dots and reveal the organizational pattern of something so sinister.

The events of 9/11 instantly changed perceptions of the words *terrorist* and *network*, especially in the United States. The international community was confronted with the need to tackle a threat which was not confined to a discreet physical location. This is a particular challenge to the standard instruments for projecting the legal authority of states and their power to uphold public safety. As demonstrated by the events of the 9/11 attack, we know that terrorist attacks can happen anywhere. It is clear that the terrorists operate in networks, with members distributed widely across the globe. To fight such criminals, we need to understand the new "terrain": networks – how they are formed, how they evolve, and how they operate.

The case for the development of mathematical methods and tools to assist intelligence and law enforcement officials is undeniable. The intelligence community is faced with the overwhelming tasks of gleaning critical information from vast arrays of disparate data, choosing the best forms of gathering information, developing new data sources and setting up personnel to effectively function in adverse circumstances. Each of these difficult tasks demands an intensive effort on behalf of the mathematical community to develop techniques and tools to assist in the asymmetric confrontation with secretive terrorist networks.

The important role of mathematics in the scientific advances of the last century, from advanced computing to lasers and optical communications to medical diagnosis, is now widely recognized. The central role of mathematical principles and techniques in assisting intelligence and law enforcement officials is equally striking. From new cryptographic strategies to fast data processing techniques, mathematical ideas have provided the critical link.

We are now faced with an extraordinary situation in the global fight against terrorism, and this book seeks to encourage the mathematical community to bring its full capabilities to bear in responding to this challenge. It presents the most current research from mathematicians and computer scientists from around the world aimed at developing strategies to support counterterrorism and enhance homeland security. These new methods are more important now than ever in order to glean the maximal possible benefit from the tremendous amount of information that has been gathered since 2001 regarding terrorist cells and individuals potentially planning future attacks.

I am confident that this book will help to advance the discourse, and enable its insights to be shared among a broad range of researchers, policy makers, politicians, and the members of intelligence and law enforcement agencies. The articles the book contains can help create momentum in the effort to transform theoretical techniques and strategies into concrete results on the ground.

Finally, I congratulate Nasrullah Memon, co-editors of the book, and the authors for their substantial efforts to address one of the most important crises faced by the international community today.

Brussels, January 2009 *Gilles de Kerchove*

Gilles de Kerchove is the EU Counterterrorism Coordinator and Law Professor at the Catholic University of Louvain and the Free University of Brussels.

Contents

Mathematical Methods in Counterterrorism: Tools and Techniques for a New Challenge

David L. Hicks, Nasrullah Memon, Jonathan D. Farley, and Torben Rosenørn

1 Introduction

Throughout the years mathematics has served as the most basic and fundamental tool employed by scientists and researchers to study and describe a wide variety of fields and phenomena. One of the most important practical application areas of mathematics has been for national defense and security purposes. For example, during the Second World War, the mathematical principles underlying game theory and cryptography played a very important role in military planning. Since that time, it has become clear that mathematics has an important role to play in securing victory in any global conflict, including the struggle faced by national security and law enforcement officials in the fight against those engaged in terrorism and other illicit activities.

Recent events of the past decade have produced an increased interest in and focus upon the area of counterterrorism by a broad range of scholars, including mathematicians. At the same time, government decision makers have often been skeptical about mathematics and statistics, even while faced with the considerable challenges of sifting through enormous amounts of data that might hold critically important clues. Realizing that policy makers were not always receptive, the mathematical

David L. Hicks
Department of Computer Science, Aalborg University, Esbjerg, Denmark, e-mail: hicks@cs.aaue.dk

Nasrullah Memon
The Mærsk Mc-Kinney Møller Institute, University of Southern Denmark, Odense, Denmark, e-mail: memon@mmmi.sdu.dk

Jonathan D. Farley
Institut für Algebra, Johannes Kepler Universität Linz, Linz, Austria, e-mail: lattice.theory@gmail.com

Torben Rosenørn
Esbjerg Institute of Technology, Aalborg University, Esbjerg, Denmark, e-mail: tur@aaue.dk

N. Memon et al. (eds.), *Mathematical Methods in Counterterrorism,*
DOI 10.1007/978-3-211-09442-6_1,

community has pondered about how best to put what they knew to work in building a more secure world. They felt especially qualified to help decision makers see the important patterns in the haystack of data before them and detect the most important and relevant anomalies.

Though governments have begun to engage the research community through grants and collaborative opportunities, across the sciences, and in particular within the fields of mathematics and statistics, the interesting problems and viable methodologies are still at a very early and speculative stage. The recently increased interest in counterterrorism has driven the research focus towards revisiting and strengthening the foundations necessary to build tools and design techniques capable of meeting the new challenges and producing more accurate results. This book provides a look at some of the latest research results in a variety of specialty topics that are central to this area.

2 Organization

This volume is composed of 21 contributions authored by some of the most prominent researchers currently focused on the application of mathematical methods to counterterrorism. The contributions span a wide variety of technical areas within this research field. In this book they have been organized into the five categories of network analysis, forecasting, communication/interpretation, behavior, and game theory. The remainder of this section provides a brief overview of the contributions in each of those categories.

Section 1: Network Analysis. The first section of the book begins with a contribution by Brantingham, Glässer, Jackson, and Vajihollahi. The authors describe their work on the development of a comprehensive framework and tool to support the mathematical and computational modeling of criminal behavior. They focus on criminal activities in urban environments, but also seek to extend the approach beyond conventional areas and support the application of computational thinking and social simulations to the analysis process in the area of counterterrorism. The next contribution in this section is from Skillicorn and discusses methods to obtain knowledge from graphs that are used to represent and study adversarial settings. It describes that, while graphs are appropriate for use in such analyses, they can also be more difficult to analyse than more traditional representations, and this article presents practical methods to help understand the structures these graphs contain.

The section continues with a contribution from McGough that examines the modeling of terrorist cells. The focus is on discussing and determining the strength of terrorist cell structures, and using the partially order set model and algorithms to do so. The next contribution in the first section is from Zaidi, Ishaque, and Levis. It describes an approach to combine and apply temporal knowledge representation and reasoning techniques to criminal forensics. An emphasis is placed on answering questions concerning time sensitive aspects of criminal or terrorist activities. The section concludes with a contribution by Farley that examines the structure of

the "perfect" terrorist cell. In particular it examines two theoretical questions related to the number of cutsets that exist in partially ordered sets that are used to represent that structure.

Section 2: Forecasting. A contribution by Gutfraind begins the second section of the book. It describes a dynamic model that is capable of representing the relevant factors involved as a terrorist organization changes over time. The approach enables those factors and their effects to be considered together, in a quantitative way, rather than individually, and for predictions to be made about the organization based on these analyses. In the next contribution of the section, Rhodes describes methods for constructing social networks of covert groups. The focus is on the use of Bayesian inference techniques to infer missing links, making the approach suitable for cases where only limited and incomplete data is available.

The following contribution, by Pinker, develops a model to represent the uncertainty in the timing and location of terror attacks. A framework is then described for guiding the issuance of terror warnings and the deployment of resources to combat attacks, and balancing the tradeoffs between these two defensive mechanisms. In the final contribution of the section, Ferry, Lo, Ahearn, and Phillips consider detection theory from the traditional military domain and its relationship to network theory. In particular they describe ways in which the detection theory approach might be used to leverage results from network theory as a way to find and track terrorist activities.

Section 3: Communication/Interpretation. The third section begins with a contribution by Hartnell and Gunther that focuses on communication in covert groups. A graph is used as a theoretical model to study the tradeoff between the competing demands of ease of communication and the potential danger for betrayal when members are captured, and a number of related questions. The section continues with a contribution from Woo that examines constraints on terrorist network plots imposed by the intelligence gathering efforts of law enforcement services. It also describes some of the implications inherent in both the intelligence gathering level and the methods that are utilized, and their relation to the important campaign to win the hearts and minds of the larger population.

In the following contribution, Lindelauf, Borm, and Hamers examine a theoretical framework for analysing homogeneous networks, especially in terms of the competing factors of secrecy and operational efficiency. An evaluation and comparison are provided of the different network topologies and their suitability for use for different graph orders. In the final contribution of the section, Ozgul, Erdem, and Bowerman describe two models for representing terrorist groups and analysing terrorist groups. The models are evaluated and compared in terms of their ability to support the semi-supervised detection of covert groups.

Section 4: Behavior. The fourth section starts with a contribution by Sliva, Subrahmanian, Martinez, and Simari in which they examine group behavior. Moving forward from previous research efforts focused on models derived mainly from past group behavior, they describe an architecture and algorithms to predict the conditions under which a group is likely to change their behavior. The next contribution in this section is from Baccara and Bar-Isaac and discusses the impact of methods of interrogation on terror networks. They investigate how the legal limits of interroga-

tion to extract information under which authorities operate relate to and have impact on the degree to which terrorist organizations diffuse or distribute their information in the attempt to increase the efficiency of the organization.

The section continues with a contribution from McCormick and Owen that discusses state-terrorist coalitions. They examine the factors of mutual advantage and mutual trust, how they change and evolve over time, and the impact these changes have upon the behavior of the partners in this type of coalition, and in particular the circumstances that might lead to continued cooperation, or the dissolution of the partnership. A contribution by McGough concludes the section with a look at the modeling of the behavior of terrorist groups. A mathematical model is described to represent terrorist groups and their behavior, and then it is evaluated through experimentation to test its projections, shedding light on the question of theory versus reality.

Section 5: Game Theory. A contribution by Melese begins the fifth section of the book. It describes a game theory approach to simulate terrorist cells. The principal question that is considered in the analysis examines under what conditions a threat of preemptive action by a world leader might successfully deter terrorist organizations or a sovereign state from the acquisition of weapons of mass destruction. The section continues with a contribution from Arce and Sandler in which they examine continuous policy models and describe an extension of them including differentiable payoff functions. They also consider the relationship between terrorist actions and government responses, and the effects they might have on the future support for terrorist organizations, within a game-theoretic context.

In the final contribution of the fifth section, Shapiro and Siegel discuss funding aspects of terrorist organizations. A model of a hierarchical terror organization is used to examine the implications of an arrangement in which leaders delegate financial and logistical tasks to middlemen, and, for security reasons, are not able to closely monitor their actions. A series of policy implications based on the analysis are also discussed.

Section 6: History of the Conference on Mathematical Methods in Counterterrorism. A contribution from Farley in the last section provides a look at the background and history of the Mathematical Methods in Counterterrorism conference series along with some personal observations.

3 Conclusion and Acknowledgements

The science of counterterrorism is still unfolding. As demonstrated by the variety of contributions to this volume, researchers welcome the opportunity to influence and shape the landscape of this important emerging area. The task before them is a challenging one, nothing less than to develop a new kind of mathematics, one that can equip national security and law enforcement officials with the tools they urgently need to face a new challenge – a new kind of war.

The editors would like to gratefully acknowledge the efforts of all those who have helped with the creation of this volume. Firstly, it would never be possible for a book such as this one to provide such a broad and extensive look at the latest research in the field of mathematical methods to combat terrorism without the efforts of all those expert researchers and practitioners who have authored and contributed papers.

Thanks are also due to Gilles de Kerchove, Counter-Terrorism Coordinator for the Council of the European Union, for providing a very motivational Foreword for the book. The support and guidance of a publisher is a critical component to a project such as this one. In this case thanks are due to Stephen Soehnlen and Edwin Schwarz, and their colleagues at Springer. Lastly, but certainly not least, the editors would like to acknowledge the considerable effort and support provided by Claus Atzenbeck with the layout, typesetting, and formatting of the book.

Part I
Network Analysis

Modeling Criminal Activity in Urban Landscapes

Patricia Brantingham, Uwe Glässer, Piper Jackson, and Mona Vajihollahi

Abstract Computational and mathematical methods arguably have an enormous potential for serving practical needs in crime analysis and prevention by offering novel tools for crime investigations and experimental platforms for evidence-based policy making. We present a comprehensive formal framework and tool support for mathematical and computational modeling of criminal behavior to facilitate systematic experimental studies of a wide range of criminal activities in urban environments. The focus is on spatial and temporal aspects of different forms of crime, including opportunistic and serial violent crimes. However, the proposed framework provides a basis to push beyond conventional empirical research and engage the use of computational thinking and social simulations in the analysis of terrorism and counter-terrorism.

1 Introduction

Innovative research in criminology and other social sciences promotes mathematical and computational methods in advanced study of social phenomena. The work

Patricia Brantingham
Institute for Canadian Urban Research Studies (ICURS), School of Criminology, Simon Fraser University, B.C., Canada, e-mail: pbrantin@sfu.ca

Uwe Glässer
Software Technology Lab, School of Computing Science, Simon Fraser University, B.C., Canada, e-mail: glaesser@sfu.ca

Piper Jackson
Software Technology Lab, School of Computing Science, Simon Fraser University, B.C., Canada, e-mail: pjj@sfu.ca

Mona Vajihollahi
Software Technology Lab, School of Computing Science, Simon Fraser University, B.C., Canada, e-mail: monav@sfu.ca

N. Memon et al. (eds.), *Mathematical Methods in Counterterrorism,*
DOI 10.1007/978-3-211-09442-6_2, © Springer-Verlag/Wien 2009

presented here proposes a comprehensive framework and supporting tool environment for mathematical and computational modeling of criminal behavior to facilitate systematic experimental studies of a wide range of criminal activities in urban environments. The focus is on uncovering patterns in the spacial and temporal characteristics of physical crime in urban environments, including forms of crime that are opportunistic in nature, like burglary, robbery, motor vehicle theft, vandalism, and also serial violent offenses such as homicide; they can involve multiple offenders and multiple targets. The principles of environmental criminalogy [1] suggest that criminal events can best be understood in the context of people's movements in the course of their everyday lives: offenders commit offenses near places they spend most of their time, and victims are victimized near places where they spend most of their time. This line of theory and supporting research argues that location of crimes is determined by perception of the environment – separating good criminal opportunities from bad risks [2] – and implies there is a set of patterns and/or rules that govern the working of a social system: one composed of criminals, victims and targets. They interact with each other and their environment, and their movements are influenced by the city's underlying land use patterns and high activity nodes like shopping centers and entertainment districts, street networks and transportation systems.

Computational methods and tools arguably have an enormous potential for serving practical needs in crime analysis and prevention, namely as instruments in crime investigations [3], as an experimental platform for supporting evidence-based policy making [4], and in experimental studies to analyze and validate theories of crime [5, 6]. The approach presented here proposes a formal modeling framework to systematically develop and validate discrete event models of criminal activities; specifically, it focuses on describing dynamic properties of the underlying social system in abstract mathematical terms so as to provide a reliable basis for computational methods in crime analysis and prevention. Besides training and sandbox experiments, our approach aims at intelligent decision support systems and advanced analysis tools for reasoning about likely scenarios and dealing with 'what-if' questions in experimental studies. Building on a cross-disciplinary R&D project in Computational Criminology [7], called *Mastermind* [8], we describe here the essential design aspects of the Mastermind system architecture in abstract functional and operational terms, emphasizing the underlying principles of mathematical modeling in an interactive design and validation context. The description presented here extends and complements our previous work [8, 9, 10] in several ways, opening up new application fields.

Mastermind is jointly managed by the Institute for Canadian Urban Research Studies (ICURS) in Criminology and the Software Technology Lab in Computing Science at SFU and has partly been funded by the RCMP "E" Division over the past three years. Crossing boundaries of research disciplines, the Mastermind project is linked to a wide range of research areas and application fields spanning criminology, computing, mathematics, psychology and systems science. Not surprisingly, any attempt to integrate such diverse views within a unifying computational framework in a coherent and consistent manner faces a number of challenging problems to be

addressed. A particular intriguing aspect is finding the right level of accuracy and detail to model real-world phenomena so that the resulting observable behavior is meaningful, at least in a probabilistic sense. This is closely related to the question of how *micro-level behavior* affects *macro-level behavior* and the observable phenomena under study. Another challenging aspect is the question of how to draw the boundaries of any such system, clearly delineating the system from the environment into which it is embedded: that is, what needs be included in the model and what is irrelevant in terms of the resulting behavior of interest?

The nature of modeling something as complex and diverse as crime is an ongoing and potentially open-ended process that demands for an interactive modeling approach – one that embraces frequent change and extensions through robustness and scalability of the underlying mathematical framework. The formal approach taken here builds on modeling and validation concepts using the *Abstract State Machine* (ASM) [11, 12] multiagent modeling paradigm together with CoreASM [13, 14], an open source modeling tool suite[1], as the formal basis for semantic modeling and rapid prototyping of mobile agents and their routine commuting activities through a virtual city they inhabit.

The remainder of this chapter is structured as follows. Section 2 discusses fundamental concepts in Computational Criminology and specific challenges and needs in mathematical and computational modeling of complex social systems. Section 3 introduces the mathematical framework and the tool environment used in the Mastermind project. Section 4 illustrates the main building blocks for modeling criminal activity, namely the representation of the urban environment and the agent architecture, and also summarizes some lessons learned from this project. Section 5 concludes the chapter.

2 Background and Motivation

This section briefly reviews the benefits of applying computational methods to studying crime. We first explain how this new way of thinking and problem solving benefits researchers in criminology. We then discuss related practical requirements of developing software tools in a collaborative research environment.

2.1 Computational Criminology

The use of computational techniques has become well-established and valuable in advancing the boundaries of many research disciplines, such as biology and chemistry. Research in Criminology, like other social sciences, faces the problem of lack of control in running experiments. Computational Criminology aims at pushing

[1] CoreASM v1.0.5 is readily available at http://www.coreasm.org.

these limits through interdisciplinary work with mathematics and computing science. By employing novel technologies and formal methodologies, existing theories of crime can be extended and new applications developed. Computational models allow for running experiments in simplified artificial situations where abstraction is used conveniently and systematically to adjust the influence of different elements under study. This facilitates dealing with the highly complex and dynamic nature of criminal activities. As such, beyond seeing computers as tools, *computational thinking* [15] presents a way of solving problems and designing systems in Criminology.

Conventional research in crime is empirical in nature and tends to use methods that rely upon existing data. In order to analyze the data, mostly statistical methods are used to derive a more abstract view of the data. Nowadays, however, computational methods offer a new way of thinking about the data that leads to new perspectives and new models for analyzing the problems.

Using computer simulations to conduct experiments virtually or to analyze "what-if" scenarios is now commonly practiced in the social sciences. The agent-based modeling paradigm is widely used for describing social phenomena, including criminal events, where individuals are represented by agents interacting with one another in a given environment. Different sub-areas of crime analysis have already benefited from the blending of criminology, computing science and mathematics [4, 16, 17]. For a more detailed review of computational modeling approaches in crime analysis we refer to [8].

Mathematical and computational modeling also introduces a great degree of precision. In the process of modeling (or defining) a theory of crime in abstract computational and mathematical terms, any incompleteness or lack of rigor becomes evident. This demands far more effort on the *theory development* side to complete existing theories, or discard incomplete ones, or even precisely identify the limitations of existing ones. This opens up new opportunities for criminologists to *test* the existing theories beyond conventional means.

There is particular value in expanding the use of computational thinking and social simulations in the analysis of terrorism and counter-terrorism. Terrorism is, as would be expected, a growing area of research in criminology where knowledge gained about human behavior in legal and traditional illegal activities such as robbery, homicide, and burglary is being expanded to include terrorist bombings, kidnapping, execution and general vulnerability to covert actions [18, 19]. Terrorism requires networks, exchange of information, and actions. There is a need for fitting into normal activity patterns until the terrorist act and to maintain safe locations. This is similar, yet different in severity, to many criminal activities where offenders operate in normal environs and search for targets close to established time-space patterns, with a preference for locations where the crime is acceptable or where actions by individuals are not closely watched. For example, shoplifting, motor vehicle theft and robbery occur in high activity shopping areas where diversity is accepted. These high activity and diverse areas are also attractors of terrorist bombing.

In particular, the development and modification of theories that cover terrorism need to push beyond conventional empirical research and engage in methods that allow the logical exploration of alternatives, the modification of contextual back-

ground, including cultural and economic differences [20], the interaction between terrorist support networks and anti-terrorism networks [21], covert and adaptive networks, and the ability to explore "what-if" scenarios of alternative policies, and the dynamic nature of simulation models.

2.2 Challenges and Needs

For the social sciences, applying computational techniques helps in overcoming some of the core limitations of studying social phenomena. Social scientists have always been limited by the inextricability of the subject of their research from its environment. Hence, it is difficult to study different factors influencing a phenomena in isolation. Particularly for fields that fall under the umbrella of security, safety and ethical issues can be an obstacle to innovation. For criminologists, it is very difficult to get first-hand evidence of crimes while they are being perpetrated – an observer would most likely be legally required to try to prevent the crime rather than letting it take place. Developing response strategies to unpredictable and dangerous situations is difficult to do in the field, since such situations are unpredictable and by their nature very difficult to control. Computational methods allow us to circumvent these problems by generating scenarios inside a virtual environment. In particular, modeling and simulation allow us to dynamically and interactively explore our ideas and existing knowledge.

Computational thinking about social phenomena, however, means thinking in terms of multiple layers of abstraction [15], which facilitates a systematic study of the phenomena by adjusting the level of detail given to the various factors under study. Computer models of social systems simulate dynamic aspects of individual behavior to study characteristic properties and dominant behavioral patterns of societies as a basis for reasoning about real-world phenomena. This way, one can perform experiments as a mental exercise or by means of computer simulation, analyzing possible outcomes where it is difficult, if not impossible, to observe such outcomes in real life.

2.3 Modeling Paradigm

In the process of interactive modeling of behavioral aspects of complex social systems, one can distinguish three essential phases, namely *conceptual modeling*, *mathematical modeling*, and *computational modeling*, with several critical phase transitions and feedback loops as illustrated in Fig. 1. Starting from a conceptual model that reflects the characteristic properties of the phenomena under study in a direct and intuitive way, as perceived by application domain experts, a discrete mathematical model is derived in several steps, gradually formalizing these properties in abstract mathematical and/or computational terms. This model is then transformed

into an initial computational model that is executable in principle; that is, any aspects that have been left abstract provisionally ought to be details to be filled in as the result of subsequent refinement steps. Ideally, any such refinement would be restricted to just adding details as required for running experiments both to help establishing the validity of the formal representation of the conceptual model and for further experimental studies. In reality, however, modeling is a highly iterative and essentially non-linear process with feedback loops within and also across the various phases, potentially affecting the design of the model in its entirety (see Fig. 1).

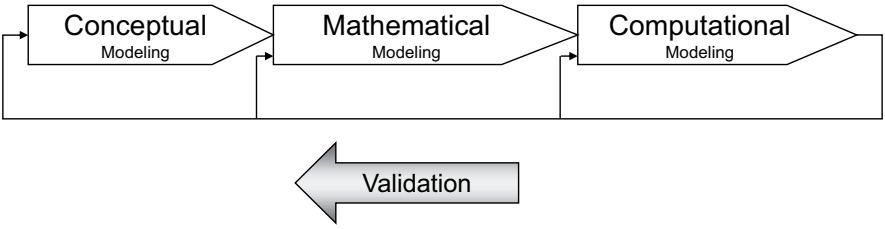

Fig. 1 Phase transitions in social system modeling

Specifically, the role of the mathematical model is to assist in formalizing the conceptual view of the target domain so as to provide an exact description of the characteristic properties as a reliable basis for deriving a computational model. Marking the transition from an informal (or semi-formal) to a formal description, this model serves as "semantic middleware" for bridging the gap between the conceptual and the computational view. As such, it provides a blueprint of the essential understanding, allowing systematic analysis and reasoning about the initial transformation step – typically, the most challenging one. Mathematical precision is essential to accurately characterize the key attributes, ensuring that they are properly established and well understood prior to actually building the computer model.

3 Mastermind Framework

Our approach to modeling complex criminal behavior in the Mastermind project follows the modeling paradigm of Sect. 2.3, emphasizing the need for mathematical rigour and precision. In order to accomodate the highly iterative process of modeling and validation, we build on common abstraction principles of applied computational logic and discrete mathematics using the Abtract State Machine method as the underlying mathematical paradigm. This section provides an overview of the ASM method, the CoreASM tool suite, which allows for rapid protyping, and the Control-State Diagram editor (CSDe), which was designed to facilitate a more interactive modeling approach.

3.1 Mathematical Framework

A central question in computing science is how to precisely define the notion of *algorithm*. Traditionally, Turing machines have been used in the study of the theory of computation as a formal model of algorithms [22]. For semantic purposes, however, this model is utterly inappropriate due to its fixed level of abstraction. The origin of Abstract State Machines (formally called *evolving algebras* [23]) was the idea to devise a generalized machine model so that any algorithm, never mind how abstract, can be modeled at its natural level of abstraction. That is, every computation step of the algorithm essentially has a direct counterpart (usually a single step) performed by the machine model. Theoretical foundations show that both the notion of sequential algorithm and of parallel algorithm are captured respectively by the models of *sequential ASM* [12] and *parallel ASM* [24] in the aforementioned sense. For distributed algorithms (including concurrent and reactive systems), the *distributed ASM* framework provides a generalization of the two other models that is characterized by its asynchronous computation model with multiple computational agents operating concurrently.

This section outlines the basic concepts for modeling behavioral aspects of distributed systems as abstract machine *runs* performed by a distributed ASM. The description builds on common notions and structures from computational logic and discrete mathematics. For further details, we refer to [11, 12].

3.1.1 Concurrency, Reactivity and Time

A distributed ASM, or DASM, defines the concurrent and reactive behavior of a collection of autonomously operating computational agents that cooperatively perform distributed computations. Intuitively, every computation step of the DASM involves one or more agents, each performing a single computation step according to their local view of a globally shared machine state. The underlying semantic model regulates interactions between agents so that potential conflicts are resolved according to the definition of *partially ordered runs* [23].[2]

A DASM M is defined over a given vocabulary V by its program P_M and a non-empty set I_M of initial states. V consists of some finite collection of symbols for denoting the mathematical objects and their relation in the formal representation of M, where we distinguish *domain symbols*, *function symbols* and *predicate symbols*. Symbols that have a fixed interpretation regardless of the state of M are called *static*; those that may have different interpretations in different states of M are called *dynamic*. A state S of M results from a valid interpretation of all the symbols in V and constitutes a variant of a first-order structure, one in which all relations are formally represented as Boolean-valued functions.

Concurrent control threads in an execution of P_M are modeled by a dynamic set AGENT of computational *agents*. This set may change dynamically over runs

[2] For illustrative application examples, see also [25, 26].

of M, as required to model a varying number of computational resources. Agents interact with one another, and typically also with the operational environment of M, by reading and writing shared locations of a global machine state.

P_M consists of a statically defined collection of agent programs $P_{M_1}, \ldots, P_{M_k}$, $k \geq 1$, each of which defines the behavior of a certain *type* of agent in terms of state transition rules. The canonical rule consists of a basic update instruction of the form

$$f(t_1, t_2, \ldots, t_n) := t_0,$$

where f is an n-ary dynamic function symbol and each t_i $(0 \leq i \leq n)$ a term. Intuitively, one can perceive a dynamic function as a finite function table where each row associates a sequence of argument values with a function value. An update instruction specifies a *pointwise* function update: an operation that replaces a function value for specified arguments by a new value to be associated with the same arguments. In general, rules are inductively defined by a number of well defined rule constructors, allowing the composition of complex rules for describing sophisticated behavioral patterns.

A computation of M, starting with a given initial state S_0 from I_M, results in a finite or infinite sequence of consecutive state transitions of the form

$$S_0 \xrightarrow{\Delta_{S_0}} S_1 \xrightarrow{\Delta_{S_1}} S_2 \xrightarrow{\Delta_{S_2}} \cdots,$$

such that S_{i+1} is obtained from S_i, for $i \geq 0$, by firing Δ_{S_i} on S_i, where Δ_{S_i} denotes a finite set of updates computed by evaluating P_M over S_i. Firing an update set means that all the updates in the set are fired simultaneously in one atomic step. The result of firing an update set is defined if and only if the set does not contain any conflicting (inconsistent) updates.

M interacts with a given operational environment – the part of the external world visible to M – through actions and events observable at external interfaces, formally represented by externally controlled functions. Intuitively, such functions are manipulated by the external world rather than agents of M. Of particular interest are *monitored functions*. Such functions change their values dynamically over runs of M, although they cannot be updated internally by agents of M. A typical example is the abstract representation of global system time. In a given state S of M, the global time (as measured by some external clock) is given by a monitored nullary function *now* taking values in a linearly ordered domain $\mathsf{TIME} \subseteq \mathsf{REAL}$. Values of *now* increase monotonically over runs of M. Additionally, $'\infty'$ represents a distinguished value of TIME, such that $t < \infty$ for all $t \in \mathsf{TIME} - \{\infty\}$. Finite time intervals are given as elements of a linearly ordered domain DURATION.

The ASM concept of *physical time* is defined orthogonally to the concept of state transition, flexibly supporting a wide range of time models, also including continuous time [27]. A frequently used model is that of distributed real-time ASM with time values ranging over positive real numbers.

3.1.2 ASM Ground Models

The ASM formalism and abstraction principles are known for their versatility in mathematical modeling of algorithms, architectures, languages, protocols and apply to virtually all kinds of sequential, parallel and distributed systems. Widely recognized ASM applications include semantic foundations of programming languages, like JAVA [28], C# [29] and Prolog [30], industrial system design languages, like BPEL [31], SDL [32], VHDL [33] and SystemC [34], embedded control systems [35], and wireless network architectures [36].[3] Beyond hardware and software systems, this approach has been used more recently in computational criminology [8, 9] and for modeling and validation of aviation security [37, 38]. A driving factor in many of the above applications is the desire to systematically reveal abstract architectural and behavioral concepts inevitably present in every system design, however hidden they may be, so that the underlying *blueprint* of the functional system requirements becomes clearly visible and can be checked and examined by analytical means based on human expertise. This idea is captured by the notion of *ASM ground model* [39] as explained below.

Intuitively, a ground model serves as a precise and unambiguous foundation for establishing the characteristic dynamic properties of a system under study in abstract functional and operational terms with a suitable degree of detail that does not compromise conceivable refinements [40]. A ground model can be inspected by analytical means (verification) and empirical techniques (simulation) using machine assistance as appropriate. Focusing on semantic rather than on syntactic aspects, the very nature of ASM ground models facilitates the task of critically checking the consistency, completeness and validity of the resulting behavioral description. Depending on the choice and representation of the ground model, the transformation from the mathematical to a computational model can be less problematic, whereas the validation of the outcome of the computational phase usually poses another difficult problem.

3.2 Rapid Prototyping with CoreASM

CoreASM is an open source project[4] focusing on the design and development of an extensible, executable ASM language, together with a tool environment that supports *high-level design* in application-domain terms, and *rapid prototyping* of executable ASM models [14, 13]. The tool environment consists of a (1) platform-independent extensible engine for executing the language, (2) various plugins that extend the language and the behavior of the engine, and (3) an IDE for interactive visualization and control of simulation runs. The design of CoreASM is novel and the underlying design principles are unprecedented among the existing executable

[3] See also the ASM Research Center at http://www.asmcenter.org.

[4] CoreASM is registered at http://sourceforge.net/projects/coreasm.

ASM languages, including the most advanced ones: Asmeta [41], AsmL [42], the ASM Workbench [43], XASM [44], and AsmGofer [45].[5]

The CoreASM language and tool suite specifically support the early phases of the software and system design process, emphasizing freedom of experimentation and the evolutionary nature of design as a creative activity. By minimizing the need for encoding in mapping the problem space to a formal model, the language allows writing highly abstract and concise specifications, starting with mathematically-oriented, abstract and untyped models, gradually refining them down to more concrete versions with a degree of detail and precision as needed. The principle of minimality, in combination with the robustness of the underlying mathematical framework, makes design for change feasible, effectively supporting the highly iterative nature of modeling complex system behavior.

Executable specifications offer invaluable advantages in model-based systems engineering, serving as a tool for design exploration and experimental validation through simulation and testing [47]. Pertinent to our purpose, they greatly facilitate validating a ground model by executing different scenarios and comparing the resulting behavior with the behavior *expected* by the domain experts. In many cases, observation of system behavior can lead to discovering *new* concepts or elements in the underlying system that may have been previously neglected.

3.3 Interactive Design with Control State ASMs

One of the fundamental principles of our approach is the direct involvement of non-computing experts in the design and development process. Arbitrary design choices made by computing experts not intimately familiar with the social system under study are potentially dangerous and can lead to fatal design flaws due to misconceptions or oversights. However, it is usually difficult for non-computing team members to understand the development process and especially the formal representation of a system. Hence, it is necessary to make development as transparent as possible, for instance, by using visual representation means, such as *ASM control state diagrams (CSD)*[6] as illustrated in Fig. 2. Despite similarity to the more complicated UML activity diagrams, ASM CSDs do not require any special training to understand. Their simplicity allows the interdisciplinary reader to focus on the content of the description rather than the formalism. The accessibility and ease of use of CSDs make them an integral part of our design process. In our experience, the domain experts were able to understand a CSD, and even suggest changes to it, regardless of their technical background. As such, CSDs act as both a means of clarifying communication between development partners and of enabling straight-forward validation.

[5] An in-depth introduction to the architecture of the CoreASM engine and its extensibility mechanisms is provided in [13, 46].

[6] Control state ASMs provide "a normal form for UML activity diagrams and allow the designer to define machines which below the main control structure of finite state machines provide synchronous parallelism and the possibility to manipulate data structures" [11].

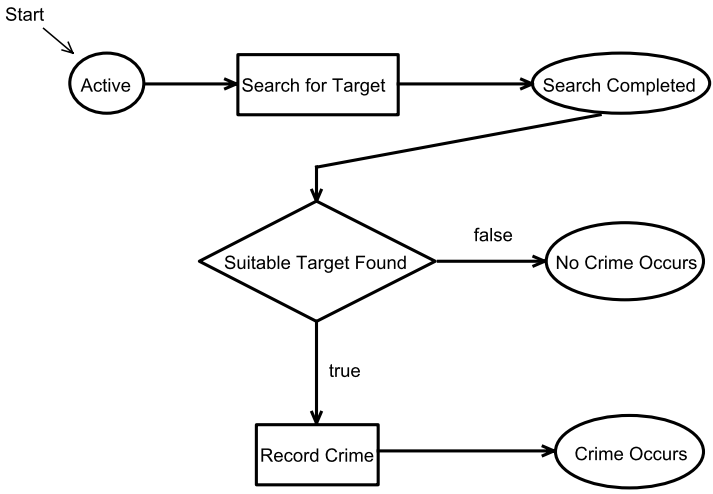

Fig. 2 CSDe: A Control State Diagram editor plugin for the Eclipse development environment, with automatic translation to CoreASM code.

We have further facilitated the involvement of non-computing experts in the development process by the construction of an ASM Control State Diagram editor (CSDe). This editor allows not only the construction and editing of CSDs through a graphical interface, but also automatic translation of the diagrams into the CoreASM language[7].

4 Mastermind: Modeling Criminal Activity

In this section, we describe the scope of the Mastermind project, its development phases, and the core technical aspects of the Mastermind system architecture.

4.1 Overview

Mastermind is a pioneering project in Computational Criminology, employing formal modeling and simulation as tools to investigate offenders' behavior in an urban environment. The goal is to capture the complexity and diversity of criminal behavior in a robust and systematic way. A variety of software development methods were applied and constantly reviewed with respect to their usability, expressiveness

[7] A diagram contains no initial state, so the code may not be immediately executable. However, it will provide a structural foundation for an executable model.

and effectiveness, the result of which has lead to the development of the modeling framework presented in Sect. 2.3.

Crime is understood to be comprised of four main elements: the law, the offender, the target and the location [1]. We construct a multi-dimensional model of crime in order to study the interactions of these elements. Our focus is on the concepts of environmental criminology, which argues that in spite of their complexity, criminal events can be understood in the context of people's movements in the course of everyday routines [1, 48]. Therefore, we place possible offenders in an environment they can navigate. Through their movement within this environment, they develop mental maps that correspond to the ideas of *awareness space* (the places a person knows) and *activity space* (the places a person regularly visits) [1, 49]. In the course of a routine activity, the agents move from one location to another, and may visit potential targets on the way [48]. In its core, Mastermind captures what is suggested by crime pattern theory: crime occurs when a motivated individual encounters a suitable target [49]. Figure 2 shows this behavior captured in terms of a control state ASM.

The main building block of Mastermind is a robust ASM ground model developed through several iterations. To this end, we applied a simple graphical notation for communicating the design (using CSDe) and utilized abstract executable models in early stages of design (using CoreASM). Furthermore, the ground model is refined into more concrete models with specific details systematically added, an example of which is the simulation model of Mastermind implemented in Java. This version provides a responsive user interface and a simulation environment based on real-world Geographical Information System (GIS) data. We also refined the CoreASM executable ground model to derive specific refinements to create more controlled experiments, which allow for a structured analysis of theories in a hypothetical world. Both versions also provide visualization features which are a priority for criminology publications.

The results of our work on the Mastermind project have been well received both by the researchers in academia and law enforcement officials. For additional information on the project and the results, we also refer to [8, 9] and the project website[8]. Next, we describe the main components of the Mastermind architecture, highlighting several key aspects.

4.2 Agent Architecture

The central component of our model is an autonomously acting entity, called a *person agent*, which represents an individual living in an urban environment and commuting between home, work, and recreation locations. Person agents *navigate* within the environment and may assume different roles such as offender, victim, or guardian; depending on the role they exhibit different behaviors.

[8] http://stl.sfu.ca/projects/mastermind/