Modeling Urban Crime Patterns:
Viewing Multi-Agent Systems as Abstract State Machines

P. L. Brantingham[2], U. Glässer[1], B. Kinney[2], K. Singh[1] and M. Vajihollahi[1]
{pbrantin/glaesser/bkinney/komals/mvajihol}@sfu.ca

[1] School of Computing Science
[2] School of Criminology
Simon Fraser University
Burnaby, B.C., Canada

Abstract. By combining the abstract state machine (ASM) formalism with the multi-agent modeling paradigm, we obtain a formal framework for semantic modeling and integration of established theories of crime analysis and prediction. We focus here on *spatial* and *temporal* aspects of crime in urban areas. Our work contributes to a novel interdisciplinary research effort broadly classified as Computational Criminology.

1 Introduction

Crime is not random. Any crime consists of four dimensions — the law, the offender, the target and the place [4]. Criminologists have developed various theories of crime [3] that contend there is definite patterning in the temporal and spatial characteristics of physical crime.

The main theoretical theme argues that criminal events can be understood in the context of people's movements in the course of 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 also argues that location of crimes is determined through a premeditated decision process of the offenders shaped by perceptions of environment that separate good criminal opportunities from bad risks — targets located along highly accessible street networks attract crime. This implies there are a set of patterns/rules that govern the working of a social system — one composed of criminals, victims and targets, interacting with each other — the movements of whom are influenced by the city's underlying land use patterns, street networks and transportation system.

Conventional research for crime analysis is statistical and empirical in nature using methods that entirely rely on direct extrapolations from past data. Novel research directions [7] however focus on a fundamentally different approach. Due to the growing dynamics and intricate nature of the underlying sociological systems, empirical deduction is not sufficient any more; mathematical and computational models are needed for reasoning about most likely scenarios. The introduction of computational models [27] in addition to statistical methods extends the given limitations and offers a rigorous tool for *understanding* of and *experimenting* with crime patterns and for *evidence based policy making*. Furthermore, although the theories are well-established in their respective fields, the lack of a coherent and consistent framework for integrating the four dimensions of crime inhibits their applicability to real-life scenarios.

By combining the abstract state machine (ASM) formalism [28, 2] with the multi-agent modeling paradigm [40], we obtain a formal framework for semantic modeling and integration of established theories of crime analysis and prediction. Specifically, we focus here on crime in urban areas and model *spatial* and *temporal* aspects of crime involving multiple offenders and multiple targets. Our goal is twofold. Semantic modeling simplifies the task of systematic integration and validation of crime patterns and theories and helps gaining a better understanding of complex social systems. At the same time, we obtain discrete event models that serve as simulation models providing effective instruments in prediction and prevention of crime. A prerequisite for building such simulation models in a well defined and systematic way, is a solid mathematical foundation serving as a ground model [1] for refinement and validation of simulation models.

The system context calls for a combination of two basically different views — agents interacting with each other and with the environment in which they live form a dynamic system whose behavioral pattern is best characterized as a combination of *reactive system* and *cognitive system*. Our work is part of a novel cross-disciplinary research effort broadly classified as Computational Criminology. The scope of related research disciplines spans criminology, formal methods, modeling & simulation, AI, and psychology. This gives us an interesting exploratory opportunity for applying ASMs beyond their natural habitat of hardware/software systems to social systems — a truly discrete, dynamic type of system; one which the very original definition of ASMs intended to capture.

This paper is structured as follows. Section 2 briefly outlines the criminology background and the context in which this work is carried out. Section 3 discusses the role of multi-agent modeling for social systems, while Section 4 presents a distributed ASM model of the underlying multi-agent system. Section 5 addresses the fitness of the approach, and Section 6 concludes the paper.

2 Computational Criminology

Criminology has a rich history of interest in the spatial distribution of crime and criminal events [10, 37]. The sub-field of environmental criminology stems, in part, from this early ecological tradition [3]. Key concepts include the routine nature of many of our daily activities [20], and the structured way in which we become aware of, and interact with, our environment [4, 5].

The task of simulating the movement and decision-making processes for our hypothetical offender (agent) is complex, but the main elements can be shown here. Principles of environmental criminology suggest that offenders commit crime in areas with which they are most familiar. If we assume that our agent moves about in space much like most people, the agent is then tied to at least three main classes of activity nodes: home, work and recreation. The agent often travels between these nodes using familiar pathways; the more often an area is visited, the more knowledge the agent will gain regarding the immediate surrounds for both the nodes and the pathways connecting them [6]. At a general level, each agent has an *awareness space* that characterizes his or her working knowledge of the environment. Within this general area more specialized knowledge is formed by direct experiences, known as *activity spaces*. It is here that criminal opportunities are observed and potentially acted upon [6].

If the agent starts his or her day at home, and then travels to either work (or school), for example, he or she will typically take the most direct and most easily navigated route. Along the way, the agent will take notice of a range of phenomena in his or her activity space. For example, the agent might see a favorite coffee house, or take notice of a particular shopping opportunity along the way to his or her work node. Even if the agent does not stop and interact specifically with potential activity sites, he or she will often remember such sites the next time he or she wants to purchase a cup or coffee or to patronize a particular business. The same learning process applies to the travel paths to and from other significant activity spaces. Environmental criminologists view the learning and decision-making process for crimes to be much the same as those for non-criminal activities. Criminals are rarely criminal all of the time. Everyone has a degree of criminal potential; what differs is the level of criminal propensity that each person 'brings along with us' on our daily routines. More criminally predisposed agents will respond to observed crime opportunities more frequently than people with low (or very low) crime potentials. What requires modeling is the process by which agents move from node to node, and when, if at all, does he or she act upon an observed criminal opportunity.

The process of choosing a target of opportunity involves target templating [5]. Each agent compares potential criminal opportunities to his or her crime template in order to assess the value potential rewards against the risks or degree of energy required for successful execution [11, 35]. A typical crime in large cities is that of burglary, otherwise known as break and enter, or "B&E". In the large majority of cases, for the agent to be aware of a potential target, the site must be located within the agent's activity space -which, in turn, is defined by the set of common activity nodes, as discussed earlier. The burglar would travel from Node 1 (home) to Node 2 (work), as his or her routine may require. Along the way he or she recognizes a residential building that is suggestive of a "good" target, as it "fits" within that particular agent's crime template. Research on burglars [34, 35, 18] suggests that variables such as: property value, lack of occupants or potential witnesses capable of intervening, and obvious entry opportunities, all form 'cues' [17] from which the agent assesses quickly to determine if either the criminal event will be attempted, or at least investigated further.

Related modeling and simulation work in criminology [30, 7], suggest the value of pursuing computational methods to predict patterns in crime. Although a 'new' area for criminology, this area is showing signs of gaining momentum.

3 Multi-Agent Based Modeling

Multi-Agent Based Systems (MABS) [40, 42, 22] have gained widespread popularity as a modeling paradigm which facilitates modeling distributed systems, open dynamic environments, and systems that can naturally be regarded as a society of interacting agents [25]. Specifically, using agent-based models for simulating social systems (social simulation) is justified by its long history [26] and successful applications [15, 16] in the field of social sciences. The history of social simulation starts with differential equations and goes through stochastic processes, game theory, cellular automata and finally distributed artificial intelligence (multi-agent systems)[26].

Figure 1 illustrates how the research area of Agent-Based Social Simulation (ABSS) can be characterized as the intersection of three scientific fields, agent-based computing, social sciences, and computer simulation [12]. In comparison with other modeling techniques depicted in the figure, agent-based modeling is proven to be the most useful one to capture non-linear system dynamics [12].

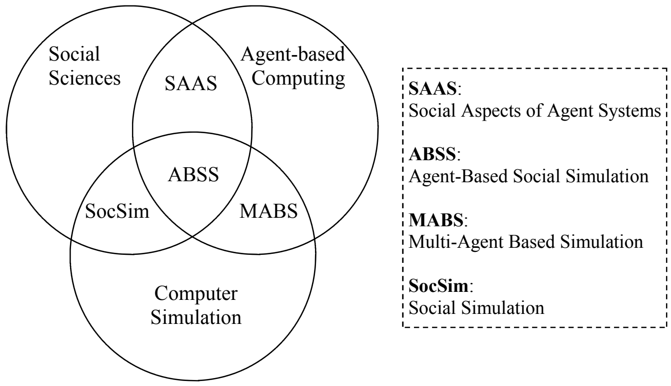

Fig. 1. Agent Based Social Simulation (ABSS) as the Intersection of Three Fields

Pertinent to our context, the multi-agent modeling paradigm plays a vital role in bringing together the predominant view of the world of social systems with the formal ASM view. This nexus provides a coherent framework to incrementally refine the model by integrating approaches from other disciplines such as AI and psychology, thus incorporating the cross-disciplinary nature in an elegant fashion.

Given that there is a bewildering list of notions of agents [24], for our purpose, we adopt a generic definition of an agent that subsumes most extant views. We define an agent as an active entity that is situated in an environment of which it has a partial representation, possesses resources and skill sets [22], exhibits autonomy, social ability, reactivity and pro-activeness [41]. A multi-agent system is then a collection of objects, agents and autonomous agents interacting with each other in a given environment according to the given laws of action [22].

3.1 Formal Approaches to Agent-Based Systems (FAABS)

Although MABS is widely used, it still lacks a formal semantic basis [14]: *'There is a lot of formal theory in the area but it is often not obvious what such theories should represent and what role the theory is intended to play. Theories of agents are often abstract and obtuse and not related to concrete computational models.'*

Relatively little work has been done in terms of formal approaches to multi-agent systems. Wooldridge and Fisher [23] outline a formal approach to specification, verification, and rapid prototyping of multi-agent systems in PML and Concurrent METAEM. Luck and Inverno [32] propose a formal framework for agency and autonomy using the Z specification language. Hilaire et al [29] present a formal specification of a multi-agent system based organizational model using Object-Z and state charts.

3.2 Linking Social Systems to DASM Models

We build on the existing knowledge and expertise of applying the multi-agent modeling paradigm to modeling social systems, and use the DASM framework for building a concise mathematical model of a particular class of such systems. As depicted in figure 2, the final DASM model of the system is linked to the original social system through an intermediate layer which is the multi-agent system.

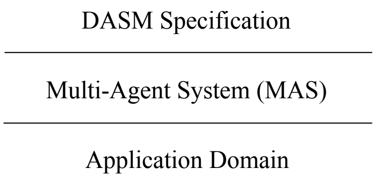

Fig. 2. Mapping Social Systems to DASM Models

However, the introduction of this intermediate layer should be seen only as a facilitating means for modeling social systems; i.e. only those principles of multi-agent systems that help in streamlining our target system are incorporated.

Since the underlying social urban environment is composed of different types of entities, a distinction needs to be made between the constituent entities. We propose a generic classification of entities into three different categories[1]: *passive*

[1] It must be noted that although generic, this categorization is not intended to be universal; It is only meant to capture the dynamics of our target system

Social Urban Environment (Bottom Layer)	**Multi-Agent System** (Intermediate Layer)	**DASM Model** (Top Layer)
Criminal, Non-Criminal, Cop	Autonomous Agent	DASM Agent
Traffic Light	Active Object	DASM Agent
ATM, Building, Jewelery	Passive Object	Object

Fig. 3. Agent Classification and Taxonomy Through Different Layers

objects, *active objects* and *autonomous agents*. The core of this classification is based on the essentials of the *Belief Desire Intention* (BDI) agent architecture [9, 41] and the *framework for agency and autonomy* [32].

A passive object is an entity that comprises of a set of *attributes*. An attribute is nothing but a characteristic feature. For instance, an ATM machine is a passive object that has attributes such as location and bank name. An active object is an entity that, in addition to a set of attributes, has an associated *behavior*. The behavior is described as an action that produces a change in the external environment (by way of *action rules*) or in the internal state of the entity (by way of *cognition rules*). A traffic light is an active object that has a set of attributes (such as location) and a predefined but changing behavior (such as being red, yellow, or green).

An autonomous agent is an entity that, in addition to attributes and behavior, has a set of *motivations* and a *memory*. Motivations are certain constructs that give rise to the behavior. This in turn implies that an autonomous agent is responsible for generating all its behavior. Memory plays the same role as *beliefs* in the BDI agent architecture. It is nothing but a collection of facts, everything that an agent 'knows' about its environment, past actions, et cetera. A criminal is an autonomous agent that has attributes like age, and behavior like working, eating, or stealing which are determined by motivations like hunger or greed. It also has a memory that saves the agent's knowledge about the environment such as the locations of the targets.

The next step is to define the entity framework in the DASM model. We map each entity from the multi-agent system layer to an entity in the DASM model as follows. A passive object is modeled as a DASM object. Active objects are modeled as DASM agents where the program of each agent characterizes the behavior of the respective active object. Finally, autonomous agents are also modeled as DASM agents and their memory and motivations are abstractly represented by functions.

Figure 3 illustrates the agent mapping through the three different layers.

4 Distributed ASM Model

The DASM model we present here is built on a multi-agent system view where the behavior of agents is defined by two different sets of rules, reactive rules and cognitive rules. Agents operate in a highly dynamic, discrete environment. They are capable of learning using a form of behavioral reinforcement learning, where based on past experiences, certain preferences are developed that may influence future choices that an agent will make. Likewise, communication between agents may influence the behavior of an agent in various ways. Agents communicate with one another by reading and writing shared locations of global states. For synchronization purposes, we use an explicit event triggering mechanism defined in [19]. Intuitively, communicating agents may develop various forms of interactions like *indifference*, *cooperation* or *antagonism* which in turn lead to the formation of organizations [22]. At this stage, we essentially restrict on two forms of interaction, indifference (agents operate independently without influencing one another) and, to some extent, cooperation, but potentially allow for other forms of interactions as well.

4.1 Agent Architecture

The main component of our model is an autonomously acting entity, called the *person agent*, which is defined to model the criminal behavior of some offender in an urban environment. The behavioral description of a person agent is structured into several modules, each of which models certain aspects of the overall behavior. Figure 4 illustrates the structural decomposition of a person agent into its different logical components following the classical Divide and Conquer approach.

The approach taken on modeling the urban environment complements extant theories of environment in the behavioral sciences. Conceptually, the environment can be divided into two broad categories, the objective environment and the subjective environment [31]. The objective environment, in our case called the *geographic environment*, is external to an agent and encompasses the totality of all things jointly forming the physical reality.

The Space Evolution Module (SEM) is responsible for carving out the *activity space* and the *awareness space* of the person agent. It uses a navigation algorithm to move the agent from an origin to a destination considering the particular preferences of the agent. These preferences include a variety of aspects such as choosing shorter routes, avoiding traffic, using familiar roads, avoiding blocked roads, et cetera.

The Target Selection Module (TSM) is responsible for monitoring potential targets on the routes taken by the agent, and for selecting attractive targets based on given selection criteria. This leads to the creation of the *opportunity space* and the *crime occurrence space* of the agent.

The Agent Decision Module (ADM) is the component which models intelligent decisions made by the agent. Most of these decisions are made based on personal attributes of the agent (recorded in the person's profile) and the agent's

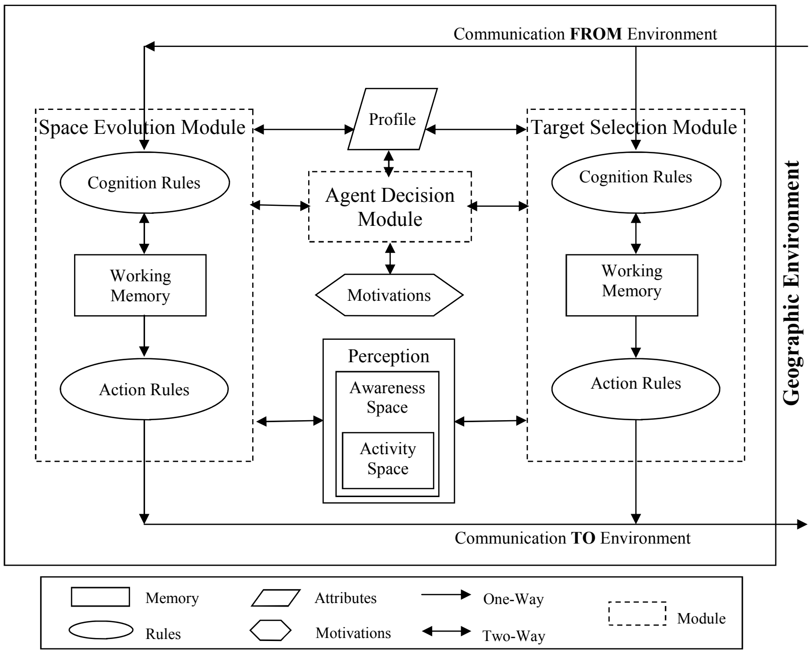

Fig. 4. Agent Architecture

motivations. This module is left abstract in the current model but is part of the architectural design and serves as an interface to incorporate existing AI methods into our model.

The SEM and the TSM both use two kinds of rules: *action rules* and *cognition rules*. Cognition rules change the state of the internal memory, whereas action rules, when fired, change the state of the external geographic environment. Each of these modules is identified with a DASM agent and the corresponding rules form the program of the respective agent.

Memory is a collection of facts/beliefs/knowledge, present and past, which may change dynamically over time. In our model, each of the above mentioned modules has an internal memory component, called *working memory*, which stores volatile information needed for path computation or target selection.

The agent's main memory component is called *perception* and represents the subjective environment. Perception stores a filtered view of the geographic environment and represents the environment as the agent 'perceives' it; it is thus the subjective reality and is constantly updated by the SEM and the TSM.

The subjective environment may further be divided into sub-categories [38]. The part of the perception that an agent is aware of by way of current events, past experiences, interaction with other agents forms the awareness space of the agent. The activity space is the part of the awareness space that the agent has frequently visited over a recent period of time. The agent typically has very detailed information about this part of the environment. This categorization of environment is illustrated in Figure 5.

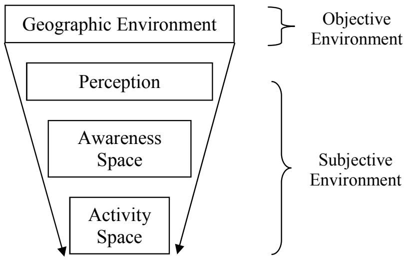

Fig. 5. Categorization of the Environment

Finally, the *profile* represents the personal attributes of the agent and stores information such as preferences, skills, home locations and demographic factors that are taken into account for determining certain aspects of an agent's individual behavior when moving in space and time.

4.2 The Environment

We formally define the geographic environment as a directed attributed graph in several steps as follows.

Let $H = (V, E)$ be a directed graph representing the road map of some urban environment, where $V = \{1, ..., n\}$ is the set of vertices representing the intersections on the map and $E \subseteq V \times V$ is the set of directed edges representing roads, unidirectional ones by a single edge and bidirectional ones by a pair of oppositely directed edges connecting the same two vertices.

Further, let Θ_v and Θ_e be the attribute set for vertices and edges respectively. Θ_e splits into two disjoint subsets, Θ_e^{stat} and Θ_e^{dyn}, representing the edge attributes that are static and those that my change dynamically.

The geographic environment can now be defined as a directed attributed graph $G_{GeoEnv} = (V, E, \theta)$ with $\theta = (\theta_v, \theta_e)$, $\theta_e = (\theta_e^{stat}, \theta_e^{dyn})$ where θ defines the attribution of vertices and edges of G_{GeoEnv} by means of three finite mappings such that

- $\theta_v : V \rightarrow \mathcal{P}(\Theta_v)$ assigns a set of vertex attributes to each vertex in V,
- $\theta_e^{stat} : E \rightarrow \mathcal{P}(\Theta_e^{stat})$ assigns a set of static edge attributes to each edge in E, and
- $\theta_e^{dyn} : E \rightarrow \mathcal{P}(\Theta_e^{dyn})$ assigns a set of dynamic edge attributes to each edge in E.

Vertex attributes, for instance, specify the geographic coordinates of the vertex, while edge attributes, for instance, specify the length of the road, the type of the road, the traffic situation, the road condition, et cetera.

G_{GeoEnv} abstractly represents the objective environment and serves as the basis for defining the subjective environment — the agent's perception of the objective environment. We model perception by defining an additional attribution on top of G_{GeoEnv} where the fact that, in general, each agent perceives the geographic environment differently implies that different agents have different attributions of G_{GeoEnv}.

Let Λ_v and Λ_e be the attribute set for vertices and edges respectively. The environment is defined as a directed attributed graph $G_{Env} = (V, E, \theta, \lambda)$ with $\lambda = (\lambda_v, \lambda_e)$ where λ defines the attribution of vertices and edges of G_{Env} by means of two injective mappings such that

- λ_v : AGENT$\times V \rightarrow \mathcal{P}(\Lambda_v)$ for each agent and each vertex in V yields a non-empty set of vertex attributes, and
- λ_e : AGENT$\times E \rightarrow \mathcal{P}(\Lambda_e)$ for each agent and each edge in E yields a non-empty set of edge attributes.

G_{Env} can be seen as a colored, directed attributed graph, where each color represents the perception of a specific agent. Λ_v, for instance, specifies the number of visits to a vertex and the agent's interest in a vertex. Λ_e, for instance, specifies the number of visits of an edge, reinforcement factors, the importance of the trips over an edge, and the intensity of awareness.

4.3 High-Level DASM Specification

An offender is modeled by a collection of DASM agents, a master agent of type PERSON and three sub-ordinate agents operating under the control of the master agent. Each of the sub-ordinate agents we identify with one of the three operational units, SEM, TSM, and ADM (see Section 4.1). The interactions between these modules is based on an event triggering mechanism carried out using a signaling framework in which each signal specifies its sender, the receiver and could carry any type of data. For more details on the syntax and semantics of this approach the reader is referred to [19].

domain PERSON
domain SEM
domain TSM
domain ADM

MODULE $\equiv$ SEM $\cup$ TSM $\cup$ ADM
AGENT $\equiv$ PERSON $\cup$ MODULE

parentAgent : MODULE $\rightarrow$ PERSON

In the following sections, we introduce the SEM and the navigation algorithm in more detail. For more information on the TSM and its functionality the reader is referred to [8].

Space Evolution Module The main responsibility of the SEM is to control how a person traverses the geographic environment given by G_{Env} (see Section 4.2) based on personal preferences. We access the graph structure and its attributes through function names that will be introduced on the fly.

As a byproduct of navigation, that is as the person visits more places, the activity space of this person evolves which will affect attribute values of G_{Env}. The SEM works in five different modes: *idle*, *estimation*, *calculation*, *thinking* and *running*, and moves the person from one vertex on the graph to another until it arrives at the destination.

In the *idle* mode, the SEM is waiting for the next destination to be identified. The next destination is provided by the ADM through a signal. When the next destination is identified, a series of calculation is required to find the best path toward it (*calcRequired* is set to true), and the mode is changed to *estimation*.

SEMidleMode $\equiv$
  onsignal s : NEW_DEST
    $destNode(self) := newDest(s)$
    $calcRequired(self) := true$
    $mode(self) := estimation$

The navigation algorithm takes into account a number of factors to find the most preferable path to the destination. The importance of each of these factors is determined by an associated weight which can change dynamically. For instance, generally major roads are more favorable than minor roads, but when the person gets closer to the destination, the minor roads that directly lead to the destination can be more attractive. These weights are re-evaluated in the *estimation* mode. Furthermore, while in the *estimation* mode it is possible to reset the destination and start moving towards a new destination. This means that the person can change his or her mind in the middle of a trip.

SEMestimationMode $\equiv$
  onsignal s : NEW_DEST
    $destNode(self) := newDest(s)$
    $calcRequired(self) := true$
  RE-EVALUATE_FACTOR_WEIGHTS($self$)
  // Re-evaluates the weights, if any is changed *calcRequired* is set to true
  $mode(self) := calculation$

In the *calculation* mode, auxiliary calculations for finding best paths are performed. Due to the nature of the navigation algorithm, which will be described later, a re-calculation is required whenever the destination has changed, or the weights have changed (*calcRequired* = true). When the calculation is complete, the mode is changed to *thinking*.

SEMcalculationMode $\equiv$
  if $calcRequired(self)$ **then**
    GLOBAL_RECALCULATION($self$)
  else
    $mode(self) := thinking$

In the *thinking* mode, if the agent has already reached the destination, a signal is sent to the ADM to announce arrival at the destination, and the mode is changed back to *idle*. Otherwise, the next edge towards the destination is chosen; i.e. an edge with the best overall preference. The mode is then changed to *running* and the *currentNode* function becomes *undef* which indicates that the agent is no longer residing on a specific vertex but is traversing an edge. The current time is also recorded which allows having a rough estimate of the time when the agent reaches the next vertex (the tail of the selected edge).

SEMthinkingMode $\equiv$
  if $\neg$*destNodeReached* **then**
    CHOOSE_NEXT_ROAD(*self*)
    // The next edge is chosen and *currentEdge* is set to the chosen edge
    currentNode := *undef*
    traverseStartTime(*self*) := *now*
    // The agent starts passing the edge
    mode(*self*) := *running*
  else
    mode(*self*) := *idle*
    trigger s : INFORM_ARRIVAL, *decisionModule*(*parentAgent*(*self*))
    // Informing the decision module that the agent has arrived at the destination
      arrivalTime(s) := *now*
where
  currentNode $\equiv$ *currentNode*(*parentAgent*(*self*))
  destNodeReached $\equiv$ *currentNode* = *destNode*(*self*)

While in the *running* mode, persons may communicate with each other. This communication is modeled through signals that are received by the SEM and might influence the perception of the person. The communication between persons and the mechanism for exchanging knowledge have been studied extensively in other disciplines. Although it is left abstract in our model so far, we have identified it as a possible future extension to the model. Finally, in the *running* mode if the edge traversal is complete and the person has reached the next vertex (*currentEdgeTraversed* = true), it is recorded in the perception that the current edge has been visited once and the mode is changed back to *estimation*.

SEMrunningMode $\equiv$
  onsignal s : AGENT_ENCOUNTER
    EXCHANGE_KNOWLEDGE(*neighbor*(s))
  if *currentEdgeTraversed*(*self*) **then**
    RECORD_IN_PERCEPTION(*self*, *currentEdge*)
    currentNode := *edgeTail*(*currentEdge*)
    mode(*self*) := *estimation*
where
  currentEdge $\equiv$ *currentEdge*(*parentAgent*(*self*))

Navigation Algorithm The navigation algorithm that we present here reflects the natural and intuitive choices a person makes while moving in an urban landscape. The path taken might not be a globally optimal and the best one, but it is a more natural and a good-enough one. The algorithm takes into account the factors that are known to influence human path planning and is developed in collaboration with the domain experts.

Generally, a navigator has some global information about its environment which is incorporated in the algorithm through *global factors*. Global factors include distance, number of intervening vertices, road type and angle towards the destination. Global factors do not change during a trip to the destination, and the navigator has normally a priori knowledge of them. Intuitively, major roads are preferred to minor roads and an optimal path is sought to minimize the distance, the number of stops, and the overall angle toward the destination (It is uncommon to choose a road on the opposite direction of the destination). In addition, past experiences of the person are also considered as global factors. The factors related to past experience are categorized as *learning factors* and include frequency of trips, importance of trips and positive/negative reinforcement. Typically, paths that have been visited more frequently before are more preferable.

The algorithm should also consider the local information that is discovered by the navigator on-the-fly. This forms the *local factors* which are highly dynamic, including traffic, road conditions and any other local information about a road. The idea is to avoid dense traffic or any road construction by using the local information available.

The navigation algorithm reflects the balance between global planning, local negotiation and learning. It can be seen as a combination of A*-algorithm [36] and Dijkstra's algorithm [13] for finding the shortest paths.

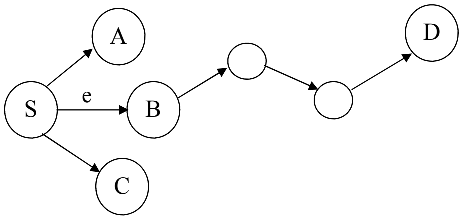

Fig. 6. Selecting a Path from Source S to Destination D

Assume a person wants to move from source S to destination D (Figure 6). The preference of an edge e for an SEM agent a is defined as:

$$edgePref(a, e, D) \equiv globalWeight(a) * globalEdgePref\,(a, e, D) + localWeight(a) * localEdgePref\,(a, e)$$

where

- $globalEdgePref(a,e,D)$ is the preference of taking a 'best' path from S to B to destination D influenced solely by global factors which corresponds to global planning. A best path is computed using Dijkstra's algorithm.

– *localEdgePref*(a,e) is the preference of choosing edge e influenced solely by local factors which corresponds to local negotiation.

Learning is incorporated in path planning by considering past experiences of an agent as part of the global factors. Local knowledge of an edge is also considered as part of the local factors which resembles a form of short term learning.

Furthermore, the effect of local and global factors in the overall preference of an edge is controlled by the weights assigned to each type of preference, namely *globalWeight* and *localWeight*.

Global preference of an edge serves as a heuristic estimate of the preference of the path which goes through this edge towards the destination. This estimate is provided by Dijkstra's shortest path algorithm and takes into account only the global factors that are fixed while the destination has not changed. Hence, whenever the destination is changed or any factor weight is changed, it is required to re-apply the algorithm and update these estimates. The GLOBAL_RECALCULATION routine takes care of this whenever necessary.

Local preference of an edge is considered on the fly and as the agent reaches the respective edge. Overall, choosing the next edge towards the destination is done as follows:

CHOOSE_NEXT_ROAD$(self : \mathrm{SEM}) \equiv$
  choose $edge$ **in** $outIncidentEdges(currentNode)$ **with** $maxPref(edge)$
    $currentEdge := edge$
where
  $maxPref(edge) \equiv \forall e(e \in outIncidentEdges(currentNode)$
      $\Rightarrow edgePref(self, edge, dest) \geq edgePref(self, e, dest))$
  $dest \equiv destNode(self)$

5 Fitness of the Model

Mathematical modeling of crime serves the purpose of studying most likely scenarios in crime analysis and prevention and provides a basis for experimental research allowing experiments that can often not easily be done in the real world. Typical purposes of such experimental research, among others, include [39]: a) use of a model for generating and testing hypothesis through controlled experiments; b) use of a model for predicting the effects of change in the system under study. In this context, the role of modeling and simulation is descriptive rather than prescriptive and, for instance, focuses on aspects such as: a) identification of behavior characteristics from response patterns generated by simulations; and b) identification of the system boundary and the factors that influence the behavior of interest. Clearly, the key issue is not to obtain a simulation response as the main result but to inspect the underlying trace that generated the response and suggest changes on that basis. Hence, the problem of providing evidence for the validity of a model has a different meaning than in prescriptive modeling where quantitative measures of reliability of predictions apply.[2] An obvious way of validating the model is by simulating scenarios and comparing the results with what the theories predict. However, there is no guarantee that the theories make correct predictions under all circumstances. This aspect deserves a more thorough investigation and is subject of our future work.

6 Conclusion

We define a multi-agent system in terms of a distributed ASM serving as abstract mathematical framework for semantic modeling of spatial and temporal aspects of crime in urban environments. Although unconventional, the application of the ASM formalism and abstraction principles to modeling social systems turns out to be a promising approach; it nicely combines with the established view of multi-agent modeling of social systems and provides a precise semantic foundation — something multi-agent modeling is lacking.

The particular challenge we are facing here is the complexity and diversity of the problem space due to two major factors: a) the inherent complexity and dynamics of social systems, and b) the cross-disciplinary nature of Computational Criminology spanning criminology, modeling & simulation, AI, and psychology. In this light, our main theoretical result is the abstract ground model of person agents (based on the *agent architecture*) interacting with their objective and subjective environments. Our model is designed for robustness with the intention to extend and refine it as required to gradually incorporate principles and techniques from other research disciplines, for instance, for dealing with various cognitive aspects, especially in modeling the target selection process. To this end, we can say that the ASM abstraction mechanisms greatly simplified the task of extracting and formalizing behavioral aspects of the system under study and were invaluable for delineating the borderline between the system and its operational environment.

Our main practical result is an executable version of the abstract multi-agent model which is based on AsmL [33] and serves as a platform for the construction and experimental validation of discrete event simulation models. Simulation is considered a practical tool in crime analysis and prevention.

References

1. E. Börger. The ASM Ground Model Method as a Foundation for Requirements Engineering. *Verification: Theory and Practice*, pages 145–160, 2003.
2. E. Börger and R. Stärk. *Abstract State Machines: A Method for High-Level System Design and Analysis.* Springer-Verlag, 2003.
3. P. J. Brantingham and P. L. Brantingham. *Patterns in Crime.* New Tork: Macmillan Publishing Company, 1984.

[2] For instance, in software cost estimation, a widely accepted rule of thumb suggests that a model is acceptable if 75 percent of the predicted values fall within 25 percent of their actual values [21].

4. P. J. Brantingham and P. L. Brantingham. Introduction: The Dimensions of Crime. In P. J. Brantingham and P. L. Brantingham, editors, *Environmental Criminology*, pages 7–26. Waveland Press, 1991.
5. P. J. Brantingham and P. L. Brantingham. Environment, Routine and Situation: Toward a Pattern Theory of Crime. *Advances in Criminological Theory*, pages 259–294, 1993.
6. P. J. Brantingham and P. L. Brantingham. Nodes, Paths and Edges: Considerations on the Complexity of Crime and the Physical Environment. *Journal of Environmental Psychology*, pages 3–28, 1993.
7. P. J. Brantingham and P. L. Brantingham. Computer Simulation as a Tool for Environmental Criminologists. *Security Journal*, pages 22–30, 2004.
8. P. L. Brantingham, U. Glässer, K. Singh, and M. Vajihollahi. Mastermind: Modeling and Simulation of Criminal Activity in Urban Environments. Technical Report SFU-CMPT-TR-2005-01, Simon Fraser University, Feb 2005.
9. M. E. Bratman, D. Israel, and M. E. Pollack. Plans and Resource-Bounded Practical Reasoning. *Computational Intelligence*, 4:349–355, 1988.
10. E. W. Burgess. The Growth of the City. In M. Stewart, editor, *The City: Problems of Planning, Penguin Interdisciplinary Readings*, pages 117–129. Penguin Books Ltd., 1972(1925).
11. D. Cornish and R. V. Clarke. Introduction. *The Reasoning Criminal: Rational Choice Perspectives on Offending, Research in Criminology*, pages 1–16, 1986.
12. P. Davidsson. Agent Based Social Simulation: A Computer Science View. *Journal of Artificial Societies and Social Simulation*, 5, January 2002.
13. E. W. Dijkstra. A Note on Two Problems in Connection with Graphs. *Numerische Mathematik*, 1:269–271, 1959.
14. M. d'Inverno, M. Fisher, A. Lomuscio, M. Luck, M. de Rijke, M. Ryan, and M. Wooldridge. Formalisms For Multi-Agent Systems. In *The First UK Workshop on Foundations of Multi-Agent Systems*, 1996.
15. J. Doran, M.Palmer, N. Gilbert, and P. Mellars. The EOS project: Modelling Upper Palaeolithic Social Change. In N. Gilbert and J. Doran, editors, *Simulating Society: The Computer Simulation of Social Phenomena*, chapter 9, pages 195–221. UCL Press, London, 1994.
16. A. Drogoul and J. Ferber. Multi-agent Simulation as a Tool for Studying Emergent Processes in Societies. In N. Gilbert and J. Doran, editors, *Simulating Society: The Computer Simulation of Social Phenomena*, chapter 6, pages 127–142. UCL Press, London, 1994.
17. J. E. Eck and D. Weisburd. Crime Places in Crime Theory. In *Crime and Place, Crime Prevention Studies*, pages 1–33. The Police Executive Research Forum, 1995.
18. P. Ekblom and N. Tilley. Going Equipped: Criminology, Situational Crime Prevention and the Resourceful Offender. *British Journal of Criminology*, pages 376–398, 2000.
19. R. Farahbod. Extending and Refining An Abstract Operational Semantics of the Web Services Architecture for the Business Process Execution Language. Master's thesis, Simon Fraser University, Burnaby, Canada, July 2004.
20. M. Felson. Routine Activities and Crime Prevention in the Developing Metropolis. *Criminology*, pages 911–931, 1987.
21. N.E. Fenton and S.L. Pfleeger. *Software Metrics: Rigorous and Practical Approach*. International Thomson Press, 1997.
22. J. Ferber. *Multi-Agent Systems: An introduction to Distributed Artificial Intelligence*. Addison Wesley, February 1999.

23. M. Fisher and M. Wooldridge. Towards Formal Methods for Agent-Based Systems. In D. Duke and A. Evans, editors, *BCS-FACS Northern Formal Methods Workshop. Electronic Workshops in Computing*. Springer Verlag, 1997.
24. S. Franklin and A. Gasser. Is It an Agent, Or Just a Program?: A Taxonomy for Autonomous Agents. In *Proceedings of the Third International Workshop on Agent Theories, Architectures and Languages*. Springer-Verlag, 1996.
25. N. Gilbert and J. Doran. *Simulating Societies*. UCL Press, 1995.
26. N. Gilbert and K. G. Troitzsch. *Simulation for the Social Scientist*. Open University Press, 1999.
27. L. Gunderson and D.Brown. Using a Multi-Agent Model to Predict Both Physical and Cyber Criminal Activity. *IEEE International Conference on Systems, Man, and Cybernetics*, 4:2338–2343, 2000.
28. Y. Gurevich. Evolving Algebras 1993: Lipari Guide. In E. Börger, editor, *Specification and Validation Methods*, pages 9–36. Oxford University Press, 1995.
29. V. Hilaire, A. Koukam, P. Gruer, and J. P. Müller. Formal Specification and Prototyping of Multi-agent Systems. *Lecture Notes in Computer Science*, 1972, 2001.
30. P. Hooimeijer and A. Oskamp. Advances in the Microsimulation of Demographic Behaviour. In *Population Issues: An Interdisciplinary Focus, The Plenum Series on Demographic Methods and Population Analysis*, pages 229–263. Plenum Press, 1999.
31. K. Koffka. *Principles of Gestalt Psychology*. Harcourt, 1967.
32. M. Luck and M. d'Inverno. A Formal Framework for Agency and Autonomy. In V. Lesser and L. Gasser, editors, *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*, pages 254–260, San Francisco, CA, USA, 1995. AAAI Press.
33. Microsoft FSE Group. *The Abstract State Machine Language*. cited Oct 2004, `http://research.microsoft.com/fse/asml/`.
34. G. F. Rengert. Burglary in Philadelphia: A Critique of an Opportunity Structure Model. In *Environmental Criminology*, pages 189–201. Waveland Press Inc., 1991.
35. G. F. Rengert. The Journey to Crime: Conceptual Foundations and Policy Implications. In *Crime, Policing and Place: Essays in Environmental Criminology*, pages 109–117. Routlede, 1992.
36. S. C. Shapiro. *Encyclopedia of Artificial Intelligence*. John Wiley & Sons, 1992.
37. C. R. Shaw and H. D. McKay. *Juvenile Delinquency and Urban Areas: A Study of Rates of Delinquency in Relation to Differential Characteristics of Local Communities in American Cities*. University of Chicago Press, 1969 (1942).
38. J. Sonnenfeld. Geography, Perception and the Behavioral Environment. In P. W. English and R. C. Mayfield, editors, *Man, Space and the Environment*, pages 244–251. Oxford University Press, New York, 1972.
39. P. J. Starr. Modeling Issues and Decisions in System Dynamics. In JR A. A. Legasto, J. W. Forrester, and J. M. Lyneis, editors, *System Dynamics*. North Holland, 1980.
40. K. P. Sycara. Multi Agent Systems. *AI Magazine*, pages 79–92, 1998.
41. M. Wooldridge. Intelligent Agents: The Key Concepts. In M. Luck, V. Marik, O. Stepankova, and R. Trappl, editors, *Multi-Agent Systems and Applications II*, pages 3–43. Springer, 2001.
42. M. Wooldridge. *An Introduction to Multi Agent Systems*. John Wiley and Sons Ltd, 2002.