2015

Modeling social norms in real-world agent-based simulations

Rahmatollah Beheshti
University of Central Florida

MODELING SOCIAL NORMS IN REAL-WORLD AGENT-BASED SIMULATIONS

by

RAHMATOLLAH BEHESHTI
B.S. Kharazmi University, 2008
M.S. Iran University of Science and Technology, 2011

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2015

Major Professor: Gita Sukthankar

ABSTRACT

Studying and simulating social systems including human groups and societies can be a complex problem. In order to build a model that simulates humans actions, it is necessary to consider the major factors that affect human behavior. Norms are one of these factors: social norms are the customary rules that govern behavior in groups and societies. Norms are everywhere around us, from the way people handshake or bow to the clothes they wear. They play a large role in determining our behaviors. Studies on norms are much older than the age of computer science, since normative studies have been a classic topic in sociology, psychology, philosophy and law. Various theories have been put forth about the functioning of social norms. Although an extensive amount of research on norms has been performed during the recent years, there remains a significant gap between current models and models that can explain real-world normative behaviors. Most of the existing work on norms focuses on abstract applications, and very few realistic normative simulations of human societies can be found.

The contributions of this dissertation include the following: 1) a new hybrid technique based on agent-based modeling and Markov Chain Monte Carlo is introduced. This method is used to prepare a smoking case study for applying normative models. 2) This hybrid technique is described using category theory, which is a mathematical theory focusing on relations rather than objects. 3) The relationship between norm emergence in social networks and the theory of tipping points is studied. 4) A new lightweight normative architecture for studying smoking cessation trends is introduced. This architecture is then extended to a more general normative framework that can be used to model real-world normative behaviors. The final normative architecture considers cognitive and social aspects of norm formation in human societies. Normative architectures based on only one of these two aspects exist in the literature, but a normative architecture that effectively includes both of these two is missing.

For *Leila*,

my partner in all things.

ACKNOWLEDGMENTS

First and foremost I should thank my adviser, Dr. Gita Sukthankar, whom I had the privilege of working with while preparing this dissertation. I learned so many things from her, although she never taught them directly to me.

Thanks to my committee members, including Dr. Lotzi Bölöni and Dr. Annie Wu, who gave me valuable feedback, and corrected me on many mistakes that I made while completing my dissertation. Specially, I should thank Dr. Samarth Swarup (Virginia Tech VBI) who had excellent knowledge from both computer science and public health domains, and gave me extremely insightful ideas for shaping my work.

Dr. Mary Schmidt-Owens, the associate director of medical health administration at UCF, helped me on several critical occasions by providing health related data of UCF students. She helped me to understand the meanings of some medical aspects of smoking behaviors.

UCF Parking Services provided me with statistical data about parking usage at UCF. The Office of the Provost at UCF, and specially Dr. MJ Soileau, Vice President for Research and Commercialization, helped me to distribute my survey about transportation patterns to all the UCF students.

I had the honor of talking and consulting with Dr. David L. Sallach from University of Chicago about category theory. He provided me with several references including one article that he was still working on.

Dr. Sandip Sen from University of Tulsa, kindly gave his time to me, and described the details of his work on the game-theoretical implementation of social learning. I have frequently used some ideas from his work on modeling norm emergence in agent-based systems.

Last but not least, I wanted to thank my teammates in Intelligent Agents Lab (IAL), specifically Bulent Tastan, Xi Wang, Erfan Davami, Alireza Hajibagheri, and Hamidreza Alvari who helped me on various occasions to find better solutions for solving my research challenges.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

The modeling and simulation of human behaviors are known to be complex problems, particularly at the level of modeling human societies. In order to study the behaviors of humans usually it's necessary to model more than one aspect. For instance, choosing what type of clothes to wear, cannot be modeled just by knowing the personal preferences of a human; other factors like her friendship network, the content of TV and social media, and the clothing retail markets can affect this behavior. As more factors, especially the social ones, come into the play, a more powerful method is needed to model these factors. Modeling the emergence of norms in human societies is one example of a complex large-scale human modeling problem. Psychological characteristics [49], friendship network status [41] and financial incentives are important to the norm formation process.

The general way to model complex human behaviors is to decompose them into subdivisions. This could be done either using a top-down or bottom-up approach. In the later case, each member of society is considered as a separate entity. Agent-based modeling (ABM) is a popular technique that functions using the bottom-up approach. The ability of ABM comes from its focus on defining micro details about agents which leads to the emergence of macro behaviors in the society as a large. This method is employed in this dissertation to simulate the behavior of agents in the systems. Here, agents' social interactions, transportation behaviors and personal characteristics are modeled through ABM. Having a detailed agent-based model facilitates the modeling of norms in realistic situations. A major aspect of this dissertation is creating an agent-based model to simulate human normative behaviors on the main campus of Universitity of Central Florida.

A *normative agent* refers to an autonomous agent who demonstrates normative behavior; these agents must be able to reason about the norms with which they should comply, and occasionally violate them if they are in conflict with each other or with the agent's private goals [120].

For individual agents, reasoning about social norms can easily be supported within many agent architectures; Dignum [65] defines three layers of norms (private, contract, and convention) that can be used to model norms within a BDI framework. At the population level, norm emergence, whether a group of agents converges to a consistent set of norms, is an interesting question, and both theoretical and computational models have been presented to describe norm emergence in social systems [155, 172]. Previous work on norms, such as the EMIL project [119], has shown promising results on modeling real-world phenomena such as traffic patterns, Wikipedia article authorship, and financial decisions.

In this work, two theories from mathematics and social sciences are employed to study and describe the presented ideas. The mathematical one is category theory, and the other is the theory of tipping points from social sciences. A case study on the use of category theory is presented to formally describe and analyze agent-based models. The power of category theory is that it can be used to express different types of systems in a common language. It was originally introduced in order to handle problems in algebraic topology and homology theory [106]. Category theory enables one to abstract a formal system by eliminating superfluous details. By mapping a problem to a seemingly unrelated problem in another domain, it becomes possible to leverage known proofs and solutions from the original domain. The main purpose of using category theory is to mathematically show the logic behind the hybridization of ABM and Markov Chain Monte Carlo (MCMC) techniques. It should be noted that theoretically any modeling technique could be used to construct a hybridized technique. Category theory helps us show why ABM and MCMC are good candidates. Additionally, the mathematical representation of ABM presents a new formal representation for agent-based models. The common problem with these models is that it's very difficult to reproduce the results obtained from them. This problem arises mainly because there is not a practical unified approach for formal representation of agent-based models. By using category theory, these concerns will be addressed.

On the other hand, the theory of tipping points deals with seeming minor causes to deep changes in the behavior of human societies. These changes start with gradual ones, and end with sharp shifts in a population-level behavior. Three elements of this theory that were popularized by Malcolm Gladwell [125] are studied in this work. This is done, using some ideas from social network analysis domain, where the effects of these three factors are studied in the emergence of norms in multi-agent systems.

1.1 Motivation

One barrier to creating realistic large-scale models of human social systems is the lack of good general purpose computational models of human interactions; without such models, it is impossible to accurately account for the intricate action dependencies engendered by both explicit and implicit interpersonal communications. However research on special purpose human interaction models has flourished, bringing a greater understanding of the computational processes underlying teamwork [159], information diffusion [107], and adversarial situations [39]. Armed with these tools, social scientists have been able to mathematically describe more complicated social phenomena. Similarly, the research on computational models of norms and normative agent architectures is ripe for greater inclusion in social simulations. Normative multi-agent systems are a powerful tool for modeling complex social problems, including energy consumption, water usage, and soil conservation. For instance, social norms have been found to affect enrollment in payment for ecosystem services (PES) [48].

Group cohesion, the set of personal and task-related social forces uniting the members of a group, can exert a powerful influence on the actions of group members, increasing the incidence of correlated action [140]. Many group attributes influence cohesiveness — the expectation of future rewards resulting from group action, members' similarity, group size, and the presence of external threats [75]. The desire for increased group cohesion can motivate group members to change their actions without formally entering into a state of joint commitment [101]. Group members often exhibit a tendency towards groupthink in their decision-making, causing group members to minimize conflict and rapidly reach consensus [94]. Cohesion and groupthink combine to create measurable action dependencies among group members, reducing the number of potential actions considered by group members when deciding on a course of action and creating action synchrony.

The existence of social norms, implicit expectations about the behavior of in-group members, can be viewed as a consequence of these group-based social forces. Norms play a significant role in determining the behavior of people in human societies, and have been used as a computational mechanism for creating coordinated action within normative multi-agent systems. Previous work on modeling norm lifecycles can be organized into two categories: internal and external. In the first category, norms are characterized as arising from internal mental processes that can be specified using cognitive modeling techniques, and normative behavior is viewed as the outcome of internalizing external preferences. The normative agents are able to acquire new norms, rather than relying on preexisting constructs, and can deliberate about norm compliance autonomously [55]. In the second category, the focus is on social interactions, and game-theoretic models are used to quantify the bottom-up process of recognizing and complying with norms in the external social system [155]. Convergence occurs when agents arrive at a mutually agreed upon utility maximization strategy. A limitation of this type of system is that the agents lack a sense of normative expectation and do not distinguish between a strategy and a social norm [148]. However complex human behaviors often contain elements of both types of mechanisms embedded within the decision-making process. Ideally a realistic simulation of human behavior should support both mechanisms.

The proposed normative architecture in this research, Cognitive Social Learners (CSL), bridges the gap between these two types of architectures and provides a computational mechanism for transitioning behaviors learned during repeated social interactions into the agent's internal cognitive model of preexisting beliefs, desires, and intentions. Rather than modeling the normative lifecycle as a sequence of stages (e.g., recognition, adoption, compliance), CSL implements norms through an iterative process in which the normative behavior is developed incrementally within each agent's cognitive model before it emerges in consistent patterns of observable behavior.

As mentioned earlier, simulating real-world normative behaviors needs a model that includes adequate details of agents' characteristics and their environment properties. Since the real-world scenarios in this work occur in an urban environment, the agents are defined in a way that they mirror specific details required for this type of environments. Benenson et al. [29] present two motivations for defining urban agents as a distinct group within the general class of autonomous agents:

1. urban agents often have a high degree of mobility resulting in rapidly changing spatial relationships.
2. to succeed, urban agents require a strong capability to perceive and adapt to the evolving urban environment shaped by neighboring agents.

Urban simulation is a particularly fertile domain for research in agent-based simulation since it requires modeling a large number of interdependent agents making sequential decisions within a small region. Agent-based models have been used specifically to recreate urban environments for a wide variety of domains including: 1) civil and environmental transportation analysis [102, 4], 2) geographic information systems (GIS) for visualizing patterns and trends in spatial areas [105, 3], and 3) archaeological studies of land site usage in ancient civilizations [110].

Although these urban simulations do not necessarily have to model low-level physical interactions, including spatial information and heterogeneity in agent-based models will help us build models that can simulate complicated characteristics of real world environments in a more effective way [36]. With the inclusion of GIS to represent a spatially, georeferenced environment, the impact of human behavior patterns can be linked to specific spatial locations and when used correctly can provide a powerful tool for policy makers and the public to understand the potential consequences of their decisions [87].

Yet modelers attempting to analyze a complex urban region face a similar problem to the six blind men touching an elephant, who describe the whole elephant based on touching it. Since none of the men can feel more than a single small part of the elephant—the tail, the ear, a tusk, the belly, the trunk, and the leg—they each bring back a different report.[1] In the same way, different modeling techniques are very likely to produce slightly different answers to the same question. This phenomenon poses problems when urban simulations are used to influence important public policy debates, regulatory decisions, and to guide resource allocation. For instance, the public debate about human influence on climate change has been shaped by a disproportionate level of discussion about minor discrepancies between predictions, while the general trend consensus between models has remained largely ignored [73].

As part of this dissertation, a novel architecture for combining two powerful modeling techniques is introduced: agent-based models (ABM) and Markov Chain Monte-Carlo (MCMC) estimators. Although both of these methods have a long history of practical usage (summarized in the next two sections), they have weaknesses as well. ABMs can be used to simulate very complex social phenomena, but constructing easily reproducible agent-based models is difficult due to the possibility of emergent behaviors and lack of formal representation. According to [169], many ABMs, with the exception of a few classic models, have never been replicated by anyone but the original developer. It is difficult to bring mathematical analysis tools to bear on the problem, so instead models are typically studied through empirical simulation studies [97]. Yet the results of the simulation study can vary considerably by changing the range, or even the step size, of just one or two variables [133].

On the other hand, the Markov Chain Monte-Carlo simulation process can be described by a relatively simple set of mathematical equations and a resampling procedure; this methodology is sometimes referred as the most powerful idea in computational statistics [141]. The aim of the process is to approximate the posterior distribution of the model parameters based on the

[1]The parable of the blind men and the elephant appears in a number of religions originating from the Indian subcontinent.

observed data. However, the selection of the proposal distribution can have a significant impact on model convergence. In cases where the proposal distribution is far from the desired posterior distribution the algorithm may converge to a poor local minimum or require a long time to achieve convergence [86]. The nearer the proposal distribution is to the target distribution, the better the performance of the MCMC algorithm [127]. The reader can find more details about the role of proposal distribution in [6].

Agent-based modeling has been used successfully for studying many types of social and biological phenomena. Although the gold-standard test for an ABM is comparing its predictions to real-world data, often paucity of data can eliminate this form of comparison. More commonly, domain experts can be used to guide the modeler during the creation of the model and tuning of parameters. However, comparing one model to another remains a difficult challenge, particularly because it is often problematic to formally specify many types of agent-based models. The ideas from category theory are employed to address these issues, in addition to showing the relation between ABM and MCMC methods.

The power of category theory mainly comes from its focus on relations among the objects rather than the objects themselves. Historically, most of structures defined in category theory were defined in order to study and represent complex structures in a consistent way. Healy et al. [95] use the following analogy to illustrate the role that category theory could play in studying different disciplines. Imagine a scientist viewing an electrical circuit and a chemical compound. At first glance, they might appear to be very different structures, but a deeper look reveals that chemical bonds are also electrical in nature. A common meeting ground between electricity and chemistry can be found within the abstractions of physics: quantum states and the large-scale static/dynamic properties of electrons. These abstractions allow the scientist to define the relationship between electrical circuits and chemical compounds, transfer insights from one discipline to another, and study electrochemical reactions. Agent-based models are often used to encode discipline-specific ideas from psychology, sociology, or biology on the function of a complex system [123]. Representing these models in category theory could be the key to understanding the relationship between multiple agent-based models of the same system. Category theory empowers us to create mappings between the models and understand their operation in a functional way, rather than simply comparing the predictions of the simulations.

1.2 Approach

In general, there are two major approaches for constructing normative architectures which will be also discussed in Section 3.5. In one approach, the focus is on cognitive aspects of normative reasoning, and the norm reasoning is modeled mostly as an internal process that occurs inside an agent's mind. In the other approach, normative procedures are modeled mostly as external processes; it's an agent's interaction with the environment, especially the other agents that determines how the agent behaves. The approach taken in our proposed architecture unifies elements from both groups. In the presented architecture, an agent interacts with other agents and learns about normative behaviors through social communications and its observations from the environment. In addition, the agent has internal cognitive abilities to reason about norms using BDI (belief, desire and intention) structure. Figure 1.1 shows the overall workflow of the dissertation.

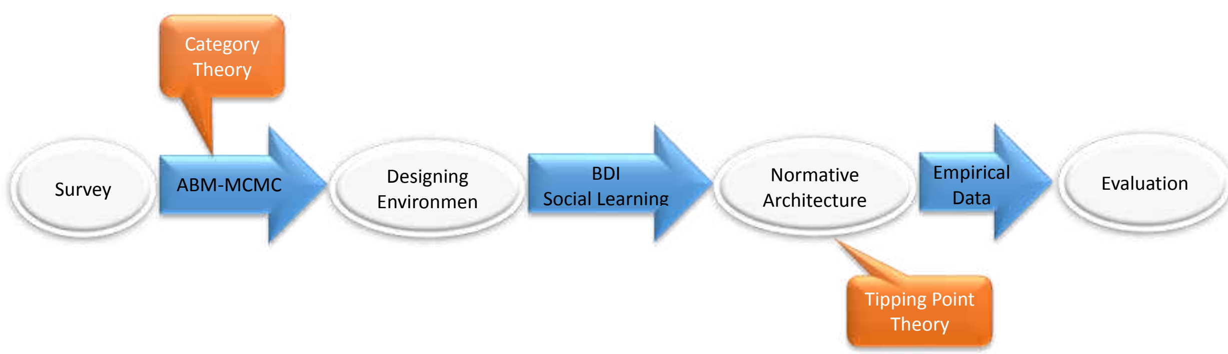


Figure 1.1: Workflow of the dissertation

Survey: The needed data for building the simulated environment is gathered through an online survey. Specifically, the students of University of Central Florida were asked about their daily commuting times to the school, places they visit on campus and the frequency of their visits.

Designing Realistic Environment: A big part of this dissertation is devoted to describing the process of building a realistic simulation of transportation patterns of people. Having a good model of an urban environment, is a necessity for studying the normative processes in this system. For instance, for studying smoking norms at UCF, a transportation model of students is needed in order to calculate encounter frequency between smoking and non-smoking students on campus. One major contribution of this dissertation is proposing a modeling technique that is basically a hybridization of two other modeling techniques, namely agent-based modeling and Markov Chain Monte Carlo. This new technique is used to build the model that can simulate transportation patterns of students. In order to show theoretically that hybridization is logical and mathematically sound, ideas from category theory are employed. Using category theory, which is a classic mathematical theory, the two methods are described, and then the relation between them is formally shown.

Normative Architecture: Two normative architectures are proposed and studied in this dissertation. The first one presents a simpler architecture without advanced cognitive and learning abilities. The second architecture includes the features from both cognitive based and social learner architectures. Also, the role of some ideas from the tipping point theory in emergence of norms in normative multi-agent systems is studied.

Evaluation: The agent-based model was updated such that each agent deliberates and behaves using the proposed normative architecture. The simulated normative behavior of agents was examined using several independent datasets. For instance, the number of campus smokers determined by the proposed model was compared to the numbers available from UCF Health Services. Moreover, the performance of the two proposed normative architectures is compared using various sets of experiments.

1.3 Problem Statement and Significance

The goal of this dissertation is to build a normative structure, suitable for real-world usages. In order to get to this point, several milestones are defined:

Creating a realistic agent-based model As introduced earlier, to create such a model, a new hybrid method of agent-based modeling and Markov Chain Monte Carlo is employed. Based on this method, an agent-based model is constructed to generate simulated data which is then used to initialize the proposal distribution of the MCMC. The combination of the two models, agent-based and MCMC, produces a more accurate result than either of the parent models and facilitates the MCMC convergence. An additional benefit is that manipulating the operation of an agent-based model can empower researchers with better intuitions about the reasons behind emerging group phenomena rather than merely observing the unfolding of a stochastic process [134]. To demonstrate the strengths of this approach, a case study on modeling and predicting transportation patterns and parking lot usage on a large university campus (UCF) is presented.

Creating the normative structure This dissertation describes a lightweight architecture along with a more complicated BDI version for simulating normative effects using agent-based models. The overarching aim of this research is to create a general purpose agent-based modeling (ABM) and simulation system for studying the effects of public policy decisions on a large range of social phenomena, including personal health decisions, sustainability behaviors, and opinion formation. In addition, we employ some ideas from the theory of tipping points to show how they can be applied to the computational models of normative behaviors.

1.4 Overview

This dissertation is organized into the following chapters:

Theoretical background: In this chapter, a set of a general definitions, terms and assumptions useful for remaining chapters are presented. A concise introduction is given for the agent-based modeling method, Markov Chain Monte Carlo, category theory and also the theory of tipping points. In addition, some abbreviations used in this dissertation are introduced.

Related work As the name of this chapter implies, the chapter is dedicated to reviewing current literature. This includes similar applications of agent-based modeling, and also applications of the Markov Chain Monte Carlo technique in modeling. Moreover, several similar works that use ideas from category theory are presented. A detailed review of related work on norms, specifically norms in agent-based societies is provided. Normative structures and architectures are also discussed here.

A hybrid modeling approach for parking and traffic prediction in urban simulations The key elements of our urban simulation for forecasting transportation patterns and parking lot utilization are summarized in Section 4.1. Section 4.2 presents our hybrid approach for unifying agent-based and MCMC models.

Analyzing agent-based models using category theory This chapter presents a case study in the usage of category theory for comparing different variants of an urban simulation system, designed to study traffic congestion and parking scarcity on a large university campus. Here, category theory is initially used to represent our agent-based model and a Markov Chain Monte Carlo sampler that can be combined with survey data to estimate quantities of interest. This chapter describes how category theory can be used to represent the relationship between the two models and how insights from the category theory representation can facilitate the creation of hybrid modeling methods.

A normative agent-based model for predicting smoking cessation trends Chapter 6 presents our first lightweight normative architecture (LNA). This architecture is introduced in conjunction with describing the required components for our smoking model. This chapter presents an ablative study showing the relative contribution of the different layers of the ABM on predicting the impact of a smoke-free campus initiative on student smoking cessation behavior. Our proposed model to simulate smoking behaviors includes three factors: 1) personal values, 2) social networks, and 3) environmental influences; a detailed description is provided in Section 6.3. The norm in the smoking case study is the acceptability of smoking on a smoke-free campus. Agents modify their beliefs based on a combination of personal, environmental and social factors. The normative model is operationalized as part of an activity-oriented microsimulation of transportation patterns on a large university campus. Inclusion of a detailed transportation model facilitates simulating propinquity effects that arise from physical proximity. Section 6.5 presents results on the performance of our model at predicting smoking cessation attitudes. Although this chapter focuses on smoking behavior, the architecture is sufficiently general to permit the study of a variety of public policy scenarios.

Here we seek to integrate normative effects with other types of human behavior models to produce a more comprehensive picture of human communities, rather than limiting our analysis to norms alone. Hence the proposed ABM simulates both environmental and network effects, in combination with norms.

Modeling norm emergence with the cognitive social learner architecture The new architecture for modeling emergence of social norms in societies (CSL) is introduced in this chapter. The performance of CSL is evaluated on an abstract case-study first. After that, CSL is applied to the problem of modeling smoking behavior of students at UCF. The same smoking model which was introduced in the previous chapter is used to apply CSL to our smoking case study. The results for CSL are compared to the results obtained by the lightweight normative architecture presented in previous chapter.

Modeling tipping point theory using normative multi-agent systems This chapter proposes that normative multi-agent systems (NorMAS) can serve as excellent computational models for modeling and predicting tipping points. The process of norm emergence in these systems is analogous to the social epidemics that occur at tipping points. Tipping points occur when a large number of group members radically modify their behaviors in response to small but significant events; after a critical point is reached, the behavior of the entire social system changes irrevocably. Sociologists have attempted to categorize common triggering factors for these tipping points. The chapter illustrates how tipping point theory can be modeled with a standard social learning approach and replicate some of the key findings.

CHAPTER 2: THEORETICAL BACKGROUND

Since this dissertation uses several different methodologies, a set of definitions is provided here. This chapter provides background on agent-based modeling, Markov Chain Monte Carlo, category theory, social norms and finally tipping point theory.

2.1 Agent-based Modeling

Agent-based modeling (ABM) is a technique of modeling which looks at the problems using a bottom-up approach in which the system is modeled as many interdependent components rather than a single overarching set of mathematical equations. The main idea in agent-based modeling is that by defining a population of agents, and defining rules governing the behavior of agents, complex notions that are hard to model emerge from the system. This way, the key challenges in designing agent-based models are defining a set of agents with appropriate properties, and more importantly defining proper rules. For instance, while designing an agent-based model for studying the effects of a certain virus on tissue cells, it is important to equip the agents representing the cells with abilities consistent with the behavior of real cells. The set of required rules could relate to the cells' movement abilities, the way they interact with other cells and the characteristics determining the end of their lives. Outcomes in ABMs can be equilibrium points or distributions or complex patterns. Instead of pre-planned outputs, the outcomes of agent-based models emerge from the interactions among agents [61]. ABM has been successfully applied to a long list of different domains.

2.2 Markov Chain Monte Carlo

Markov Chain Monte Carlo is a family of methods principally used to perform Bayesian inference with stochastic simulation. The aim of the process is to approximate the posterior distribution of the model parameters based on the observed data. By using Monte Carlo simulations to perform the high-dimensional integration necessary to calculate marginal and posterior distributions, algorithms such as Metropolis-Hastings (MH) can make the Bayesian inference process tractable [129]. The MH algorithm is the oldest and perhaps most commonly used of these methods. The basic procedure is as follows:

- Select a proposal distribution Q (also known as the proposal transition matrix)
- Initialize the starting point, x_0
- Do
  - Generate a candidate point x_c, according to the probability $Q(x_c|x_i)$
  - Calculate the acceptance probability according to

$$\alpha(x_i, x_c) = \min(1, \frac{\pi(x_c)q(x_i|x_c)}{\pi(x_i)q(x_c|x_i)}) \tag{2.1}$$

  - Choose $x_{i+1} = x_c$ with probability α, $x_{i+1} = x_i$ with probability $(1 - \alpha)$

Effectively MCMC allows us to draw samples from a distribution $\pi(x)$ without having to know its normalization. With these samples, it is possible to compute any quantity of interest about the distribution of x, such as means, confidence regions, or covariance.

2.3 Category Theory

In order to reach the point that we can define our desired representation using category theory, we need to briefly introduce the required structures. For a detailed overview of category theory elements the reader is referred to [106] and [15]. Category theory is an extensive mathematical theory which focuses on the relations of objects than the objects themselves. Basically, category theory provides its user with various abstraction mechanisms. These abstractions make it possible to show relations among objects that might seem very different from each other. For instance, using a set of abstraction techniques in category theory enabled the solution of hitherto unsolved problems in algebraic topology [74]. The basic structures that are defined in category theory are the category itself, arrow, and functor.

• A category **C** consists of: 1) a set of objects $(A, B, C, ...)$, 2) a set of arrows $(f, g, h, ...)$ also known as morphisms, 3) a way to compose arrows (composed arrows are also associative), 4) identity arrows. Each arrow has a unique source or domain and a unique target or codomain[1]. Figure 2.1 shows a simple category containing objects A, B, C, D and the arrows f, g and h. The identity arrow for object A and composite arrow of f and g are shown in this figure.

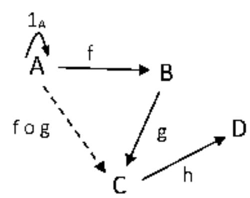


Figure 2.1: A simple category containing objects A, B, C, D, morphisms f, g and h, identity arrow 1_A and composite arrow $f\ o\ g$

• A functor $\mathbf{C} \longrightarrow \mathbf{D}$, shown by F, is a mapping from objects to objects and arrows to arrows of category **C** to category **D**. For the objects and arrows, we define $F(f : A \longrightarrow B) = F(f) : F(A) \longrightarrow F(B)$. In addition to domains and codomains, functors preserve identity arrows and composition. Figure 2.2 shows a functor mapping category **C** to category **D**.

The focus in category theory is on relations rather than objects. Accordingly, various structures defining different types of relations at multiple levels are defined. Arrows show the relations among objects of a category, and functors show relations among categories. The relation among the functors is also shown by natural transformations. One could imagine natural transformation

[1] In case of a function, one can imagine codomain as the superset of range of that function.

doing the same to two functors between two categories, as what functors do to the objects and morphisms of two categories.

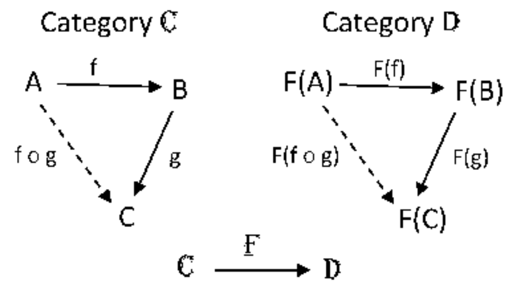
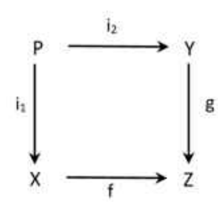


Figure 2.2: A functor going from category **C** to **D**

• Another key concept in category theory is universal property. Informally universal property refers to those set of properties that apply to all of objects in a category, and is the best and most effective set of properties they share; the idea of universal property directly relates to optimization in a system. Many ideas in category theory are based on the universal property concept such as limits, initials, products and their dual. The dual of each structure in category theory is constructed by reversing all of the existing morphisms.

• The other definition we need is the pullback structure. In the square



the morphisms i_1 and i_2 plus the object P are called the pullback of morphisms f and g. If the pullback is a universal property, there should be a unique morphism between object P and any other object like Q that is the domain of two morphisms to X and Y (Figure 2.3).

• The last structure that will be introduced here is adjoint functors. Since the main contribution of this work is presented using this structure, we will provide more details about it. Category theory excels at expressing weaker types of equality in a mathematical language. Imagine we have two categories **C** and **D**, and two functors F and G between them, as Figure 2.4 shows.

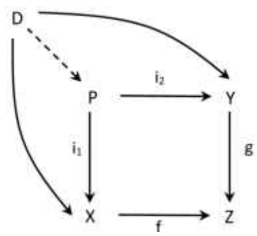


Figure 2.3: Pullback of morphisms f and g that has the universal property.

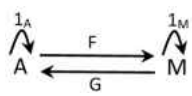


Figure 2.4: Two categories **C** and **D**, and functors F and G between them

A hierarchy of relations could be defined between these two categories as shown in Table 2.1. As the table illustrates, equality refers to the classic relation of two items that is quite rigid. It simply means that they are the same ones. Isomorphism is more lenient than equality and states that going from category **C** to category **D** and then returning (GF) is equivalent to remaining at **C** $(1_{\mathbf{C}})$. The same thing exists for category **D**. Descending the hierarchy, we reach an equivalence which says that going from **C** to **D** and returning (GF) is isomorphic to the identity functor or $1_{\mathbf{C}}$. Finally, adjunction is even weaker than the other relations. It says that there exists a natural transformation from the path starting from **C** to **D** and returning to **C** (GF), to the identity functor $(1_{\mathbf{C}})$. A similar natural transformation exists for the other case.

2.4 Norms

Although norms are ubiquitous around us; they are complicated and challenging to be studied and modeled. Here, a set of definitions that will be used in later chapters are provided.

Table 2.1: Hierarchy of relations between two categories $\mathbf{C}$ and $\mathbf{D}$, in terms of equality. F is a functor going from $\mathbf{C}$ to $\mathbf{D}$, and G is dual of F. $\cong$ shows isomorphism and $\Rightarrow$ shows natural transformation relations.

Relation	Equality	Isomorphism	Equivalence	Adjunction
Meaning	**C = D**	$1_{\mathbf{C}} = GF$ and $FG = 1_{\mathbf{D}}$	$1_{\mathbf{C}} \cong GF$ and $FG \cong 1_{\mathbf{D}}$	$1_{\mathbf{C}} \Rightarrow GF$ and $FG \Rightarrow 1_{\mathbf{D}}$

- Norm: "A norm is any behavioral rule that is considered valid by the majority of a population" [98].
- Social Norm: "A social norm is a rule of conduct derived from a social behavioral expectation" [77].
- Moral Norm: "A moral norm is a rule of conduct derived from a moral value" [77].
- Legal Norm: "A legal norm is a rule of conduct derived from the code of law" [77].
- Internalization: The process of acceptance of a set of norms and values established by people or groups which are influential to the individual through the process of socialisation. [130]
- Recognition: Refers to the ability of an agent to infer regulatory standards, conventions and norms of a society via observation and interaction with individuals. It also plays a role in monitoring norm-abiding behavior and detecting deviations [120].
- Adoption: Norm adoption is the process of an agent accepting new norms that will influence its practical reasoning. Adopting a norm does not mean that an agent will automatically comply with it (in fact, it may choose to violate norms) [120]. An agent accepts (adopts) a norm only if it believes that this norm helps in a direct or indirect way to achieve one of its goals [54].
- Compliance: Is a phase in norm development that an agent decides to comply with a norm

and possibly modify its goal according to the norm. Some of the agent's goals might conflict with the norm, but the agent usually has a computational process for determining whether it's worth complying or not.

- Normative multi-agent system (NorMAS): "A normative multi-agent system is a multi-agent system organized by means of mechanisms to represent, communicate, distribute, detect, create, modify and enforce norms, and mechanisms to deliberate about norms and detect norm violation and fulfillment" [33].

2.5 Tipping Point Theory

The term, "tipping point", was initially coined in physics to describe the situation in which the state of an object rapidly changes from one stable equilibrium to another different equilibrium. Morton Grodzins was the first to use this term in social sciences for describing an interesting phenomenon he observed in some US cities, known as *white flight* [90]. His observation was that in some metropolitan areas, the percentage of African-American people increases up to a certain point. After that point, those with white ethnicity immigrate from those cities in large numbers. Later, Thomas Schelling presented *the general theory of tipping*, which describes how individuals' micromotives and microbehavior can aggregate in the big picture [151]. Similarly, the model of collective behavior that was introduced by Mark Granovetter [89] uses thresholds to determine the path of social events. This model was initially used to describe how fads are created.

In normative studies, tipping points are usually denoted as the point of maximum return at which time the behavior has the highest level of acceptability from the population. For instance, in a certain group of friends, the number of times they shower in a week may vary, but a specific value has the highest acceptability by group members as the conventional pattern of behavior.

CHAPTER 3: RELATED WORK

Many works exist both on agent-based modeling and normative systems. ABM's simplicity in design and implementation makes it an interesting choice for researchers studying different domains from simulating the epidemic spread of Ebola virus [128] to modeling people living in ancient civilizations [110]. Here we focus on those types of work that use ABM for studying urban and transportation simulations. Next we describe the set of work that uses the Markov Chain Monte Carlo technique for modeling and simulation. Several examples of the employment of category theory ideas are presented to show how researchers use category theory to show the relations among different concepts. A review of the state of the art on normative studies concludes the chapter. The different aspects of norm life-cycle including emergence, adoption and compliance are reviewed. Our main focus will be on two issues: 1) What are the major components needed to build a cohesive and complete normative model? 2) What are the current architectures for normative agent-based models?

3.1 Agent-based Modeling

Agent-based models are a popular modeling and simulation technique due to their ease of construction [123]. The modeler simply defines a population of agents with specific properties, plus a set of rules governing the agents' behavior and decisions. It is relatively simple to rapidly prototype a complex system with emergent behaviors, even without a formal specification or complete knowledge of the system dynamics. ABMs have been applied to a range of interesting real-world problems ranging from modeling people's transportation selections to simulating the response of an organ's cells to a bacterial attack [96]. Yet, the lack of mathematical formalism can make the results of ABMs hard to validate and also render them difficult to reproduce. The results of a simulation study can vary considerably by changing the range or even the step size of just one or two variables [133]. According to [169], most of the works based on ABM, with the exception of a few classic models, have never been replicated by anyone other than the original developer.

Agent-based modeling has been used to analyze a variety of complex public policy related scenarios including climate change negotiations [84], water management decisions [118], and financial regulatory governance [157]. In general, ABMs are good at modeling a diverse population of rational, self-interested agents, allowing interesting social questions to be explored in simulation before enacting new laws. For instance, Garlick and Chli studied the effects of social influence and curfews on civil violence by creating an agent-based model that simulated the interactions between the police force and the community [82]. Some social simulations explicitly model network interactions between agents; for example this is useful when studying influence propagation [124] and the self-repairing properties of insurgent terrorist networks [100]. Social choice mechanisms can be studied using agent-based simulations as well as by game theory; for instance, Verella and Wardak examined the effects of external stimuli on collective opinion formation, in the context of voting decisions [162].

Alternatively, interactions between agents can be governed by a combination of spatial and social constraints; in these social systems the behavior of the agents is strongly affected by other agents in their local physical neighborhood, which is easy to simulate but often difficult to predict analytically. Examples of systems possessing these characteristics include traffic and crowd evacuation simulations, which are heavily influenced by geographic considerations [5, 142]. To compare the effectiveness of simultaneous and staged evacuation strategies in different road network structures, Chen designed an agent-based simulation that shows the collective behaviors resulting from the interactions of individual vehicles during an evacuation [47]. Human behavioral data can be added to the emergency evacuation and egress model to build a more realistic and consistent agent-based model as was done by [136].

In contrast to crowd evacuation scenarios which are often used to prepare for unique disaster situations, traffic simulations are designed to characterize the effects of repetitive behaviors.

Klügl and Bazzan [109] outline five advantages agent-based methodologies have over other types of traffic-related simulations including: 1) ease of modeling bottom-up decision-making, 2) capacity for imbuing entities with learning and adaptive behavior 3) simplicity of generating a population with heterogeneous behaviors. Also it is often feasible to gather survey and GPS data to verify the predictions of traffic simulations [3].

Based on detailed trip survey data from seven Traffic Analysis Zones (TAZs) in Ottawa, Canada, Jin and White present an agent-based model for analyzing the influence of neighborhood design on daily trip patterns [103]. Results obtained from a behavioral survey of driving behaviors were used by Dia [64] to identify and fit a series of agent behavior parameters defining driver characteristics, knowledge and preferences; the authors also present a case study implementing a simple agent-based route choice decision model within a microscopic traffic simulation tool. However neither of those works presents a systematic evaluation of different modeling techniques through comparison with independently collected data. In our research, a physical path planning system for modeling driving and walking is used to supplement the activity-based microsimulation that governs agent behavior selection. The model is seeded with a combination of demographic information and survey data, and compared against independently collected results. A detailed review of the applications of agent-based modeling specially in modeling traffic and transportation patterns can be found in [46].

ABMs have been successfully employed in a variety of water management tasks [20, 80]. Water management, an important aspect of urban management, is affected by geography, weather patterns, and human behavior, and is additionally complicated by interdependencies between communities that share the same watershed area. Lopez et al. introduced an agent-based simulator called FIRMABAR for integrated freshwater assessment of the Valladolid metropolitan area [118]. The simulator provides the policy makers with a tool to evaluate alternative water policies in different scenarios.

Similar simulations can be used to study the combined impact of climate change and human behavior on sustainable ecosystems. Hailegiorgis et al. presented an agent-based system for modeling interactions between climate change and conflict among herders in east Africa [91]. ENGAGE is an agent-based model that was introduced by Gerst et al. to simulate the impact of locally heterogeneous policy preferences and constituent choice on climate change negotiation at the international level [84]. A review of related works in this area can be found in [18].

In summary, agent-based models can be used to illuminate policy makers on the ramifications of complex environmental and infrastructure decisions. For the case studies described in this dissertation, we use an urban transportation model that couples an activity-oriented microsimulation with path planning. Each agent represents a student on the UCF campus, and the population was created to match the data from a survey on student transportation, parking, and dining preferences.

3.2 Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) describes a family of methods for performing Bayesian inferences using stochastic simulation [26]. MCMC allows us to draw samples from a distribution $\pi(x)$ without having to know its normalization. Having these samples, it is possible to compute any quantity of interest about the distribution of x, including confidence regions, means, standard deviations, and covariance [141].

Markov Chain Monte Carlo has been successfully used in a wide variety of scientific [114] and engineering modeling applications [115]. MCMC is often utilized as an alternative to two other commonly used approximation methods:

1. importance sampling—samples are drawn from a distribution other than the target one, then reweighted to account for differences between the two distributions;
2. variational inference—the original integration problem is transformed into an optimization problem [40].

MCMC can also be applied as part of the model fitting process in social prediction problems. For instance, Cauchemez et al. use a Bayesian MCMC approach to examine the main characteristics that affect influenza disease transmission between households [45]. Similarly, the effect of spatial influences on geopolitical conflicts has been modeled using an MCMC formulation in which the likelihood of war involvement for each nation is conditioned on the decisions of proximate states [166]. In our work, MCMC is used as a simulation technique, and the sample set used to characterize the posterior distribution is simply compared against the output of other simulation techniques, rather than used to perform Bayesian inference over model parameters. In a recent similar work, a spatial agent-based model is calibrated with a Markov chain Monte Carlo approach [128].

Our research focuses on improving the performance of the Metropolis-Hastings (MH) algorithm which is relatively sensitive to the initial proposal distribution. It is because of this sensitivity that researchers sometimes opt to use alternative MCMC algorithms, such as Gibbs sampling [83]. Our proposed method is a variation on the idea of using suboptimal inference and learning algorithms to generate data-driven proposal distributions for the MH algorithm [6]. An alternate approach for creating MCMC proposal distributions was introduced by Eaton and Murphy [72] who employed dynamic programming to create a proposal distribution for MCMC in the space of directed acyclic graphs. They showed that this hybrid technique converges to the posterior faster than other methods, resulting in more accurate structure learning of graphical models and higher predictive likelihoods on test data.

De Freitas et al. [60] introduced two different methods to overcome the problem of finding a good proposal distribution. In the first approach, a mixture of two kernels is used to drive the search process: 1) a variational kernel to broadly explore the problem domain and locate regions of high-probability and 2) a Metropolis kernel to explore the local regions. One drawback with this method is that finding a good variational kernel can be difficult to do.

To combat this issue, the authors proposed a second technique called adaptive MCMC in which the proposal distribution is updated at run-time based on the behavior of Markov chain; here, we benchmark our proposed method against adaptive MCMC. Our approach solves the problem of identifying a good proposal distribution for MCMC by constructing one from samples generated by our agent-based model. Adaptive methods generally seek to construct a better proposal distribution by combining stochastic approximation and MCMC [7]. One issue with this class of adaptive techniques is that they often rely on certain mathematical assumptions being valid, and thus can only be used in a limited set of conditions unlike our technique.

3.3 Category Theory

In this dissertation, we attempt to relate agent-based modeling and Markov Chain Monte Carlo, as two families of modeling methods, using the abstraction language of category theory (CT). Category theory has been successfully used in several branches of mathematics, including geometry, algebra, and logic [34, 122]. But CT can also be used by researchers to describe physical and social systems. A historical review of CT applied to physics abstractions can be found in [17]. Coecke [50] asserts that category theory should become part of the daily practice of the physicist. Recently, Sallach [147] illustrated the benefits of categorical analysis within the social sciences by using CT to explicate several well known social theories. For instance, he shows how the equivalence and duality relations (structures in CT) can be used to explain Pareto's theory of the circulation of elites.

There has been some use of category theory within software engineering in which CT is used as alternate formal specification language. For example, in [145] Reynolds describes how the concepts of category theory can guide the design of a programming language to avoid anomalies in the interaction of implicit conversions and generic operators. The rigorous mathematical formalism of CT can empower software developers to reason about structures within their code [143]. In addition, it provides an exact notion of modularity and composition. Another major application area of category theory within computer science is data analysis. As an example, Kokar et al. [111] formally defines information fusion in category theory, and then shows how one can carry out formal reasoning about information fusion systems. Within machine learning, specific categorical constructs were applied to determine neural structures for the re-design of a neural network [95]. By using ideas from category theory, our aim is twofold: 1) to use category theory to provide a formal representation for our ABMs and 2) to use the mapping between multiple models to motivate the development of new hybrid modeling techniques.

3.4 Norms

Norms are an important key to understanding the function of human groups, teams, and communities; they are a ubiquitous but invisible force governing many human behaviors. Bicchieri describes human norms as: "the language a society speaks, the embodiments of its values and collective desires, the secure guide in the uncertain lands we all traverse, the common practices that hold human groups together." [30]

Norms have been studied in different fields, including sociology, psychology, biology and philosophy. In the computer science community norms are mostly used to organize the relations of agents and developing societies of agents. Some of the basic definitions of normative systems were presented in Section 2.4.

In this section, an overview of the process of creating social systems with normative agents will be provided, before describing the related work on smoking cessation. Various stages are introduced as elements of the norm life-cycle including creation, identification, spreading, recognition, enforcement, acceptance, modification, internalization, emergence, forgetting, and evolution. Here, we will focus on the more important elements and introduce some of the key related work.

3.4.1 Norm Recognition

As introduced in Section 2.4, norm recognition refers to the ability of an agent to infer regulatory standards, conventions and norms of a society via observation and interaction with individuals [120]. Based on this definition, recognition is considered as the opposite to imitation as two major techniques toward norm emergence.

It is worth noting that some references refer to an earlier stage before recognition is introduced as norm creation. This refers to how the norm starts to develop from its very beginning. In [148] three ways for norm creation are presented: offline design, leader agent initiation, and entrepreneur agent initiation.

Similar to the norm creation stage, another stage can be studied which has a direct relation with norm recognition, namely norm spreading or transmission. Three core components that make this possible are: agent relationship, transmission technique, and connectivity structure [98]. These three components mainly relate to the way that agents are connected and how they pass messages or promote certain behaviors.

3.4.2 Norm Adoption

Norm adoption and compliance are key to the study of normative agents. The general assumption behind norm adoption is that an agent will adopt another agent's goal, on the condition that the adopter comes to believe that the achievement of the adoptee's goal will increase its chances of achieving a previous held goal [10]. Castelfranchi describes two types of norm adoption: 1) instrumental, in which agents are motivated to obey a norm that benefits them and 2) terminal, which implies that the agents do not have any other choice other than following the norms [43].

Norm adoption can be illustrated with the Iterated Prisoner's Dilemma by dividing agents into one or more groups and assigning an IPD strategy. The agents play against one another until one strategy appears to be stable. Then, a different strategy can be introduced into the stable system before play resumes [98].

3.4.3 Norm Compliance

Norm compliance usually refers to the process by which a normative belief becomes a normative goal [43]. Four types of theories are introduced in [10] for implementing norm compliance: 1) agents follow norms because it is individually rational. Agents comply with norms when the costs of violation exceed the costs of compliance. 2) Agents' choice is dependent upon what the other individuals do (empirical expectations), and upon what the others expect should or ought to be done (normative expectation). This is often referred as social conformity. 3) Agents show normative behavior automatically and without any deliberation about which action they should choose. 4) Norms are internalized within agents' minds through the internalization process. In this case, compliance is seen as a product of internal sanctions that agents impose upon themselves.

Note that adoption is not synonymous with compliance in norms. An agent may adopt to a norm but choose to violate that norm later. For instance, agent transgressions can occur when the expected rewards obtained with detection surpass the expected rewards obtained by being norm-compliant [78].

3.4.4 Norm Enforcement

The existence of norm conflicts raises the possibility of norm violations. In normative studies, two types of approaches are generally employed to handle violations: punishments and sanctions. Punishment is usually performed by imposing some type of cost on agents. On the other hand, punishments when the economic incentive is combined with the communication of normative information about the prescribed conduct are more effective [12]. This type of punishments is usually referred as sanctions. As Villatoro et al. report, sanction is more effective and less costly than punishment in the achievement and maintenance of cooperation, and it makes the population more resilient to sudden changes than if it were enforced only by mere punishment [163]. Norm enforcement is sometimes implemented through reputation as well [92].

A closely related idea which is frequently referred in normative studies is *deterrence*. Deterrence is usually implemented based on theories of crime. For instance, a distributed mechanism is proposed in [62] to enforce norms by ostracizing agents (as a deterrent) that do not abide by them.

As Andrighetto et al. points out, norms may be conditioned on a variety of factors including spatial, temporal, cultural and social circumstances [12]. Norm violation is the byproduct of having a flexible norm system. In a hard-wired system in which the norms are fixed and the agents must comply, it is impossible to have violation and conflicts. Accordingly, various conflict resolution techniques have been used in the literature. Some of these methods are similar to the techniques used in general multi-agent systems, but many are specific to normative domains. For instance, a meta-norm usually refers to a higher level norm that agents consult in case of conflicts. A meta-norm can be as simple as selecting a norm at random when a conflict occurs or can be a much more complex resolution procedure. Norm conflict can be also dealt with using argumentation-based approaches [135].

3.4.5 Norm Emergence

A fundamental research question is how norms emerge in social systems. Norm emergence is usually defined as a stage during which a certain portion of agents has accepted a norm and follow it. Some of the existing techniques for norm emergence are based on game-theoretical ideas. These techniques are similar to the algorithms that implement coordination or cooperation in agent societies. In these domains the assumption is that cooperation or coordination emerges when a sufficient number of agents play the same strategy. For example, one approach is to model this phenomenon through the use of learner agents that adapt their behavior based on sanctions and rewards. Sen and Airiau's work [155] in this area, in which agent interactions are modeled using payoff matrices, inspired much subsequent research on norm emergence through social learning in agent societies. A recent extension which adds network structure to the social system is described in [172].

3.4.6 Existing Normative Architectures

Various normative architectures are presented by researchers for different purposes. Some of these architectures will be introduced here, and some of them will be introduced in Section 3.5.

One of the pioneering architectures in area of normative multi-agent systems was the deliberative normative agents architecture [44]. According to this architecture, violating norms can be considered as acceptable as following them. Agents deliberate about the norms that are explicitly implemented in the model. Also, agents use the norms to change their goals, and later their plans. A different approach to normative reasoning, a norm-oriented agent, is presented in [137]; this agent takes into consideration operationalized norms during the plan generation phase, using them as guidelines to decide the agent's future action path. Also in [117] a normative architecture is proposed for self-interested agents allowing them to perform a type of normative reasoning to evaluate the positive or negative effects of these norms on their goals.

Boella and van der Torre [31] presented the idea of having two major components in a normative multi-agent system. The first part relates to the agents that should behave based on the current norms. These agents are called defenders. The second part is related to the agents that monitor the behaviors of other agents and sanction violators, who can also change norms as needed. The authors also show that these two parts could be implemented on the same set of agents; meaning that agents can simultaneously serve as defenders and controllers.

These authors later extend their work by adding logical components to their model [32]. They show how the architectural approach can be used to develop a logic of a normative system based on logics of counts-as[1] conditionals, institutional constraints, obligations and permissions.

[1]Counts-as relation expresses the fact that a state of affairs or an action of an agent "is a sufficient condition to

Counts-as conditionals and institutional constraints are defined as a pre-processing step for the regulative norms. In this work, permissions are defined as exceptions to obligations and their interaction is characterized.

Logical representation of norms have been used in other works too. Garcia et al. [81] proposed means to specify and explicitly manage the normative positions of agents (permissions, prohibitions and obligations), with which distinct deontic notions and their relationships can be captured. The rule-based formalism they present includes constraints for expressiveness and precision and allows the norm-oriented programming of electronic institutions: normative aspects are given a precise computational interpretation. Another architecture that uses logical representation is presented by Sadri et al. [146]. The logical model of agency known as the KGP model was extended in this work, to support agents with normative concepts, based on the roles an agent plays and the obligations and prohibitions that result from playing these roles.

Focusing on cognitive abilities of agents, the emergence of norms is viewed as intrinsically intertwined with the emergence of normative beliefs in [53]. The process of emergence is seen as a non-continuous phenomenon. Here, a given behavioral regularity is argued to give rise to a normative belief as long as that regularity is believed to be prescribed within the community. The spreading of norms is not only due to a passive behavioral social influence (imitation) but also to an active cognitive one (the spreading of normative wants and beliefs). Also a norm is not necessarily explicitly and deliberately issued by some normative authority, but is grounded upon the norm-addressees' beliefs that they are generally prescribed to comply with it. This architecture later led to the design of the EMIL architecture. The EMIL architecture is one of the most elaborate normative architectures described in the literature. This architecture is introduced in [9] as well as other publications. This architecture defines two sets of components for each agent: 1) Epistemic, which is responsible for recognizing norms, and 2) Pragmatic, which is responsible for the

guarantee that the institution creates some (usually normative) state of affairs." [104]"

agent's behavior based on the normative representation. The architecture has been applied to some abstract scenarios (without usage of real data) including modeling traffic, simulating conflicts in Wikipedia, and modeling financial decisions [11]. Using the EMIL architecture in real scenarios can be challenging due to the elaborate design of its cognitive mechanisms.

Many existing normative architectures are based on the BDI (belief, desire and intention) structure. These architectures are usually extensions to the BDI structure. Probably one of the best examples of such architectures is the BOID architecture [35]. BOID extends the classic BDI approach to include the notion of obligation. As another example, a combined approach to identify objectives for an architecture for self-regulating agents is proposed in [37]. Here, authors assess how changes on the inter-agent level affect the intra-agent level and how a generic BDI architecture IRMA can be adapted for self-regulation. A complete survey of normative architectures including a detailed classification of them can be found in [131].

3.5 Two Lines of Research on Normative Models

The following sections present an overview of cognitive (internal) and interaction (external) normative systems. These two lines of research on normative systems are mentioned in many references. For instance, Neumann [11] distinguishes between these two categories as being divided into models that are inspired by the conceptual terminology of game theory and models that are based on architectures of cognitive agents with some roots in artificial intelligence. Neumann notes that the main deficit of both approaches is a lack of a dynamics to describe interactions between cognitively rich mental objects. While game theoretic models are dynamic, norms are typically regarded as merely the aggregated product of individual interactions. Thus they lack the concept of mental objects. Cognitive models on the other hand, include mental objects, however, these objects are static and have a limited concept of normative obligations.

Similarly, two approaches based on rational choice theory were introduced in [77]: 1) methods that focus on the individualism aspect of agents' design and 2) approaches based on strategic interaction games. In this work, it's also noted that both rational choice theory and game theory are excellent raw material for agent-based models. Both propose an individual decision mechanism in the form of a utility calculation, thus providing the micro-level for an agent-based model. Iterations of many agents executing their rational or strategic decisions lead to macro-phenomena such as conventions or norms. As these iterations are too complicated or complex to execute on paper, agent-based models can provide a 'laboratory' in which to conduct experiments.

Savarimuthu and Cranefield [148] refer to these two categories of approaches for designing normative architectures: philosophy of law (prescriptive approach) and conventionalistic approach (emergence approach) [53]. Based on these two perspectives, research in normative multi-agent systems can be divided into two categories. The first category deals with normative architectures, representation of norms, adhering to norms and the related reward or sanction measures. The second category focuses on the emergence of norms.

Moreover, acquiring norms through a social learning process when an agent interacts with its environment, is one way of implementing norms [98]. In sociology, this process is known as socialization; in anthropology it is called enculturation. The other way is when norms are socially enforced through external sanctions or other measures until they become internalized by an agent. Once internalized, norms are enforced primarily through internal mechanisms.

3.5.1 Cognition-based Approaches

These methods provide high-fidelity models of the cognitive aspects of normative behavior, while focusing on the internal part of the norm lifecycle [76]. In comparison with the interaction-based models described in next section, this category relies less on the use of reward and punishment to motivate norm adoption, moving beyond the carrot and stick approach [13]. For instance, the EMIL architecture includes a dynamic cognitive model of norm emergence and innovation [52]. The main disadvantage of EMIL is that the agents obey all recognized norms blindly without considering their own motivations [57]. However, these architectures can model norm internalization in which agents manifest behaviors, not because of existing rewards or punishments in the environment, but as a personal objective [14].

Norm internalization is sometimes implemented via emotions [56] and is very closely related to deliberation. Dignum et al. (2000) presented an architecture that allows agents to use deliberation to decide when to follow or violate norms [66]. The agent generates behavior by creating and selecting goals on the basis of beliefs and norms, before choosing actions and plans according to the selected goals. The deliberation can also be implemented with a modified BDI interpreter loop that takes norms and obligations into account [66]. A weakness with these models is that they devote less attention to norm emergence at the population level.

Like our proposed CSL architecture, several existing normative architectures also use BDI reasoning as a core component. For instance, the BOID architecture [35] adds the notion of obligation as a fourth element to the original belief, desire and intention model. Normative BDI [55] extends the multi-context BDI architecture [156] which includes two new functional contexts (planner and communication) to support normative reasoning with additional contexts (recognition and normative).

3.5.2 Interaction-based Approaches

Interaction-based approaches create agent models that can detect norms from what they observe in the environment and their interactions with other agents. Often the agents are equipped with the ability to learn from experience, and interactions among agents are modeled as repeated games with payoff matrices. The simplest interaction approach is to imitate other agents in the environment—"while in Rome, do as the Romans do." For instance, Andrighetto et al. (2008) present a normative model in which the agents mimic the majority behaviors; this type of agent is commonly referred to as a social conformer. Generally these imitation agents lack high-level reasoning and decision making abilities.

Social learning [155] offers a richer model of norm emergence. In social learning, agent interactions are modeled as a staged game (the social dilemma game). A norm emerges when the entire population's actions converge to the same action, based on updates to the payoff matrix specifying the reward for the possible actions. Several variants of multi-agent reinforcement learning have been demonstrated for this interaction model. However, a general concern that exists about this family of repeated game interaction models is that 1) they do not capture many of the rich interactions that take place in real world scenarios and 2) can fail to converge when the agents have a large action-space [12]. In this dissertation, we show that our CSL architecture is more robust against increases in action space size.

Although reinforcement learning is popular for interaction-based approaches, other machine-learning/data-mining techniques have also been used. For instance, Savarimuthu et al. (2010) use an association rule mining technique to identify obligation norms, allowing the detection of norms through an examination of interactions among agents. These agents are able to identify conditional norms, norms that exist when some specific criterion holds. Markov decision processes have been used to create a reward-based model of norm compliance; transgressions occur when the expected rewards from norm defection surpass the expected rewards obtained by being norm-compliant [78].

3.6 Tipping Point Theory

Much existing work in normative multi-agent systems explicitly or implicitly relies on social science theories. In a recent work, some of the well-known theories of philosopher David Hume were evaluated using an agent-based model called $\text{HUME}_{2.0}$ [52]. This work demonstrates how social justice concepts can even emerge from heterogeneous agents that are not endowed with norm representations.

Self-determination theory is also referenced by some of the normative works. Here the focus is on the agents' motivation and the extent to which the motivation is intrinsic or extrinsic. Neumann studied existing normative architectures to see how much they comply with self-determination theory [132].

Practice theory is an example drawn from anthropology; this theory describes how changes in the society are based on the interactions between the human agents and social structure. For instance, an agent-based model for energy demand and supply social practices is presented in [19], which shows how energy consumption norms form and evolve in urban societies.

Similar to our usage of ideas from the social networks analysis domain, a model of norm emergence and innovation in language change is presented by Swarup et al. [158]. This work introduces a model of linguistic diffusion in social networks, to analytically derive time to convergence, and to account for the innovation phase of lexical dynamics in networks.

CHAPTER 4: A HYBRID MODELING APPROACH FOR PARKING AND TRAFFIC PREDICTION IN URBAN SIMULATIONS

In the first chapter which includes the main content of this dissertation, a hybrid method of modeling will be presented. As the name of this proposed method (ABM-MCMC) implies, it works by mixing the two classic methods of agent-based modeling and Markov Chain Monte Carlo.

The new hybrid modeling approach leverages the strengths of two existing techniques, agent-based modeling (ABM) and Markov Chain Monte Carlo (MCMC) estimation, for constructing large-scale population models. Rather than trying to change way that these two methods work, we show how the two methods can be mixed such that a single method that can show a better performance is created. The proposed method resolves the proposal distribution difficulty that affects the performance of most MCMC methods by using the ABM to initialize the proposal distribution. Figure 4.1 shows an overall view of how the proposed hybrid model works.

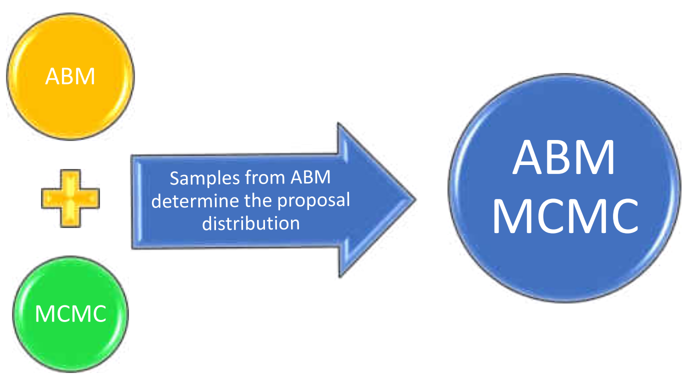


Figure 4.1: A schematic representation of the hybrid method: ABM-MCMC

Agent-based modeling is well suited for modeling and simulating large systems with emergent interactions that are not easy to characterize analytically. However, ABMs often suffer from the following issues:

- Validation
- Reproduction
- Formal representation

On the other hand, Markov Chain Monte Carlo (MCMC) describes a family of methods for performing Bayesian inference through stochastic simulations of a Markov process. Unlike ABMs, MCMC estimation is easy to describe and reproduce. However MCMC suffers from specific issues regarding:

- Mixing problem
- Proposal distribution

Here, we modify the proposal distribution used by the Metropolis-Hastings (MH) algorithm. We demonstrate the benefits of the proposed method at forecasting transportation infrastructure utilization on the UCF campus. In the next section, details about the designed agent-based model for simulating transportation patterns of students is presented. This agent-based model will be used in future chapters also.

4.1 Urban Simulation

In this section, we describe the development process for our activity-based microsimulation, including the agent-based model, survey data collection, activity profile generation, path planning, and simulation system; see [24] for additional details on the data collection and model fitting procedures. For our urban region, we selected the University of Central Florida main campus, which is one the largest academic institutions in the US with almost 59,000 students and 10,567 staff. It is adjacent to the Central Florida Research Park which is home to 116 companies with approximately 9,500 employees. The presence of nearby businesses and existence of commuters traveling between multiple UCF campuses give rise to a social system with a diverse and complex set of transportation patterns.

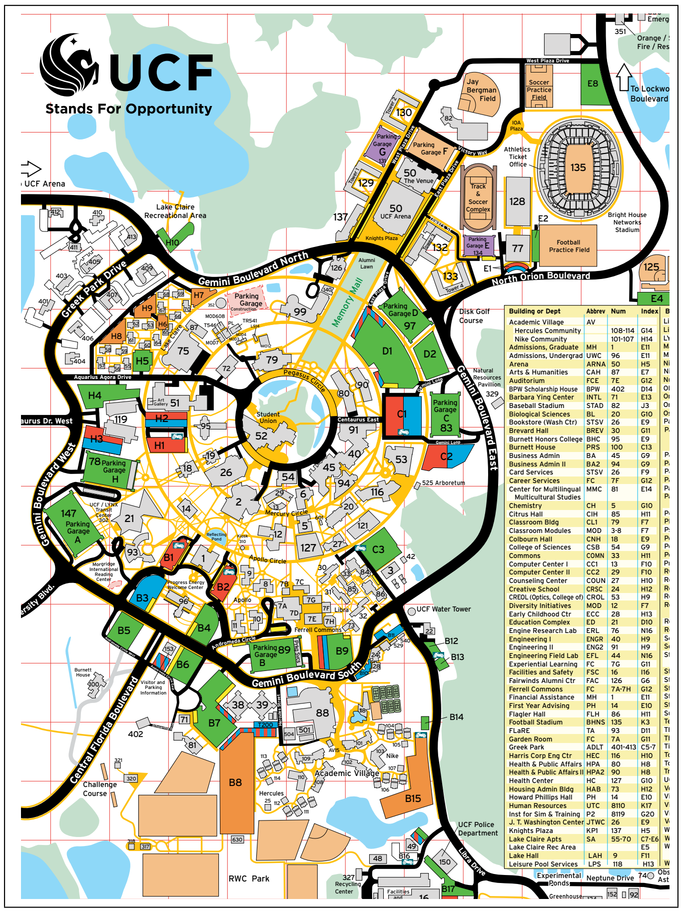


Figure 4.2: The map used in the simulation. Gray spots are buildings, black lines show the campus roads, and yellow lines indicate the walkways. Parking lots are marked in green (student), blue (staff), and red (faculty).

4.1.1 Data Collection

To simplify the data collection process, our initial study focused solely on modeling student transportation, dining, and building occupancy patterns. 1003 students responded to our online survey posted on `KwikSurveys` which was advertised on various campus email lists. The questions on the survey were grouped into six different categories, related to possible places that could be visited on the main campus:

1. Daily attendance patterns, including the days and times that the participant arrives and departs the main campus
2. Initial location, either the dorm (for on-campus students) or the entrance that was used to enter the campus (for commuting students)
3. Visitation frequency for on-campus dining locations
4. Usage patterns for recreation and athletic facilities
5. Usage of administrative and other miscellaneous locations
6. Frequency of parking lot and shuttle stop usage

For categories three through six, students were specifically queried about their visitation frequencies. For these questions, responses included one of: *never*, *rarely*, *once a month*, *several times in a month*, *once a week*, *several times in a week* and *every day*.

In addition to the survey data, our agent-based simulation used publicly available statistics about UCF[1] and the main campus building map[2]. A graph of the campus paths and roads was created from the main campus building map. The set of nodes in the graph is the union of the locations in the survey, plus the junctions between the streets and pathways. The edges of this graph represent the roads and walkways among the nodes. The weights of the edges show the distance between the connecting nodes. Each node and edge has a tag. This tag for the nodes indicates whether they are a location of interest on the map or merely a junction. Figure 4.2 shows a snapshot of the map, and Figure 4.3 shows the corresponding path planning graph.

[1]http://ikm.ucf.edu/facts-figures/

[2]http://map.ucf.edu/printable/

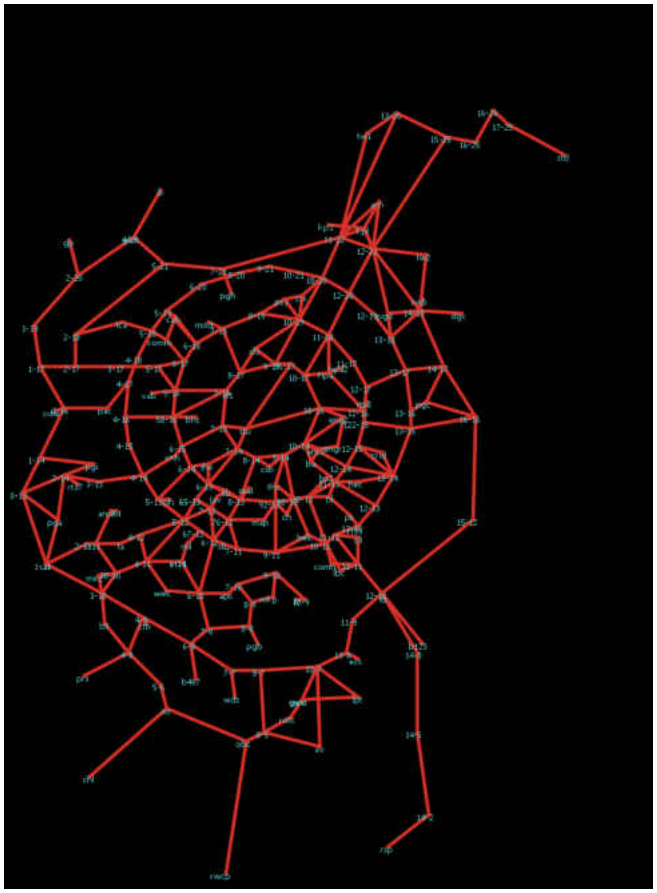


Figure 4.3: The corresponding graph to the map in Figure 4.2. The nodes represent different locations on campus, and the edges show the paths between them.

4.1.2 Agent-Based Model

To perform transportation forecasting on the UCF campus, we created an agent-based model for simulating the common activities (transportation, dining, recreation, and building occupancy) performed by the 47,000 students on the main campus. This number refers to the total number of students on the UCF main campus. Each agent in the model represents an individual student and has a unique set of parameters that govern his/her activity profile. An agent's defining parameters are: *entrance*, *dormitory*, *department*, *class building*, *arrive*, *depart*, *lunch*, *dinner*, *beverage*, *recreation and wellness*, *parking*, *shuttle*, and *miscellaneous*. The first four parameters designate the single (most common) value of the agents' entry point to the campus, housing situation, home department, and main class building. Note that we did not explicitly represent the students' class schedules in the model. Even though this would have improved the fidelity of the model, we felt that addition would not generalize well to other types of urban models. *Arrive* and *depart* are lists showing the times the agent enters the campus and leaves it. The remaining parameters are lists of locations for the agent's dining, recreation, and commuting. Additionally, each parameter that includes a location has another matching parameter that shows the time or frequency of visiting that location.

Rather than directly mapping the survey data to simulated entities that match the exact preferences of one of the survey respondents, we attempt to learn a general model of the population by fitting a statistical distribution to the answers of every question. For those questions that were related to the time of visiting a location (e.g., campus arrival and departure times), a Gaussian distribution was used to create a continuous distribution of arrival and departure times for the population of agents. For those questions where the respondents provided frequencies (e.g., how often campus dining locations were visited), we evaluated the performance of several discrete distributions and selected to the Poisson distribution as offering the best fit for most of the questions.

After fitting the Poisson distribution on the qualitative data, a mapping function is used to work with the values obtained. This function maps the qualitative frequencies to exact dates and times. Each term, from *rarely* to *everyday*, is treated separately. For instance, the term *rarely* is mapped to a random day in a 60 day period.

4.1.3 Activity-oriented Microsimulation

When the simulation commences, all the agents are initialized with parameters that remain constant over the lifetime of the agent and are used to create daily activity profiles. Our simulation is implemented in the Netlogo [168] environment.

In this environment, time is discrete and simulated by ticks where a tick is one unit of time. In our model, one tick represents one hour of activity in the real world. When the model starts, each agent runs within a loop. The loop continues until the simulation is stopped. Figure 4.4 shows the runtime process by which an agent activity profile is generated. In this loop, whenever it is determined that the agent should be somewhere on campus, it goes to the enable (visible) state, otherwise it goes to the disable (hidden) state.

```
switch current-time-status:
    case entrance-time
        if live-off-campus then
            enter-campus //go to one of the entrances
            go-to-parking-or-shuttle-stop
        end if
    case on-campus-time
        if should-go-somewhere then
            go-to-destination
        else
            stay-at-department
        end if
    case return-time
        if live-off-campus then
            go-to-parking-or-shuttle-stop
            leave-campus //go to one of the entrances
        else
            go-to-dorm
        end if
    case not-on-campus
        disable
```

Figure 4.4: Runtime generation of agent activity profiles

Based on the agent's parameters, the activity profile generator determines what should an agent do and where should be at every time (tick). If sampling the agent's profile indicates that it should be on campus, then the function compares the current time with the possible activity times produced by the mapping function that maps frequencies from the agent's distribution model to specific times and dates. If a match is found, then the agent opts to travel to that location. Otherwise, the agent remains at its department as its default place. On the other hand, if the profile generator determines that the agent shouldn't be on campus, then the agent goes to (or remains in) the disabled state.

Table 4.1: The parameter settings of ABM experiments

Parameter	Value
Agents	47,000
Days	100
Time Range	07:00 - 24:00

Various constraints are checked before an agent decides to go to a place. These constraints ensure the consistency of the whole model with the real world facts. The main consistency checks are summarized below:

- **daily schedule**: whenever an agent's model generates a date and time for visiting a location on campus, it checks the agent's arrival and departure times for that day. Campus activities that fall outside those boundaries are eliminated.
- **activity overlap**: whenever the agent's model generates trips that overlap in time, requiring the agent to be in multiple places at once, one of the overlapping tasks is shifted to a later time.
- **campus constraints**: known information about the operation hours of administrative offices, classroom buildings, and shuttle transportation is incorporated into the simulation. If

the agent's model generates trips that violate the known operation hours, those trips are discarded.

A shortest path graph algorithm is used to choose the path that an agent should traverse between its start and end positions. To speed-up the model, an all pairs shortest path graph algorithm computes all of the shortest paths. A slightly modified version of Floyd-Warshall algorithm [79] was used for this purpose. All path planning occurs at initialization; candidate paths are stored in a look-up table to be accessed later. The time complexity of Floyd-Warshall algorithm is $\theta(n^3)$. The parameter values used for all of the experiments are listed in Table 4.1.

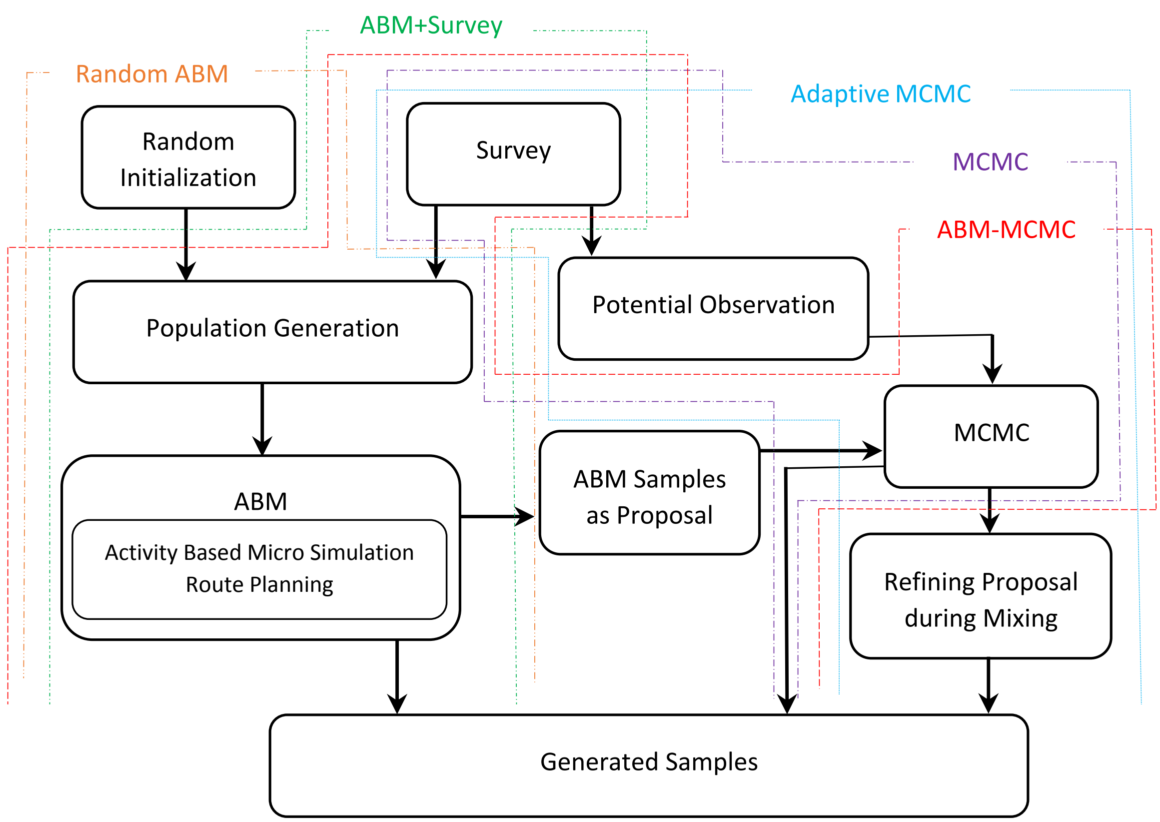


Figure 4.5: A flowchart showing the relationship between the various modeling methods.

4.2 ABM-MCMC

To evaluate the performance of our hybrid model, we compared the performance of our model against several other ABM and MCMC variants. Figure 4.5 shows the relationship between the different methods in a schematic way. In this figure, ABM-MCMC refers to the proposed hybrid method. In **ABM-MCMC**, the ABM is used to bootstrap the MCMC proposal distribution. In **ABM Random**, the agent-based model is initialized with a student population possessing randomly generated, but plausible schedules. In **ABM+Survey**, the survey data is used to create the distributions for generating agent activity profiles. **MCMC** employs the MH algorithm with a standard proposal distribution, and in **Adaptive MCMC** the proposal distribution is refined during the mixing process.

4.2.1 MCMC

To benchmark the performance of our improved hybrid MCMC model (ABM-MCMC), we created a Markov Chain Monte Carlo simulation with a standard proposal distribution (MCMC). Our MCMC simulation uses the Metropolis-Hastings algorithm. It is implemented with one of the functions in the MATLAB Statistics toolbox (`mhsample`).

Rather than creating one large monolithic simulation of the entire urban system to explore a variety of scenarios, here MCMC is used to directly forecast specific questions of interest, such as estimating the number of cars entering the parking lots at different times of a day. One can envision this as a two dimensional diagram with the horizontal axis corresponding to the time of a day, and the vertical one showing the number of cars entering a specific parking lot. The survey data from the questions about the attendance pattern and frequency of parking lot usage are used to initialize the MCMC model. Observations for the Bayesian inference process are simply obtained based on the results of the survey data for a simulation period of 90 days. Imagine that based on the survey data a student respondent enters the campus everyday before 9 am, leaves at 5 pm, and reports his general usage of parking lot A as being at a frequency of once a week. In this case, the expectation is that the student would have occupied Lot A twelve times ($90/7$) during the simulation period, so a corresponding number of samples tagged with the reported time range are produced and added to the input observation data.

Table 4.2 shows the parameter settings for MCMC used in the experiments. The burn-in value refers to the number of values that are discarded before the actual samples for the Markov chain are generated. In order to remove the correlation between the nearby samples, sometimes the samples are not gathered sequentially. The number of samples that are thrown away determines the omission rate. Here, two out of three samples are omitted. The last parameter in the table shows the number of Markov chains that are created.

Table 4.2: The parameter settings for MCMC

Parameter	α	Burn-in	Omission Rate	Chains
Value	[1 1 1]	$1e+4$	2 of 3	2

4.2.2 ABM-MCMC

Similar to the notation that is used in [16], we can describe the relationship between the ABM and MCMC models mathematically. The state of each agent i in the agent-based model can be denoted by the vector $x_i \in X_i$ that can assume values in the range $X_i \subseteq \Re^{n_i}$. The state space of the whole population can be designated as: $X = X_1 \times X_2 \times \cdots \times X_n$, and the state of the model at time t as $x(t) \equiv (x_1(t), x_2(t), ..., x_n(t)) \in X, x_i \in X_i$.

After convergence of the MCMC process, the following condition will hold: $x(t) \simeq x(t + 1)$. Here, X can be designed to be same as the variable whose distribution we are seeking using MCMC. The intuition is that the world state, X, assumes the shape of target distribution. By designing an appropriate agent-based model, this variable will be close to the sought-after target distribution.

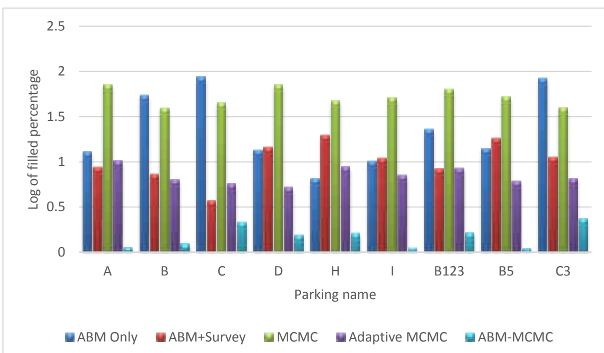
While we have enough samples from a variable x, it is easy to compute its probability distribution function (PDF). In this case, the samples drawn from the agent-based model are used to determine the pdf of the proposal distribution. In our experiments, we assign a probability value to each point x proportional to its number of occurrences in the population domain of the agent-based model: $q(\alpha) = P(x_i = \alpha)$.

In our proposed unification of ABM of MCMC, the input proposal distribution, $q(x)$, for MCMC is derived using the above assumptions. The samples that are produced by the ABM can be used to construct the proposal distribution in the MCMC. It is worth noting that there are other non-mathematical alternatives for combining the two methods. For instance, it would be straightforward to simply directly use MCMC as an embedded component to model regions of the simulation where the total occupancy is of more interest than the exact agent position.

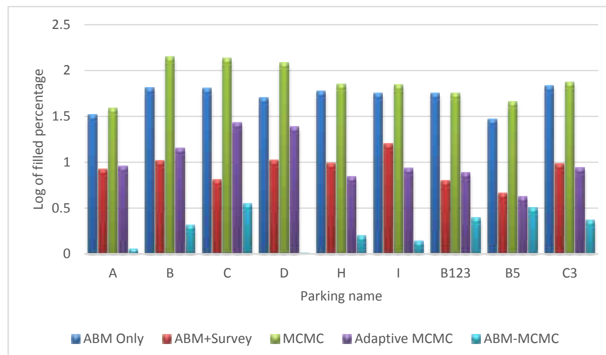
In case of our case study, the final goal of the campus modeling problem was to reach to a model describing the transformation patterns of students. Accordingly, the desired distribution should represent the time and location of students. This information was retrieved from the agent-based model by recording the x and y coordinates of agents at each hour (tick) for 90 days. A Dirichlet distribution, is used as the unnormalized distribution, $\pi(x)$. The general PDF of the Dirichlet distribution can be expressed as:

$$f(x_1, ..., x_{k-1}; \alpha_1, ..., \alpha_k) = \frac{1}{B(\alpha)} \prod_{K}^{i=1} x_i^{\alpha_i - 1}$$

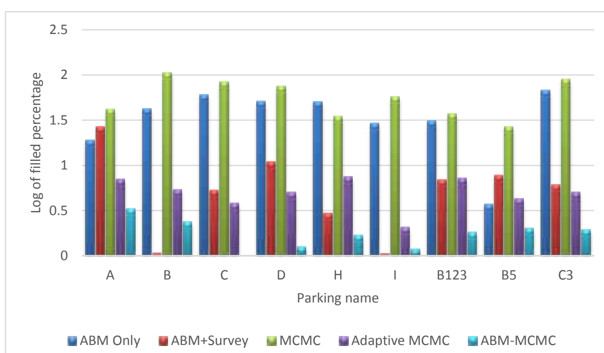
Three variables, x, y and $time$, form the three dimensional support of the applied Dirichlet distribution used by our model. Hence, k in above formula is equal to three, and x_1, x_2 and x_3 correspond to x, y and $time$. The α values are simply assumed to be equal to one. The proposal probability of each vector, containing x, y and $time$ values, is equal to the number of times the vector exists in the dataset divided by the total number of records, under the assumption that the agent-based model has produced evenly distributed samples from the population domain. The `MCmultinomdirichlet` function in R is used to implement the proposed method; this function generates a sample from the posterior distribution of a multinomial likelihood with a Dirichlet prior.



(a) 8 am.



(b) 12 pm.

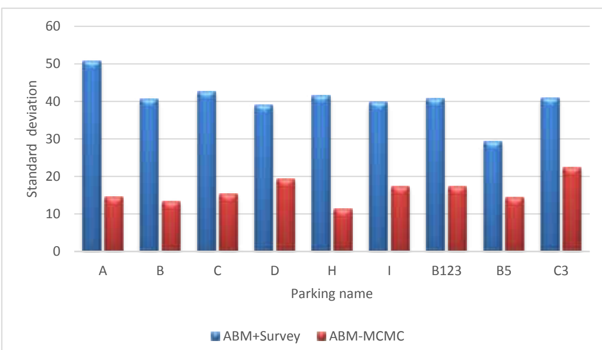


(c) 4 pm.

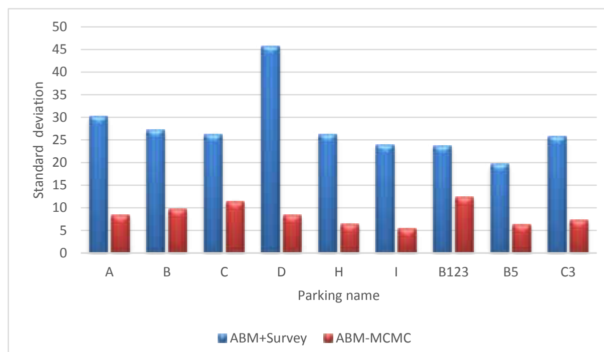
Figure 4.6: The absolute difference (plotted on a log scale) between the average occupancy percentage of the campus parking lots (shown on the horizontal axis) as predicted by different modeling methods compared to the UCF Parking Services data. Our proposed method (ABM-MCMC), shown at the far right, yields consistently better estimates of parking lot utilization with a close to zero absolute difference.

4.2.3 Adaptive MCMC

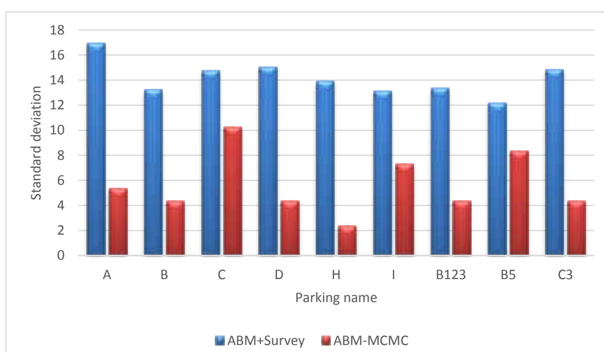
We benchmark our proposed hybrid method against a technique known as adaptive MCMC [60] in which the proposal distribution is updated at run-time based on the behavior of Markov Chain. For this method, the Metropolis-Hastings algorithm from the MCMC toolbox for MATLAB [112] was used. Our MCMC model assumes the unnormalized distribution is of the form of a Poisson distribution, the same as our ABM model. For the proposal distribution, a Gaussian is used. The MCMC attempts to find the most likely value of the the mean of the Poisson distribution (λ in $\frac{\lambda^x e^{-\lambda}}{x!}$).



(a) 8 am.



(b) 12 pm.



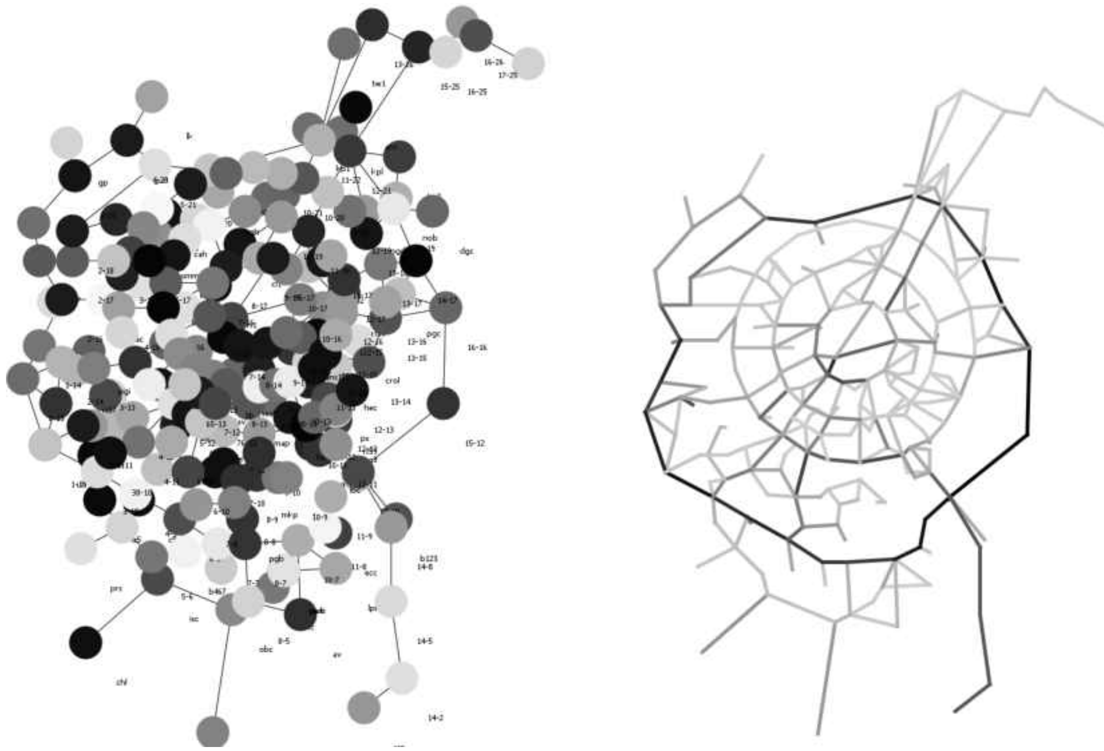
(c) 4 pm.

Figure 4.7: The standard deviation of predicted values for parking usage, obtained by the agent-based model (ABM+Survey) and the proposed hybrid approach (ABM-MCMC) for 20 separate runs. The proposed method yields a reduction in prediction variance.

4.3 Results

To evaluate the performance of the agent-based model under different initialization conditions, we examined the transportation forecasts produced by the simulation, both through visualization and by comparing the predictions against a dataset collected by the UCF Parking Services office. The occupancy percentage of UCF student parking garages (shown on the horizontal axis) predicted by every modeling method is compared. Figures 4.6a to 4.6c show the absolute differences between the forecasts for each modeling technique and the parking service data (closer to zero is better). Note that our hybrid method (shown at the far right) consistently produces the best estimates, improving upon both its parent techniques. The stability of results obtained by different modeling methods is a concern; in many agent-based models, small changes in initial conditions can result in large changes in the final prediction. Figure 4.7 shows the standard deviations obtained from 20 runs of the ABM+Survey and the proposed ABM-MCMC methods. Note that using the MCMC estimator reduces the variance of the raw ABM model, resulting in more consistent predictions.

The agent-based portion of our model can be used to create useful visualizations to provide intuitions about the students' transportation patterns. One of the common questions often asked by policy makers is the density of humans at various locations [59]. Figure 4.8 shows the probability of being in a location on campus for the students at large. In this figure darker circles show more populated areas. In addition to the spots and buildings on campus, the traffic on the streets and walkaways can be also predicted by our method. Some obvious facts that can be easily verified by a domain expert are also observed in this set of results. For instance, as on most university campuses, the student union is the most frequently visited place since it is the venue for most events and many dining locations. The wide drivable boulevard that surrounds the campus dominates the road usage as it is the only way that can be used by cars and shuttles to reach most points on campus.



(a) Student location

(b) Average traffic

Figure 4.8: Visualizations produced by our agent-based modeling system of the probability of a student being at a certain location (left) and the average traffic passing the streets and walkways on the campus (right)

4.4 Conclusion

Hybrid models are a powerful strategy for reconciling the predictions of multiple models to present a unified picture to policy makers, while retaining the diversity and flexibility of multiple approaches. This chapter introduces a new hybrid modeling method for combining agent-based models with MCMC. We demonstrate that the proposed method for initializing the MCMC proposal distribution with ABM data significantly reduces the prediction error over standard MCMC and also improves upon the ABM alone. We hypothesize that the combined ABM-MCMC finds a more general model of the the posterior distribution than the ABM alone. Although agent-based models are often difficult to formally specify and reproduce exactly, the contribution of the ABM can be entirely quantified by the single proposal distribution, which makes it possible to reproduce the results without replicating the entire ABM. In a case study, we demonstrate that our method can be used to accurately model and forecasting transportation patterns in a large urban area.

One simple improvement that we are planning to make in the future is to add faculty/staff into our simulation; this was not a priority initially since previous work has shown that faculty/staff activity profiles have a much lower entropy and are inherently easier to predict than student profiles [70]. Supplementing the simulation with additional information about semester class scheduling is likely to yield the largest forecasting improvement at the cost of making the simulation less applicable to other urban modeling problems. A large amount of class attendance and scheduling information is collected by the university and could be added to the simulation without requiring additional survey efforts.

CHAPTER 5: ANALYZING AGENT-BASED MODELS USING CATEGORY THEORY

In the previous chapter, we introduce ABM-MCMC as a new hybrid method that can achieve a better performance in comparison to the original methods. In this chapter, we use category theory to illustrate why agent-based modeling and Markov Chain Monte Carlo are good candidates for mixing. It is worth noting that to some extent any two (or more) modeling techniques might be mixed in order to create a new hybrid method. The challenging part here is to show why it makes sense to hybridize the methods, and why the hybridized method can be a better replacement for original methods.

The power of category theory comes from its ability of abstracting complex structures. Rather than objects (as usual case in mathematics), in category theory the focus is on relations. Accordingly, most of definitions in category theory are related to various relations that objects or even the relation could have. The idea here also is based on this property of category theory. First, how these two methods can be presented in categorical language will be discussed. Then using one of the structures available in category theory, the relation between these two methods will be shown. At the end of this chapter, the UCF transportation modeling case study plus another marketing case study are used to show the performance of ABM-MCMC.

5.1 Applying Category Theory

To apply category theory, the first thing we need to do is to define the required categories. We define category **A** as representing an arbitrary agent-based model, and category **M** as representing an arbitrary Markov Chain Monte Carlo model. The challenging aspect of using category theory is often showing that the desired structures can be considered a category. In order to show this, we need to show that these structures have all of the properties listed in the definition of a category. We will return to this point later in this section after introducing the elements of categories **A** and **M**.

The approach we are going to use to describe these categories is partly based on the representation approach described in [116]. This method was used to show the agent-based modeling in CT language and is mainly based on ideas from inverse theory, which is the process of finding the best values for the parameters associated with an assumed model based on the observed data [108]. Inverse theory is itself an extensive and thorough theory. Here, we just need a couple of elementary ideas from inverse theory to define the objects in our categories. The purpose of employing ideas from inverse theory is to define formally what is meant by model and data in our representation. The forward problem in inverse theory relates to the problem of predicting data based on the description of the model parameters. Using elements from category theory language, the forward function can be represented by a morphism from object M to object D as follows:

$$F : M \longrightarrow D \tag{5.1}$$

Similarly, the inverse problem can move from data to model, as shown here:

$$F' : D \longrightarrow M \tag{5.2}$$

Additionally, another object namely the universal object, U, can be defined. This object refers to all of the existing information about the system. Some portion of this information is assumed to be known through available data, and the rest will be (partially or totally) produced through the modeling technique (e.g., ABM or MCMC).

The process of moving from model to data or from data to model can be also studied on a Bayesian basis. Hence, two new objects related to the conditional probability of objects M and D can be added to the objects defined so far: $M|D$ and $D|M$. $M|D$ (model given data) refers to the addition of data to the model, or the situation of inferring the model from data, and $D|M$ (data given model) represents the opposite process.

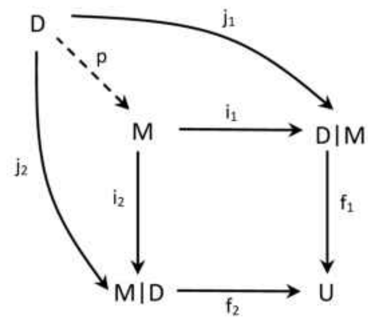


Figure 5.1: The proposed categorical representation for an arbitrary ABM or MCMC model

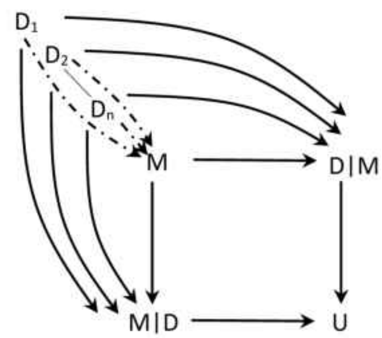


Figure 5.2: There is a unique mapping between the object model, M, and each object corresponding to the different observed datasets, $D_1 \ldots D_n$

The next step is to define the categorical representation of an ABM and MCMC model. These categories are presented using the pullback structure introduced earlier, and are shown in Figure 5.1. In this case, there is a commuting square that according to the definition of pullback should be universal, meaning that it should be the *best* among all similar squares. In CT language, this means *given any other one there should be a unique morphism/factorization to* M. M is the vertex with two projections in the square. Here, *any other one* refers to any other square which also has the two morphisms $M|D \longrightarrow U$ and $D|M \longrightarrow U$. More exactly, this can also be imagined by considering different observations or different sets of data. This is shown by D_1 to D_n in Figure

5.2.

The reason why a pullback was selected to present this structure is the inherent universal property in pullback. If we assume that only one correct model exists–which in theory is a valid assumption regarding a system–then the model exactly plays the role of an universal object in a categorical structure. That is to say no matter which modeling technique we use, as long as the system is the same, there exists a unique model showing the system. In terms of category theory, universal object is the *best* or *most efficient* object, considered as a factor of other objects[1]. Four types of morphisms are presented in this structure. Type f which shows the mapping from the conditional knowledge, $D|M$ and $M|D$, to the universal knowledge, U. This is shown in Figure 5.1 by f_1 and f_2. Type i refers to the model transition to the conditional knowledge. Type j shows the morphisms from data (observation) to the conditional knowledge. Finally, type p shows the unique[2] morphism that should exist between data D and model M. This denotes the probabilistic relation that exists between data and model in any Bayesian domain. In other words, if we look at the ABM and MCMC as both sample generator techniques, the probability that is obtained by the population of samples represents the morphism p defined between data object D and M.

Armed with these definitions, we can verify the compliance of the proposed categorical structures with the formal definition of category in CT. The suggested structure has objects and morphisms defined; the morphisms are associative. For simplicity, identity arrows for the objects are not shown in the figures. Two types of composed relations could be imagined in this structure as shown on either side of the set of equalities in Equation 5.3. In order to have associativity property, the following equalities must hold. Since we have only one model object, these equalities exist among the composite relations. Here, $\circ$ operator shows the composition of two morphisms.

[1]Not to be mistaken with the object U we defined for the categories. Object U, as its definition shows, just represents the universal knowledge about the problem. It is an ordinary object in the category we defined. It does not possess any universal property which is a separate concept.

[2]According to the definition of pullback, there should be a unique morphism from each object D to object M on the corner of square.

$$f_1 \circ (i_1 \circ p) = (f_1 \circ i_1) \circ p$$
$$f_2 \circ (i_2 \circ p) = (f_2 \circ i_2) \circ p \tag{5.3}$$

We defined our categorical structure in such way that it could be used to describe both methods at the same time, so no other category is required. This, by itself, shows the similarities between these two different methods. Now, we can describe the formal relationship between the two categories. The way that the two categories are defined allows us to observe that an adjunction exists between the two categories A and M [25], which can be represented by the same structure shown in Figure 5.1. This way, it is simple to see that there exists a left adjoint functor from category **A** to category **M**, and a right adjoint functor from category **M** to category **A**. This is shown in Figure 5.3.

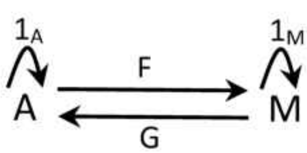


Figure 5.3: The adjunction between A, the category showing an arbitrary ABM, and M, the category showing an arbitrary MCMC model

In order to show this, we need to prove that there exists a pair of functors and a pair of natural transformations between the two categories. The first part is trivial. Since for each object/morphism in category **A**, there exists a corresponding object/morphism in category **M**, a functor from category **A** to category **M** exists. The same justification can be used to show that a functor from category **M** to category **A** exists. For the second part, we can show that two natural transformations, η and ϵ exist. These two are shown in Equation 5.4. Showing the existence of these two is again trivial. Since the functors essentially map the same type of objects and morphisms between two categories A and M, η and ϵ exist between functors F and G.

$$\begin{aligned} \eta :\ & 1_A \Rightarrow GF \\ \epsilon :\ & FG \Rightarrow 1_M \end{aligned} \tag{5.4}$$

One question that might come to mind is that among the four types of relations introduced in Table 2.1, why adjunction is chosen to show the relation between categories A and M? To answer this, we should note that in all of the other three relations some sort of ‘being the same’ exists by definition, but in adjunction, we generally do not care about being the same. Instead, we focus on the interesting relations between the two categories. Additionally, choosing adjunction for our purpose does not prevent the usage of other relations, and does not state that the others cannot exist at all. What is important is that it enables us to reach to our goal, which was to formally represent the relation among the ABM and MCMC models.

5.2 Insights from Category Theory

Thus far, we have proven that the two methods can be shown to be equal (up to natural transformation) in terms of category theory, i.e. the weakest equality. Hence it is possible to combine the two methods to produce a hybrid modeling methodology that builds on the strengths of both models. Theoretically any two or more methods that function in a shared domain and can be used in sequential manner could be considered as candidates for building a hybrid method. But, the difficult part is to show why this hybrid method is valuable. Using category language, we can evaluate the feasibility of different models for hybridization.

Our hybrid model (ABM-MCMC) uses the population of samples generated by the agent-based model to initialize the proposal distribution for the MCMC estimator [26]. In the categorical representation of ABM, prior knowledge (data) shown by D determines how the agent-model should be constructed. Then, through an iterative process the data determines the model by producing samples while agents remain active in the system. The Bayesian approach of representing this process helps us to understand the similarity between ABM and MCMC better. The samples produced by the agent-based model become the data within the MCMC component. This hybrid method is able to resolve the proposal distribution problem of MCMC methods, while possessing greater verification possibilities than the ABM alone.

5.3 Results

In order to validate the performance of our hybrid method, two case studies are presented in this section. The first case study is related to an urban modeling problem, which was presented in previous chapter. Here, we report a similar set of results with slightly different settings. The second describes a marketing scenario. In both application domains, we show that the hybrid method (ABM-MCMC) outperforms either ABM or MCMC alone. Both agent-based models presented here are implemented in Netlogo [168], and the MCMC component of the hybrid method is run using the `MCMCpack` package in R [160]. Implementation details about these case studies are omitted from this section since they are irrelevant to the main thrust of this chapter and can be found in the original descriptions of the ABMs.

5.3.1 Urban Transportation Simulation

This case study extends our earlier work described in depth in Chapter 4. The aim of this project was to model the transportation patterns of students at the University of Central Florida (UCF). The data for this study was collected through an online survey. A detailed agent-based model was created based on the survey data of students' housing, dining, and transportation preferences. The simulation can be used to perform analyses of traffic patterns, building occupancy and parking usage. Here, we specifically present the results related to student parking usage for comparing the different models since it was possible to obtain ground truth data for this quantity from UCF Parking Services.

For ABM-MCMC part, the data samples generated by the agent-based model showing the location of each agent were used to initialize the proposal distribution. Additionally, in order to test the MCMC method alone, the MCMC toolbox for MATLAB [112] was used in combination with the original survey data. In this toolbox, the prior distribution is simply assumed to be in form of Gaussian distribution. Figure 5.4 shows the results obtained using each of these three modeling approaches.

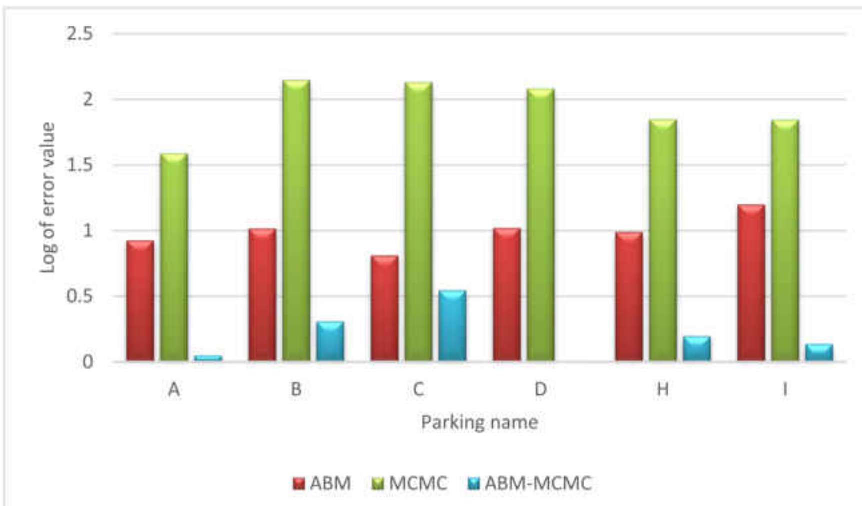


Figure 5.4: The log of the difference between the number of cars predicted by each method and the numbers from empirical data. Shorter bars show a smaller deviation between the model prediction and the actual data. Our hybrid method, ABM-MCMC, outperforms the parent methods in all cases.

As the figure shows, the hybrid method outperforms the original methods in terms of accuracy of prediction.

Table 5.1: Accuracy of ABM-MCMC in comparison with ABM and original Bass model

Product	Period of forecast	ABM-MCMC R^2	ABM R^2	Bass R^2
AC	1950-61	0.86	0.72	0.90
Bed	1950-61	0.91	0.93	0.93
Coffee	1951-61	0.77	0.74	0.69
Dryer	1950-61	0.86	0.85	0.85
Freezer	1947-61	0.64	0.60	0.47
Lawnmower	1949-61	0.93	0.93	0.89
Refrigerator	1926-40	0.63	0.61	0.76
TV	1949-61	0.13	0.19	0.07

5.3.2 *Marketing Analysis*

The second case study is based on an agent based model published by Rand et al. [144] in which the agents are used to model consumer behavior. The main purpose of this work was to simulate the famous Bass model [21] published in 1969. The authors also study the role of different network structures on the market's behavior. The code and detailed documentation of the agent based model are freely available online. The Bass model describes how a population of consumers adapt to new products. This is done by defining two type of consumers: innovators and imitators. The behavior of the model is determined by three parameters: degree of innovation (p), degree of imitation (q) and market size (m). The same parameters are used for both the Bass model and the agent-based one. The R^2 correlation between the empirical sale data (showing the number of units sold each year) and results from the original Bass model, Rand's paper, and our hybrid method are presented in Table 5.1. The correlation value is computed using the *RSQ* function in Microsoft Excel. In order to have a larger set of samples to feed the MCMC method, we ran the agent-based model 50 times.

The hybrid method shows a slight improvement in prediction accuracy. The difference between correlation values is not significant, due partially to the fact that the amount of available empirical data in the Bass original paper is fairly small.

5.4 Conclusion

In this chapter, we illustrate how category theory can be used to formally represent two popular modeling techniques, agent-based models and Markov Chain Monte Carlo simulation. Abstractions from category theory can be used to relate the different models using adjunction and form the basis for our proposed hybrid implementation of the parent models (ABM-MCMC). To demonstrate the benefits of our hybrid model, we present two case studies, urban transportation and consumer modeling, where ABM-MCMC outperforms the original modeling methodologies.

CHAPTER 6: A NORMATIVE AGENT-BASED MODEL FOR PREDICTING SMOKING CESSATION TRENDS

In this chapter, a normative model for studying smoking patterns is presented. The main contribution of this part of the dissertation is to propose a normative architecture for a real-world simulation problem which is complex by nature. In order to do this, it's crucial to model the factors that affect smoking behaviors of humans in the society of interest. These factors are categorized and modeled in three main categories: personal, social and environmental. The same smoking model will be used in the next chapter for evaluating our second normative architecture.

This normative architecture is then applied to the model of students at the University of Central Florida which was fully described in Chapters 4 and 5. One of the main advantages of having such a model of transportation patterns is the possibility of modeling social and environmental factors in a realistic way. For instance, some social relations are directly related to being in the same location, or getting in touch physically.

6.1 Norms and Smoking Modeling

In addition to the abstract usage of norms in normative multi-agent systems which was introduced in Section 3.4, the role of norms in social simulations has also been widely researched. Social control, benevolence, reciprocity, and institutions [98] are among common topics that are studied using norms.

Outside of computer science, the social norm marketing approach has become an important tool for public health messaging [1]. There the emphasis is on changing human social norms, rather than computationally modeling them. These types of methods have been very successful at curbing college drinking and substance abuse [139]. This indicates that our proposed approach of building normative effects into our model should be highly effective, given the previously demonstrated relevance of norms to human smoking behavior.

Non-normative models of smoking behavior already exist; for instance, *SimSmoke* is one of the widely used tobacco control policy simulations. It models the dynamics of smoking use and smoking-attributed deaths in the society of interest, as well as the effects of policies on those outcomes [113]. Other types of simulations have been used to model the consequences of second-hand smoking [58]. In addition to norms, our proposed approach also simulates network effects as was done in Beckman et al.'s study on the propagation of adolescent smoking behavior [22].

Most existing models within the medical and public health community are based on a statistical analysis of smoking data [121]. This set of methods are often specific to a certain aspect of the problem such as modeling abstinence based on changes in brain cells. Moreover, some models based on system dynamics approaches have been used in the public health domain [161]. An introduction to this set of techniques can be seen in [99].

The relationship between social norms and smoking behavior was examined as part of a European Union study on the impact of cultural differences on the emergence of norms in different countries after the commencement of anti-smoking legislation [63]. Our current ABM does not attempt to recreate cultural effects. Rather than studying smoking cessation behavior at the macroscopic level, we adopt a higher fidelity approach in which the daily behavior patterns of individual agents are simulated within an activity-oriented microsimulation.

6.2 Normative Model

To construct a normative model for a real-world scenario, we need to define both a normative architecture and the components that are used to recreate the real-world problem. The components for the smoking scenario will be introduced in the next section. In this section, we introduce our Lightweight Normative Architecture (LNA), and the next chapter describes the Cognitive Social Learners (CSL) architecture.

Our architecture encapsulates some of the functionality of earlier normative architectures while remaining simple and lightweight. One oft-cited work in this area, the BOID architecture, extends the classic BDI approach to include a fourth element—the notion of obligation [35]. The idea of obligation was introduced into the architecture to support social commitments, such as norms. Norms can be viewed as following a three stage lifecycle, including formation, propagation, and emergence [148]. Adding norm emergence provides scalability and flexibility to normative environments. The EMIL framework [119] was introduced after the BOID architecture and represents the culmination of extensive research on norm emergence. Similar to BDI, EMIL uses belief, goal, intention and action as the procedure for norm emergence. Using the EMIL architecture in real scenarios can be challenging due to the elaborate design of its cognitive mechanisms, so we propose the following simplified architecture for how norms affect smoking behavior.

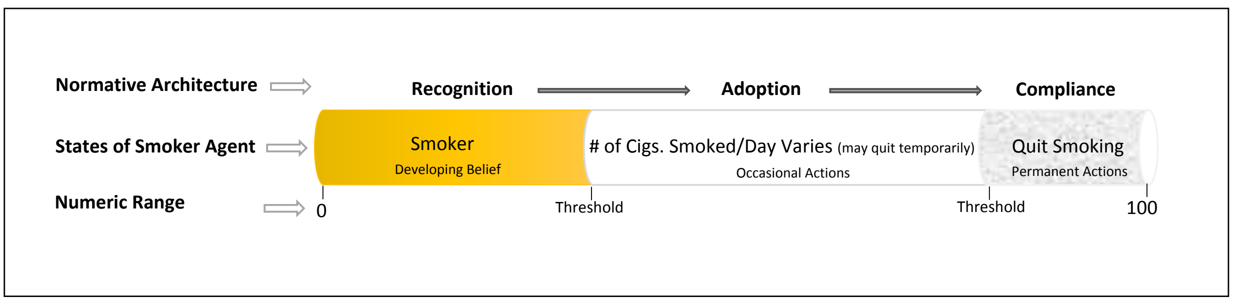


Figure 6.1: A schematic representation of our proposed architecture. The top row shows the three stages of the normative architecture. The middle row presents the observations corresponding to the stages within the context of the smoking scenario. The smoking norm life cycle is governed by a parameter (*smoking-value*) ranging from 0 to 100. The two user-defined thresholds (bottom row) determine 1) when an agent enters each stage and 2) what transpires.

Each agent has a personal *smoking-value* ranging from 0 to 100 that governs its behavior. As shown in Figure 6.1, our architecture contains three stages: **recognition**, **adoption** and **compliance**. In the first stage (recognition), the beliefs of an agent change and develop. During the adoption phase, the agent commences action. Note that the general definition of adoption in normative systems is very consistent with our smoking scenario. As described in the literature, during the adoption phase the agent can opt to violate the norm. The equivalent violation in the smoking scenario (recidivism) is quite common in those trying to quit. In order to quit smoking, a smoker usually decreases the number of smoked cigarettes, which can be considered as another adoption behavior. The compliance phase is used to simulate the situation when the agent really starts quitting. These three phases also map well to the stages that are usually considered in smoking studies: initiation, maintenance and abstinence. The next sections describe the factors considered by our model.

6.3 Smoking Model

After introducing our normative architectures, in this section, we present the smoking model that we specifically designed for simulating smoking behaviors of people. Our model considers three sets of factors that are known to affect human smokers: personal, social, and environmental influences. Considering the complex and challenging nature of modeling smoking behaviors, especially the addictive property of smoking, we tried to have an inclusive model that contains as many factors as possible.

6.3.1 Personal

Our model includes a set of personal values which are specific to each person, and depend on their personality; Dechesne et al. use a similar set of values within their model of cultural differences that affect smoking behavior [63]. According to the sociological theory of cultural value orientation introduced by Schwartz [153], three types of values determine cultural differences in societies. These values are defined by three bipolar cultural dimensions that can be used to describe possible resolutions to problems confronting societies. In our model, we adopted two of these values since the third dimension is specifically for cultural differences which are negligible for our relatively homogeneous undergrad population. The two adopted values are described below:

- **Embeddedness vs. autonomy**: This determines how much an individual's preferences, feelings, and ideas are affected by others through various relationships vs. being cultivated internally.

- **Mastery vs. harmony**: This refers to the dichotomy of being ambitious, daring, and self-assertive vs. being consistent, understanding, and appreciative of the environment.

The first item is referred as **individualism** (ind), and the second one as **achievement** (ach). The third item which is not included in our model is equality. In addition to these two personal values drawn from Schwartz's sociological (or anthropological) model, three other personal values are included:

- **Regret** (rgt) - In our scenario, this value shows how much the individual is regretful about smoking and is used to model the phenomenon of addiction. The role of regret in smoking behaviors is described in [51]; it is related to their willingness to quit smoking or decrease their tobacco usage.

- **Health** (hlt) - As the name implies, this value shows the extent to which a person cares about her health, and also pays attention to medical recommendations.

- **Hedonism** (hdn) - The pleasure-seeking aspect of one's personality. Health and hedonism were also used in the EU smoking model [63].

6.3.2 Social

The second aspect of our model is used to quantify the effects of the community on the individual. To do this, we create a synthetic friendship network for our simulated community using the method described in [165] for creating human networks that follow a power law degree distribution and possess homophily, a greater number of link connections between similar nodes.[1] The

[1] Commonly described as "birds of a feather flock together" [126]

network generator uses link density (*ld*) and homophily (*dh*) to govern network formation. A simplified version of the pseudo-code for this method is shown in Figure 7.2. For our smoking model, three elements are defined to determine the homophily of a node: age, gender and undergraduate major. The nodes of the graph represent the individuals (agents) in the simulation.

```
G = Null
repeat
    sample r from uniform distribution U(0,1)
    if r ≤ ld then

        randomChooseSource(G)
        determineCandidateSink(dh,G)
        pickSink()                              ▷ based on power-law distribution
        connect(source,sink)
    else
        add a new node to G
    end if
until desired number of nodes added to the network
```

Figure 6.2: Synthetic friendship network generator

In order to implement the diffusion of smoking behaviors in the friendship network, a game-theoretic approach [71] is used. Here, a simple two by two matrix is defined that contains four different states that can occur in the smoking scenario. Table 6.1 shows this matrix. The descriptions below the table show how the payoffs are calculated. The abbreviations on the right side of the equations relate to being a smoker (s) or non-smoker (n).

Each individual is either a smoker or non-smoker. The payoff for each of four entries of a node is calculated according to three factors: personal values, network neighbors, and whether the subsequent state is similar to the current state. In order to show the tendency of people to maintain their current state, α and β values are added to the model. These two parameters are constant positive values which make the value of the payoff higher for the cases that the agent remains a smoker or non-smoker than in the cases that a state transition occurs. The final value for the friendship element of model (frd) is calculated based on the current state of the individual and her friends, using the payoff matrix.

Table 6.1: Payoff matrix governing the diffusion process in the friendship network. Prime (′) means complement, which in this case is equal to: “100 -”. ind: individualism; ach: achievement; hlt: health; hdn: hedonism

		Node B	
		Smoker	Non-smoker
Node A	Smoker	$ss{+}\alpha$ $ss{+}\alpha$	sn ns
	Non-smoker	ns sn	$nn{+}\beta$ $nn{+}\beta$

$ss = ind' + ach' + hlt' + hdn$
$sn = ind + ach + hlt + hdn'$
$ns = ind + ach + hlt' + hdn$
$nn = ind' + ach' + hlt + hdn'$

6.3.3 Environmental

The third category of factors that affect people’s smoking behavior is what they observe or encounter in their surroundings. Four items are considered in this category: others, signs+butts, advertisements, miscellaneous.

Others (oth) - One major factor that affects norm compliance is observing other people’s behavior. Seeing other smokers can affect the agents’ decisions to obey policies, particularly when complying with smoking cessation rules. Similar behaviors in humans have been shown to exist and are usually referred to as *observational learning*. Various studies have shown the effect of observation on smoking behaviors (e.g., [2]).

Signs + butts (sbt) - This item is specifically related to the effect of installed *No Smoking* signs, that advise people to refrain from smoking. A key research challenge here is to simulate the behavior of people in response to this type of notification. A recent study by Schultz et al. [152]

on littering in public locations shows that people tend to obey installed signs when there is no trash around the sign, but when litter exists in the vicinity, the rate of people who do not follow the signs increases significantly. Using a similar approach, we consider signs and cigarette butts together and model the influence of observed cigarette butts on a person's on-campus smoking behavior.

Advertisements (adv) - Physical advertisements can also influence smoking behaviors. These advertisements are a major part of the campus smoke-free program. This category refers to tents, fliers, billboards, catalogs, posters and banners installed permanently in different locations of campus.

Miscellaneous (msc) - This category encompasses all of the other factors that might influence a smoker's decisions. One major aspect of this category is non-physical influences, especially digital, educational, and promotional activities. Also included in this category is the role of different cessation facilities available on campus, such as workshops and nicotine replacement therapy (NRT).

Each of these four elements is represented in the model with values ranging from 0 to 100. A simplified version of Q-learning is used to govern the effects of the environmental factors. As Table 6.2 shows, when encountering an environmental factor such as a banner, the state of an agent is defined by the current value of its personal and social elements. The agent can either be affected by the environmental factor or disregard it. In case of the first action, the value of that environmental factor will increase by a fixed amount, but in the second case nothing changes. The reward that agent receives from each action is calculated based on three elements of its personal value vector: regret, health and hedonism. The reward value falls between -1 and +1, and is calculated using the following formula:

$$\text{reward} = (\text{regret} + \text{health} - 2 * \text{hedonism})/200 \tag{6.1}$$

A dynamic learning schedule is utilized for the Q-learning, which results in a higher rate of learning at the beginning of the simulation, and a lower one afterwards.

Table 6.2: Q-learning definitions for state, actions, and rewards. If the agent does not pay attention, it means that the agent opts to ignore a specific environmental element. Regret and health affects the reward value positively, and hedonism affects it negatively.

States	current value of personal and social elements
Actions	pay attention or not
Rewards	calculated based on the values of regret, health and hedonism

The five elements introduced for the personal values, the social element, and the four environmental factor are all defined as ranging from 0 to 100. The main smoking-value (SV) is calculated using this formula:

$$
\begin{aligned}
\mathrm{SV} = (&k_1 * \mathrm{ind}' + k_2 * \mathrm{ach}' + k_3 * \mathrm{rgt} + k_4 * \mathrm{hlt}' + k_5 * \mathrm{hdn} \\
&+ k_6 * \mathrm{frd} + k_7 * \mathrm{oth} + k_8 * \mathrm{sbt} + k_9 * \mathrm{adv} + k_{10} * \mathrm{msc}) \,/\, \Sigma_{i=1}^{10} k_i
\end{aligned} \tag{6.2}
$$

The smoking-value (SV) falls between 0 to 100. In this formula, k_1 to k_{10} show nine coefficients that are assigned to the user. Prime ($'$) means complement, which in this case is equal to: "100 -". The friendship value (frd) is determined using the social model.

6.4 Agent-based Model

The original version of the agent-based model (ABM) used in this work was built to study the transportation patterns of people and vehicles and was described in depth in Chapter 4. To implement the smoking simulation scenario, the proposed smoking model was added to the original ABM. Personal values were added to the set of parameters possessed by each agent in the ABM. These values are calculated using distributions fitted to the available survey data (described in the next section). We added two parameters, age and gender, to each agent's parameter set to be used for measuring homophily in the social model. (The third one, field of study, was available in the original version.) Each agent is initialized as a smoker or non-smoker at the start of the ABM, based on the number of smokers in the survey data. The smoke-free campus policy is assumed to be in effect immediately after the start of the simulation.

Figure 6.3: Screenshot of the agent-based model. The advertisements (orange pentagons) and no-smoking signs (red triangles) are shown on the map.

6.4.1 Data

Our agent based model uses data from three surveys of UCF students. In Spring 2012, we did an online survey of 1003 students to collect the data used to model campus transportation patterns. The other two surveys were conducted by Health Services; one of them was done in Fall 2011, before the smoke-free policy was instituted, and the second in Fall 2012, at the end of the first year of the smoke-free campus. Both of these surveys were performed as part of the annual university ACHA-NCHA reporting process. The student answers to five questions in the first survey were used to determine the numerical values for the five personal values introduced in Section 6.3.1. The personal values and corresponding survey questions are:

- **Individualism** - Do you think breathing smoke-free air on campus is a right?
- **Hedonism** - Do you think smokers have the right to smoke on campus?
- **Achievement** - Would you feel comfortable asking someone to put out their cigarette?
- **Health** - Would a smoke-free campus policy make campus healthier?
- **Regret** - If you smoke, are you interested in attending a smoking cessation program?

Having a detailed transportation model facilitates implementing the environmental aspects of the proposed smoking model in high fidelity. The assumption is that each smoker agent smokes an average of 15 (for men) and 10 (for women) cigarettes per day. These numbers are based on the reported statistics in [38]. The effect of observing others smoking on campus is incrementally aggregated for each agent through the described reinforcement learning algorithm. The observation occurs whenever an agent is close to an agent that is smoking at the same time.

The exact location of no-smoking signs and physical advertisements are defined in the campus map used in the ABM. Based on our observational study of the campus, cigarette butt locations are marked near the large college buildings, but not general buildings like the student union and library. This trend might occur because of the frequent cleaning of these areas, or the tendency of people to avoid smoking in heavily crowded areas. While the agent moves around campus, it passes physical advertisements. Similar to observing others smoking, every encounter with an advertisement increases its effectiveness.

Figure 6.3 shows the user interface of the agent-based model. In this figure, the location of buildings, routes and also the advertisements can be seen. The last item of the environmental model (misc factors) is implemented by a random value that represents the aggregation of all other factors.

The questionnaire was designed using a Likert scale. The personal values in our work were matched to questions after the survey was conducted, and normal distributions fitted to the data were used to initialize the agents' personal values in the ABM. The university administration used the answers to the following three questions to determine the success of the smoke-free campus policy. In our work, the answers to the first and last question were used to show the accuracy of the proposed model. These three questions are:

- Do you support the campus smoke-free policy?
- Do you smoke?
- Are you likely to take smoking cessation classes?

The other data used to implement the model, including the location of advertisements and installed no-smoking signs, was obtained from campus sources.

6.5 Results

Validation is a major challenge while evaluating ABMs—how to show that the model matches reality. One approach is to evaluate the model by comparing the statistics obtained from the model with other sources of data as indicators of ground truth. Here, the data obtained from the second and third questions of the survey described in the previous section is used to evaluate the model. These two questions show the percentage of smokers among the students, and also the percentage of those who are willing to attend smoke cessation workshops.

The ABM is initialized with the same number of smokers and people willing to participate in smoking cessation classes as indicated in the survey data[2]. According to the definition

[2]Since the total number of students is known, the percentage values also determine the numbers, hence we use the terms interchangeably.

presented in Section 6.2, a smoker is an agent whose smoking-value, (sv), is below the quitting threshold. Similarly, we use the middle part of the proposed smoking-value range to identify an agent who is willing to attend smoking classes. An agent who is willing to participate in classes has a smoking-value between the two proposed thresholds. The assumption is that the adoption phase in the proposed architecture shows the situation where the agent has not reached the compliance phase. So, assuming that an agent in the compliance mode is willing to attend smoking classes is consistent with the proposed architecture, because attending class is not a clear quitting task, but is a behavior toward quitting (the action phase).

Table 6.3 shows the parameters that are used in the experiments to determine the smoking range. As the table shows, the value 50 is used for the first threshold and 90 for the second threshold shown in Figure 6.1. In our experiments, the values for the coefficients k_3, k_4 and k_5 in equation 6.2 were 3, 3 and 2. The other coefficients were equal to 1. For the network generation part, the values for the link density, ld, and homophily, dh, were 0.40 and 0.66.

Table 6.3: Experimental settings for smoking-value (sv)

Agent State	Range
Non-smoker	90–100
Willing to participate in classes	50–90

Using these assumptions, we ran our agent-based simulation for a period of a year from Fall 2011 to Fall 2012. In these experiments, we initialized the simulation with the same number of smokers and students willing to go to the classes as the initial survey data, and then compare the numbers obtained from the simulation with the final survey data. During this period, the agents commute to campus and follow schedules governed by the transportation model. The proposed smoking model simulates the smoking behavior of students during the year of study. The average results of ten runs of the model are reported in Figure 6.4. The figure also shows the corresponding statistics obtained from the conducted surveys. The two measures shown here are the percentage of smoker students and the percentage of smoker students who are willing to attend smoking cessation classes. As the figure shows, the model's results are very close to the reported statistics.

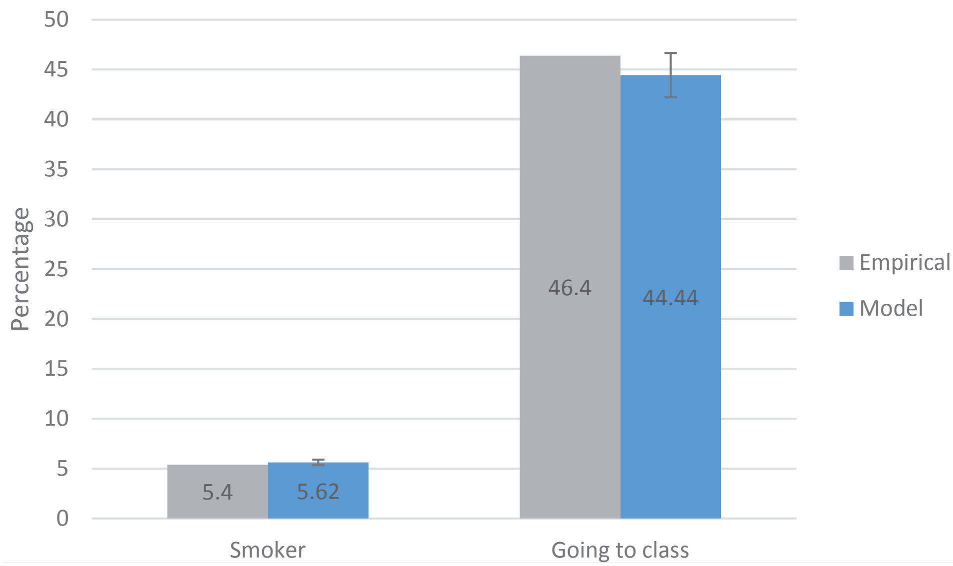
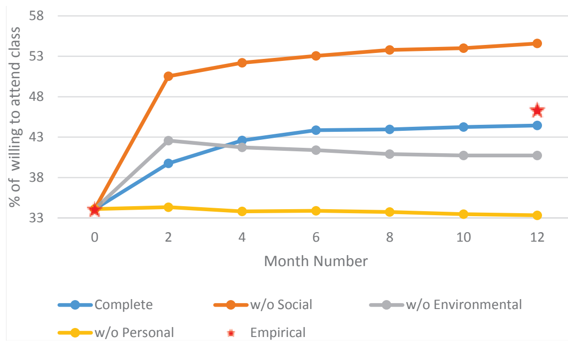
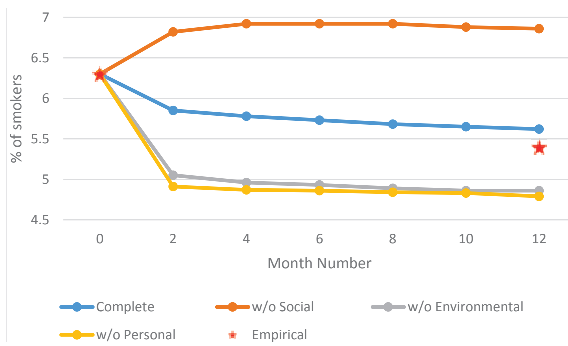


Figure 6.4: **Left:** the percentage of smokers in Fall 2012. **Right:** the percentage of students willing to participate in smoke cessation classes. The grey columns show the reported percentages based on the survey data, and the blue ones show the percentages predicted by our model.

After evaluating the complete model, we also study ablated versions of the model that lack one of the three elements (social, environmental, or personal). The results for alternate months during the year of simulation are reported in Figure 6.5. The reported results are, again, averaged over ten runs, and in all cases the initialization configuration is based on the survey data. In Figures 6.5a and 6.5b the left red star shows the starting value which is the empirically measured value, and is the same for all four experiments. Without the personal and environmental components, the model tends to underestimate results in comparison to the final empirical results. Without the social part, the model overestimates smoking behavior. Based on the size of differences between the empirical results and the other experiments, it can be concluded that the personal value is the major predictor in determining smoking behaviors. Environmental factors had the lowest impact on predicting smoking behavior.



(a)



(b)

Figure 6.5: The percentage of smoker students (a), and those who are willing to attend smoking classes (b) during the one year simulation period. The numbers from the survey data are marked by the red star icons at the beginning and end of the simulation period. The figure shows the predictions of the proposed model (complete), the model without the personal values, without the social aspect, and without environmental influence. There is a close match between the predicted values of the complete model and the survey data.

6.6 Conclusion

Despite the fact that normative agent architectures have improved significantly during recent years, implementation of normative models for large, complex real-world problems has been lacking. Most existing theories and architectures have been evaluated either on artificial scenarios or on small real-world problems. In this chapter, we presented our lightweight normative architecture (LNA) that can be initialized using survey data to model real-world scenarios and demonstrate its usage in modeling the impact of smoking cessation policies on a large university campus. We believe that our model could also be utilized (with some modifications) for similar public-policy problems in human societies.

UCF Health Services plans to promote the importance of encouraging other community members to refrain from smoking on campus. One of the measures used by the university policy makers to demonstrate the success of the smoke-free campus program was demonstrating increases in the percentage of people who feel comfortable enough to ask others to extinguish their cigarettes. Another aim is to increase the awareness of non-smoker students about the harmful effects of second-hand smoking.

CHAPTER 7: MODELING NORM EMERGENCE WITH THE COGNITIVE SOCIAL LEARNER ARCHITECTURE

Our Lightweight Normative Architecture (LNA), which was presented in detail in last chapter, models the impact of personal, social, and environmental factors on recognition, adoption, and compliance with campus smoking norms. When initialized with student survey data, it accurately predicts trends in smoking reduction over a one year timeframe.

One weakness with LNA is that it has a relatively simple internal model of the human decision-making process. To address this issue, we created a general normative architecture, Cognitive Social Learners (CSL) [23], that is capable of reasoning about any social norm. CSL provides a computational mechanism for transitioning behaviors learned during repeated social interactions into the agent's internal cognitive model of preexisting beliefs, desires, and intentions. By incorporating a more complex normative reasoning model, CSL can not only predict smoking trends but also accurately forecasts population-level perception on the social acceptability of smoking.

The first steps toward a new normative architecture that can be used for simulating real-world normative behaviors in human societies are presented in this chapter. Based on what was discussed in Section 3.5, two lines of research can be observed in the literature of normative architectures. While the focus is mainly on cognitive aspects of norm formation of agents in the first group, the other group focuses on the social and environmental aspects of agent relations. The proposed architecture in this chapter, CSL, tries to include the insights from both of these groups, and build a unique architecture.

Human behaviors such as jaywalking and littering are known to be contagious, yet are more complex than the contagious spread of yawning or coughing behaviors that are related to the human motor system. Based on human subjects studies, Schultz et al. (2013) note that the presence of litter positively predicts future littering behavior; unsurprisingly, the availability of trash receptacles is negatively correlated with littering. We selected littering for our study as a good example of an emergent human behavior arising from a combination of social norms, environmental factors, habit, and personality differences. Savarimuthu et al. (2009) also used a littering scenario to demonstrate the operation of their normative multi-agent system.

This architecture is examined on an abstract case study, and the obtained results are compared with the results from two methods belonging to the two introduced groups of normative architectures. At the end of this chapter, the plans for extensions to this work and future work are discussed.

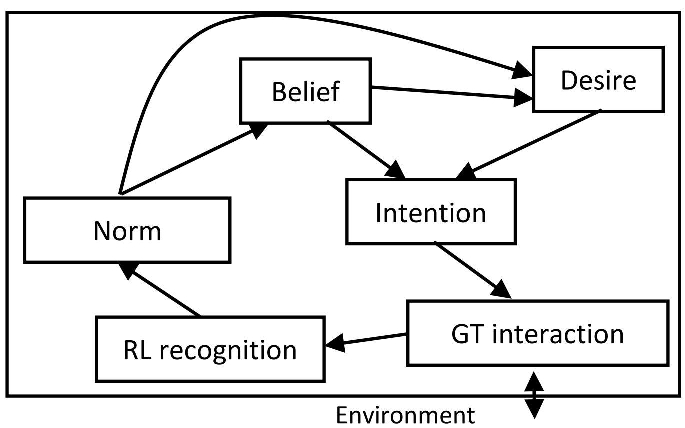


Figure 7.1: Cognitive Social Learners (CSL) Architecture

7.1 Cognitive Social Learner

This chapter introduces a new architecture, Cognitive Social Learners (CSL), that includes components from the two categories of normative architectures, and presents a cohesive model for modeling the emergence of norms. Figure 7.1 shows a schematic view of CSL. In this architecture, the belief, desire and intention components implement the cognitive aspects of norm formation, while the game theoretic (GT) interaction and reinforcement learning (RL) recognition parts implement the social aspects.

We will use a littering scenario as an explanatory example, to describe the proposed architecture's elements. Later, in the experiments section, this scenario is used to evaluate the performance of the CSL architecture at modeling norm emergence. The example scenario relates to people who visit a park. They have five possible actions: littering, recycling, violating park rules regarding animal feeding, violating park rules by trespassing on the foliage, and performing no action.

The representation used for the BDI components and the norms is based on a simplified version of the framework introduced by Casali et al. [42] and Criado et al. [55] in which a certainty degree is assigned to each representation. For example, $(D^{-}\text{payfine}, 0.45)$ designates a negative desire toward paying a fine with a certainty degree of 0.45.

7.1.1 Belief, Desire, and Intention

The CSL architecture follows a classic BDI structure. Like many normative architectures, each agent is initialized with a set of personal values that model innate preferences. In CSL, these personal values are used to create type 1 beliefs that have a certainty equal to 1; for instance $(B[happiness = 50], 1)$ indicates that the personal value of the agent regarding happiness is equal to 50. The other type of beliefs (type 2) model the agent's actions, represented as $(B[\alpha]\varphi, \delta)$. $(B[littering]botherRest, 0.30)$ indicates that the agent believes, with certainty of 0.30, that littering would bother the other agents.

Desires can be determined independently or based on the agent's beliefs. Desires are represented as $(D^{*}\varphi, \delta)$, which models the positive or negative ($* = \{-, +\}$) desire of an agent regarding state φ with certainty of δ. An agent may update its desires when its beliefs changes. This process is shown in Equation 7.1; the certainty value of desire D is updated based on function

f, which is is a user-defined function.

$$((D^{*}\varphi, \delta_{\varphi}), (B[\alpha]\varphi, \delta_{\phi})) \Rightarrow (D^{*}\varphi, f(\delta_{\varphi}, \delta_{\phi})) \tag{7.1}$$

Intentions are derived from the set of positive desires, if they have a certainty value higher than sum of the certainty values of all negative desires relevant to the intention. Equation 7.2 shows this:

$$\begin{aligned} &((D^{+}\varphi_{i_1}, \delta_{\varphi_{i_1}}), ..., (D^{+}\varphi_{i_n}, \delta_{\varphi_{i_n}}), (plan_j, \delta_j)) \\ &\qquad \Rightarrow (I_k, f(\delta_{i_1}...\delta_{i_n}, \delta_j)) \end{aligned} \tag{7.2}$$

while $\Sigma(\delta_{i_1}...\delta_{i_n}) \geq \Sigma(\delta_{l_1}...\delta_{l_n})$ and $_{l_1}$ to $_{l_n}$ are indices of negative desires toward effects of I_k. According to this formula, the set of positive desires (from i_1 to i_n) and plan j will determine the intention k based on a user defined function f. In the littering case, an agent might have positive desires toward higher happiness and spending less effort, but negative desires toward paying a fine and being observed by others. In this case, if the sum of certainty values for happiness and spending effort is more than the sum of certainty values for paying the fine and being observed (assuming that littering is part of the agent's current plan), it will litter.

7.1.2 Game-theoretic Interaction

Instead of deciding its actions based on intentions alone, which is often the case in BDI-based methods, the agent's final action is determined after playing a social dilemma game with one of its neighbor agents. The maximum certainty value of available intentions is used to create a two-by-two matrix. The two possible actions are performing or not performing that action. After calculating the payoff value for an action based on the related intentions, fixed values of α and β

are used to increase the value of the elements in the matrices representing coordinated action (the agent and its neighbor selecting the same actions) [71]. Example of this matrix for the littering scenario are shown in Table 7.1.

Table 7.1: Example payoff matrices for the littering (L=litter, NL=not litter). ι shows the computed payoff value for littering. ι' is the payoff for not littering.

	L	NL
L	$\iota+\alpha$	ι
NL	ι'	$\iota'+\beta$

Based on the outcome of played games, an agent decides what action to perform. What an agent observes after performing an action may cause an agent to update its personal values (type 1 beliefs) and learned norms, which in turn modifies its behavior in subsequent steps. For instance, in the case of our example scenario, after littering, an agent's happiness value will increase; or if there is a punisher in its vicinity, its paid-fine value will increase.

7.1.3 Norm Recognition using RL

The goal of this component is to construct a practical way of recognizing/learning norms, while connecting different components of the architecture. Our RL based recognition component plays the role of a hub among norms and personal values (beliefs) on one hand and the game theoretic interaction on the other hand.

The combination of GT interaction and RL based recognition components is used to implement the social learning process which propagates norms across the agent population. The aim of the social learning framework is different from similar processes in the domain of multi-agent reinforcement learning, in which agents play iterative games to learn a policy resulting in a competitive or cooperative equilibrium. Sen and Airiau [155] note several differences between social learning and multi-agent RL, including the lack of equilibrium guarantees. At every timestep, each agent interacts with a single changing agent, selected at random, from the population. The payoff received by the CSL agent depends only on this interaction. We use a basic Q-learning algorithm for recognizing norms in which states are the discretized current values of an agent's payoff matrices. Learning results in modifications to the certainty degree of available norms. Rewards are calculated based on the changes in the personal values.

7.1.4 Norms

The process of recognizing a social norm is modeled by an agent increasing the norm's certainty value to a positive value. The agent updates the certainty values of norms based on its observations after performing an action. Our norms are represented using the format introduced in [56], $\langle \Delta, C, A, E, S, R \rangle$, in which Δ designates the type of norm, C is the triggering condition, A and E show the activation and expiration period of the norm, and S and R indicate a reward or sanction. For example, this is an example of a possible norm: $(\langle \text{prohibition}, \text{littering}, -, -, \text{payfine}, - \rangle, \delta)$, which is always valid since there is no duration on activation, A, and expiration, E.

All of possible norms are initialized at the beginning of the simulation with the certainty value of zero. Agents update their norms by increasing or decreasing the certainty value of each norm after making an observation. For instance, if the agent receives a fine after littering, it will update its current value of (δ) in the above norm example with $(\delta + \epsilon)$, where ϵ is a user defined value.

An agent's current norms are used to update its beliefs and desires. The updating procedure is shown in Equations 7.3 to 7.5. Here, norms are abbreviated as N instead of $\langle \Delta, C, A, E, S, R \rangle$ in order to shorten the formulas. Here, if there are any relevant rewards R (or sanctions S), the positive desire D^+ (or a negative desire D^-) will be updated. f functions are user defined functions.

$$((N_i, \delta_N), (B[\alpha]\varphi, \delta_\phi)) \Rightarrow (B[\alpha]\varphi, f(\delta_N, \delta_\phi)) \tag{7.3}$$

$$((N_i, \delta_N), (D^+\varphi, \delta_\varphi), R \neq \varnothing) \Rightarrow (D^+\varphi, f(\delta_N, \delta_\varphi)) \tag{7.4}$$

$$((N_i, \delta_N), (D^-\varphi, \delta_\varphi), S \neq \varnothing) \Rightarrow (D^-\varphi, f(\delta_N, \delta_\varphi)) \tag{7.5}$$

As an example, if there exists the norm $(\langle$prohibition, littering, $-, -,$ payfine, $-\rangle, 0.75)$ and a negative desire toward paying fine $(D^-payfine, 0.55)$, assuming the agent has just paid a fine for littering $(S \neq \varnothing)$ with $f = \min(\max(0.75, 0.55), 1)$, the resulting updated desire would be $(D^-payfine, 0.75)$.

7.2 Experiments

To demonstrate the utility of our normative architecture, two case studies are presented. In first case study, we evaluate the performance of CSL at simulating norm emergence in a park scenario, as compared to the normative BDI (NBDI) and social learning (SL) architectures. The second case study is designed to evaluate the ability of CSL to model the propagation of norms in real-world environments. We compare the performance of our proposed architecture with the LNA architecture presented in Chapter 6 for simulating the propagation of smoking norms.

7.2.1 Park Case Study

This case study is designed to recreate the frequently observed "tragedy of the commons" in which humans are moving through a public area like a park and have the option to improperly dispose of trash and recycling on the ground, stow their waste for future disposal, or proactively recycle objects dropped by other passersby. Additionally, there are two other actions that the agents can perform, which are violating park visitor rules by feeding the animals and trespassing on the grass. Among this set of actions, littering, feeding animals and walking on the grass are negative, but potentially contagious, behaviors. Our scenario is a useful model for describing many public policy social dilemmas, and is more complicated than the *rules of the road* scenario, often used to simulate the emergence of driving conventions.

Agents - In this scenario, the agents have the following action selections: litter, recycle waste, violate park rules by feeding animals, violate park rules by trespassing on grass, or take no action. For these experiments, we fixed the population size at 1000. There is an observable vicinity defined for each agent. Within that range an agent can observe other agents' actions. A certain percentage of agents are assumed to be punishers (20 percent), which means they will punish agents who litter, feed animals, and walk on the grass, if those agents perform these actions in their observable area. Moreover, recycling while there is someone to observe the agent, will increase the reputation of agent.

Beliefs, Desires, and Intentions - Each agent has a set of beliefs, desires and intentions. Also, as part of its beliefs, each agent has a set of personal variables: *happiness*, *park usability*, *reputation*, *spent time*, and *paid fine*. The certainty values (δ) for beliefs and desires are assigned uniformly at random at the beginning of the scenario. Intentions are derived from the set of beliefs, desires and plans. The intentions are determined according to Equation 7.2.

Payoff Matrices - In both CSL and SL, the agent plays a game with the closest agent within its observable area each time that it needs to make an action decision. For each action, an agent has a two by two payoff matrix that determines the agent's decision. The agent picks the intention with the highest certainty value. The values of this payoff matrix are determined by the certainty degree of the selected intention, as described in the method section. This means that in our architecture, the intentions do not directly determine agent's actions, instead they define payoff matrix values. For instance, each time that an agent generates a piece of trash, and needs to decide whether to litter or not, it uses its littering payoff matrix, and plays a social dilemma game with the closest agent. Similarly, every time that the agent observes garbage in its vicinity it uses its recycling payoff matrix to decide whether to recycle the garbage or not. Since the agents move through the park in a random walk, they have the possibility of encountering new agents during every round.

```
init(blf, des, pln, q-tbl)
repeat
    generateIntention(blf, des, pln)                        ▷ Equation 2
    updatePMatrix(maxIntention)
    if (converged-Qtbl) then
        playGame(pMatrix,neighbors)
        performAction()
        update-qTable(rew, san)
    else
        performAction()
    end if
    update-norms(rew, san)
    update-beliefs(rew, san, norms)                         ▷ Equation 3
    update-desires(rew, san, norms)                         ▷ Equation 1, 4 & 5
until agent not selected
```

Figure 7.2: CSL pseudocode
(blf=Beliefs, des=Desires, pln=Plans, rew=Rewards, san=Sanctions)

Q-learning - The learning component is implemented using the Q-learning algorithm. The current values of the payoff matrices determine the states of the Q-table. The selected action modifies the certainty value of norms. After an agent performs an action, it observes the consequences of its action to compute the overall received payoff, which is then used to update the Q-table. Each of the agent's actions increases or decreases agent's personal variable values according to a fixed formula applied to all agents in the scenario. For example, littering would increase *happiness*, but would decrease *park usability*. Littering decreases *reputation* when there is an agent in the vicinity; in the presence of a punishing agent, the offending agent pays a fine.

Norms - All possible norms are initialized as having a certainty value of zero. During initialization, we create all of possible norm combinations based on the introduced norm representation: $\langle \Delta, C, A, E, S, R \rangle$. The type of norm and its reward or sanction nature can be determined by the value for C. We assume that all norms are always valid during the experiment, so we don't need to take A and E into account. Thus 24 possible norms are defined for this scenario: |obligation, prohibition, permission|*|littering, recycling, feeding animals, walking on grass|*|reward, sanction|.

Figure 7.2 shows the pseudocode describing an agent's behavior for one time-step in the CSL implementation. The certainty value of beliefs and desires are initialized uniformly at random at the beginning of the scenario.

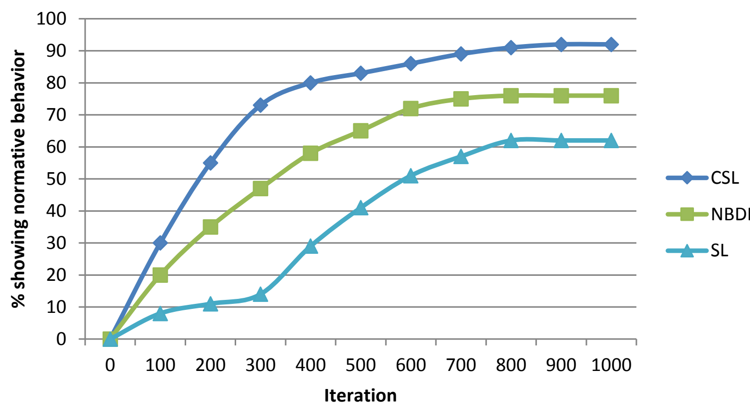


Figure 7.3: % of agents exhibiting normative behaviors

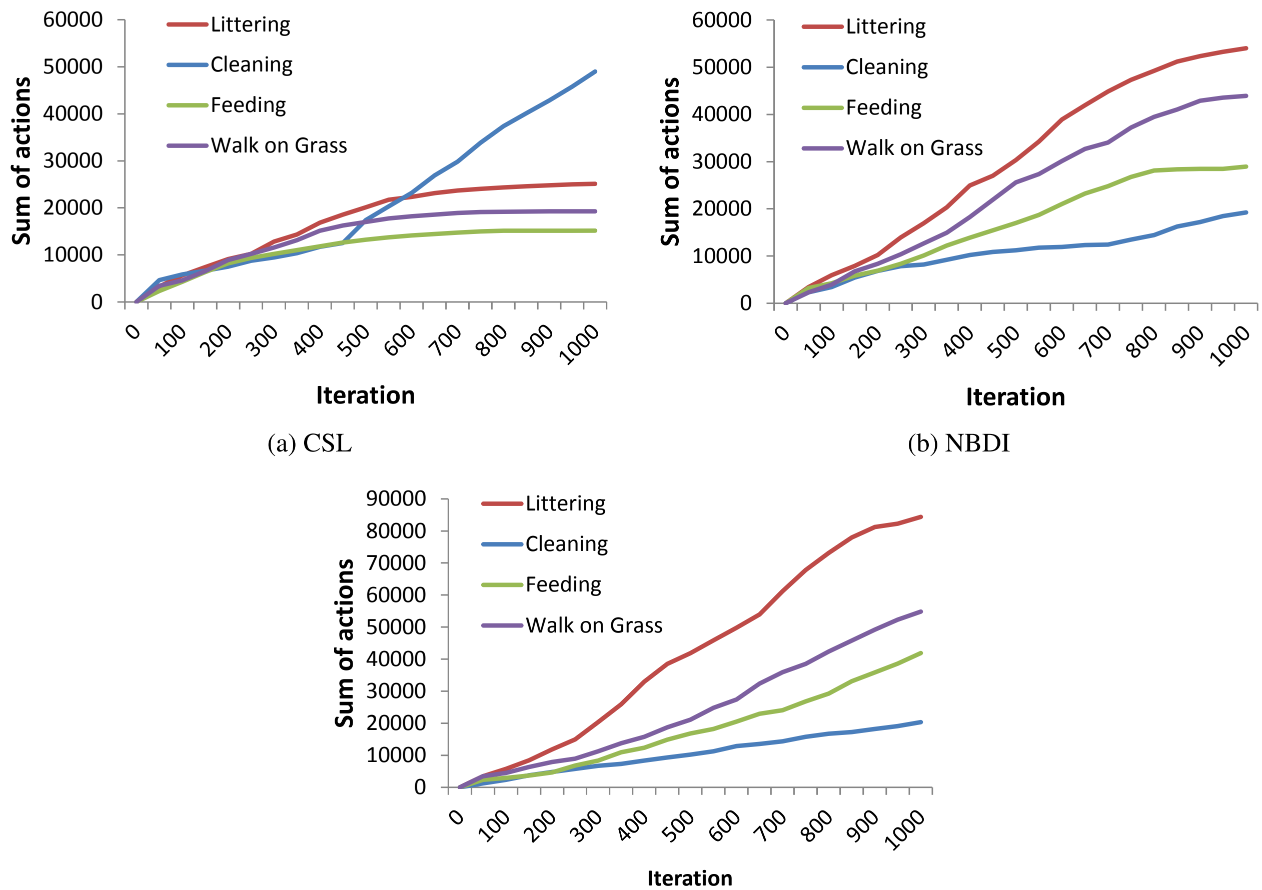
Results - Our proposed framework (CSL) was compared against two other benchmarks. The first one, NBDI, is a version of the normative BDI architecture described in [55], and the second one, SL, is the social learning framework introduced in [155]. In order to make a fair comparison between different architectures, the NBDI and SL frameworks are implemented by removing some of the components of CSL. The NBDI benchmark does not play the social dilemma game and does not use reinforcement learning to generate and update norms. In this case, intentions determine actions, and then the norms are updated based on the feedback received from the environment. Note that the way that the norm representation was implemented (by modifying the certainty value of norms) is not part of the original version of NBDI. The norm recognition part in the original NBDI was assumed to work as a blackbox, and there was insufficient detail about its implementation to recreate it. Hence we simply used the same norm recognition structure for both CSL and NBDI. For the SL framework, each agent has payoff matrices, and updates them using Q-learning. SL lacks the BDI representation, as well as the internal features and explicit norm representation. Results are presented for an average of 20 runs of the social simulation.

The percentage of agents demonstrating normative behavior is shown in Figure 7.3. The purpose of this experiment was to study the overall ability of the agent population to recognize and adopt to social norms. For each agent, normative behavior is assumed to be occurring when more than 90 actions of the agent's last 100 actions are normative actions. Normative actions refer to: recycling, not littering, not feeding animals and not trespassing on the grass. Obviously, their action is counted only when the agents have the possibility of performing these actions. For instance, an agent can only feed animals when they are within close proximity. As the chart shows, a greater percentage of the CSL agents evince normative behavior, compared to NBDI and SL.

Figures 7.4a, 7.4b and 7.4c illustrate differences between the cumulative normative vs. non-normative actions that were performed by a population of 1000 agents averaged over 20 runs of the models. The main goal of this experiment was to evaluate the ability of each method to propagate conformity to social norms. In all cases, the sum of all action types initially rises. In the CSL case, growth of non-normative behaviors reaches an asymptote while performance of the (normative) recycling behavior rises sharply. In NBDI and SL, the amount of recycling is low compared to the other behaviors. Moreover the speed and extent of norm emergence exhibited by CSL is more than the NBDI and SL methods.

7.2.2 Smoking Cessation Case Study

The performance of the CSL architecture was also measured in our real-world scenario, modeling the propagation of smoking cessation norms after a smoke-free campus initiative. The same smoking model as presented and used for the LNA architecture in Chapter 6 is also used for implementing CSL architecture. Here we compare CSL versus our proposed architecture LNA that was developed specifically for modeling normative smoking behavior [27], and was described in detail in Chapter 6. Here we present the details of our CSL model of smoking behavior.



(a) CSL

(b) NBDI

(c) SL

Figure 7.4: The recycling (cleaning) norm only strongly emerges in CSL, not in NBDI and SL.

Beliefs, Desires, and Intentions - The two first personal values, individualism and achievement, are implemented as fixed value elements of beliefs (Type 1). The remaining three personal factors, regret, health and hedonism, plus environmental factors are implemented as variables, and are part of each agent's beliefs. The certainty values (δ) for beliefs and desires are assigned uniformly at random at the beginning of the scenario. The intentions are determined according to Equation 7.2. The main desires and intentions defined in this system refer to smoking and not smoking.

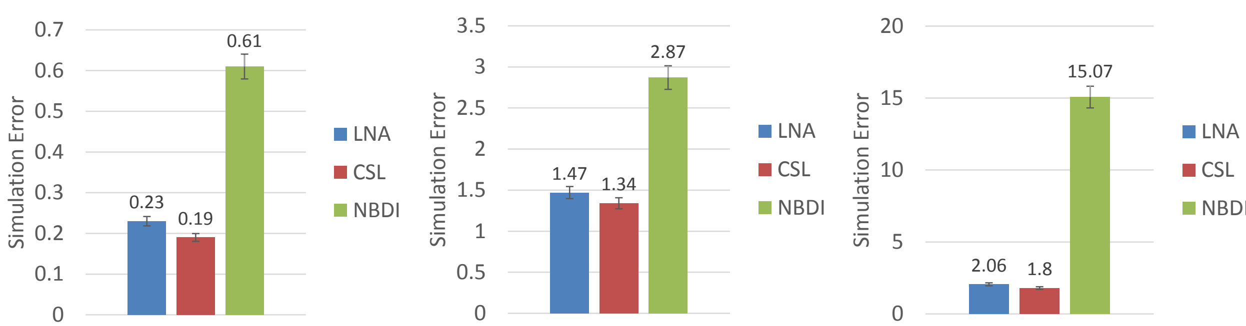
Payoff Matrices - An agent plays games with both its friends and other agents in close proximity to determine its actions. For each action, an agent has a two by two payoff matrix that determines the agent's decision. The agent picks the intention with the highest certainty value. The values of this payoff matrix are determined by the certainty degree of the selected intention, as described in the method section. This means that in our architecture, the intentions do not directly determine agent's actions, instead they define payoff matrix values. The friendship value (frd) in the smoking model is calculated using the payoff matrix values.

Norm Recognition - The learning component is implemented using the Q-learning algorithm. Actions are the action performed by the agent: smoke or not smoke. The reward value is assumed to be the same as the reward value defined for the reinforcement learning and smoking diffusion in LNA. The current values of the payoff matrices determine the states of the Q-table. The selected action modifies the certainty value of norms. After an agent performs an action, it observes the consequences of its action to compute the overall received payoff, which is then used to update the Q-table.

Norms - Norms are created using the same procedure introduced. Only dynamic (variable) parts of beliefs are updated. All possible norms are initialized as having a certainty value of zero. During initialization, we create all of possible norm combinations based on the introduced norm representation: $\langle \Delta, C, A, E, S, R \rangle$. The type of norm and its reward or sanction nature can be determined by the value for C. We assume that all norms are always valid during the experiment, so we don't need to take A and E into account. Thus 12 possible norms are defined for this scenario: |obligation, prohibition, permission|*|smoking, not smoking|*|reward, sanction|.

In order to have a fair comparison between the two methods, we modified the LNA model as little as possible. In addition to comparing CSL with LNA, we also examine the performance of the NBDI architecture on this dataset. Since LNA includes a component very similar to the social learning method, the SL method was not implemented independently.

Using these assumptions, we ran our agent-based simulation for a period of a year from Fall 2011 to Fall 2013. In these experiments, we initialized the simulation with the same number of smokers and students willing to go to the classes as the initial survey data, and then compare the numbers obtained from the simulation with the final survey data. During this period, the agents commute to campus and follow schedules governed by the transportation model. The proposed smoking model simulates the smoking behavior of students during the year of study. The average simulation error of ten runs of the model are reported in Figure 7.5. Simulation error refers to the difference between the values obtained from each method and the real value from the experimental data. The two measures shown here are the percentage of smoker students and the percentage of smoker students who are willing to attend smoking cessation classes. The empirical data for the percentage of smokers was also available for 2013.



(a) % of predicted smokers vs. empirical data for Fall 2012

(b) % of predicted smokers vs. empirical data for Fall 2013

(c) % of predicted students willing to attned classes vs. empirical data for Fall 2012

Figure 7.5: Comparison between the performance of different normative architectures. The simulation error refers to the difference between the obtained value by each method and the empirical survey data.

Figure 7.5 shows the comparison between the number of students who were smokers and students willing to participate in smoking cessation classes. The performance of CSL at predicting the actual adoption of the smoking cessation norm is comparable to the LNA and superior to NBDI. The statistical significance of the reported results for LNA and CSL are also shown in table 7.2.

Table 7.2: Statistical significance of reported smoking percentages using CSL and LNA

	Smoking %	p level
CSL-2012	0.19	0.032
CSL-2013	1.8	0.043
LNA-2012	0.23	0.024
LNA-2013	2.06	0.053

A powerful feature of agent-based models is their ability of predicting future trends. This can be a great tool for policy makers who want to analyze the effects of modifying various parameters of a specific model. In Figure 7.6 the predicted percentage of smokers for the period of years 2011 to 2016 is shown. The values shown for the years 2011 to 2013 are the same as shown in Figure 7.5. The current assumption in our model is that various properties of the whole system remain the same during the simulated years.

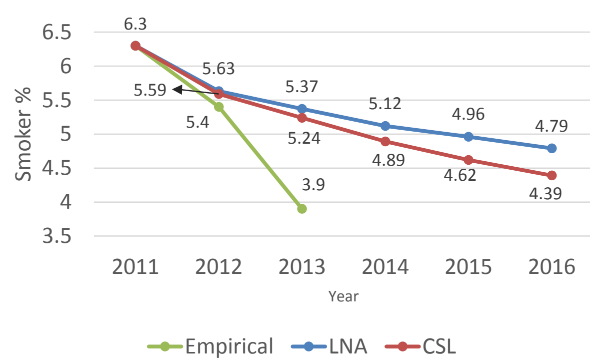


Figure 7.6: Predicted percentage of smokers for future years

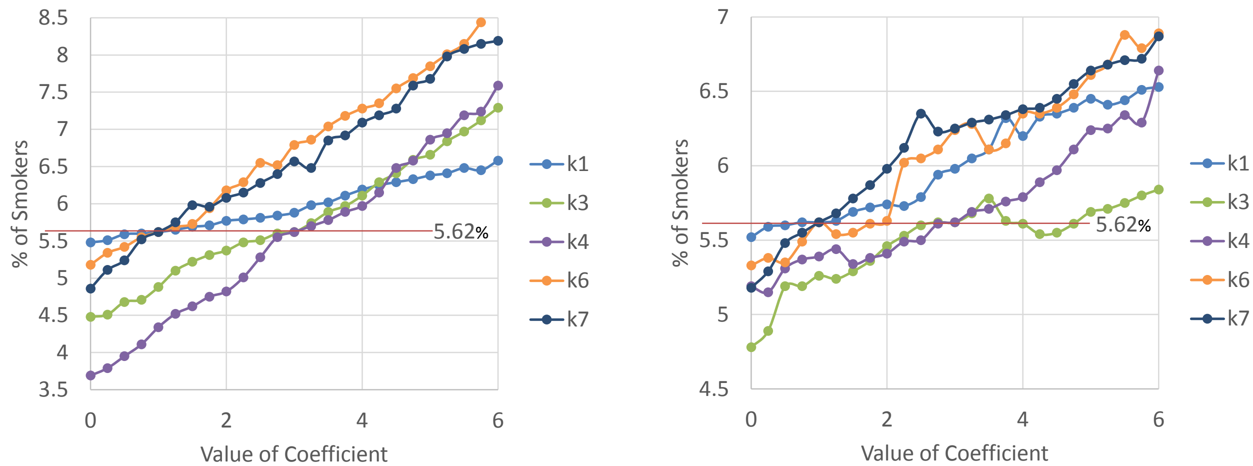
One factor that our model does not take into account is the gradual change of the population as students arrive to the school and graduate. It is worth noting that since the survey methods used by the UCF Health Services each year were slightly different, there could be differing forms of error in the reported statistics for each year. For the year 2013, it was confirmed by the Health Services department that the reported rate (3.9%) was a bit lower than what they were expecting based on national and state averages.

Table 7.3 shows a comparison between the different architectures at predicting the perceived social unacceptability of smoking. This phenomenon is reported in many smoking studies including [67] and [93] as occurring when smoking bans exist in human cities. Brown et al. [36] showed that perceived social acceptability of smoking among referent groups is independently associated with both strength of intention to quit and actual quitting behavior.

Table 7.3: Standard coefficient (Beta) values of the applied linear regression to perceived social acceptability of smoking (independent var.) and quit intention (dependent var.)

	Beta	*p* level
CSL	0.22	0.001
LNA	0.001	0.007
NBDI	-0.01	0.005

In our smoking model, it is assumed that an agent has the intention to quit smoking if its smoking value (SV) is within the first and second threshold values. The social unacceptability of smoking across the population of agents is determined using the value for one of the agent's personal characteristics (IND). The value of this factor was initialized based on data from a survey question asking whether the participant believes smoking is acceptable on campus. Following the works mentioned above, a linear regression model was used to examine the relationship between these two elements, and the standard coefficient (Beta) value of the applied linear regression is shown in Table 7.3. The CSL model produces a positive Beta value, which is consistent with the real-world data. This shows that, using CSL, agents are able to reason about the socially perceived unacceptability of smoking behavior, and modify their behaviors accordingly. Therefore, CSL is modeling norm emergence in a more realistic manner. On the other hand, the Beta values for the LNA and NBDI architectures is close to zero, which does not accurately reflect the results reported in independent smoking studies.

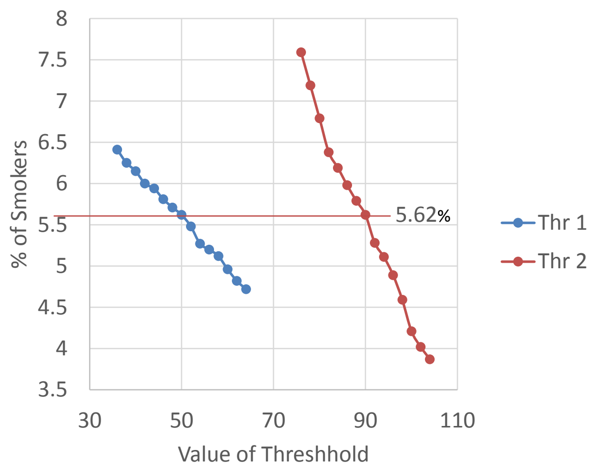


(a) LNA (b) CSL

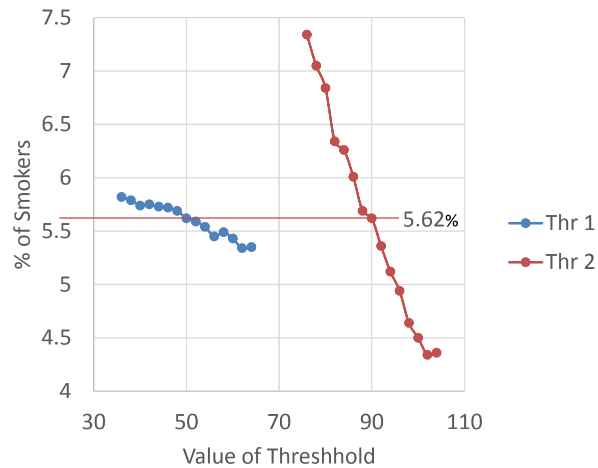
Figure 7.7: Sensitivity analysis for five coefficient values used for determining the final smoking value in our models. Horizontal red lines show the current values used by our models.

In addition, we performed a sensitivity analysis on the results that we obtained from the the two architectures. Since our models include a number of variables that directly affect the final behavior of our system, the sensitivity analysis can help us understand the extent of the effect that each variable can have on the final outcome. In order to do that, five of the ten coefficients that were used in Equation 6.2, plus the the two threshold values for determining the three stages of norm formation are used as the independent variables in our sensitivity analysis model. The remaining five coefficients are not shown due to their close relationship to the current coefficients. The analysis is done on one independent value at a time.

Figure 7.7 shows the range of output values for different values that can be assigned to five of the k_i coefficients, and similarly Figure 7.8 shows the output range for the two threshold values.



(a) LNA

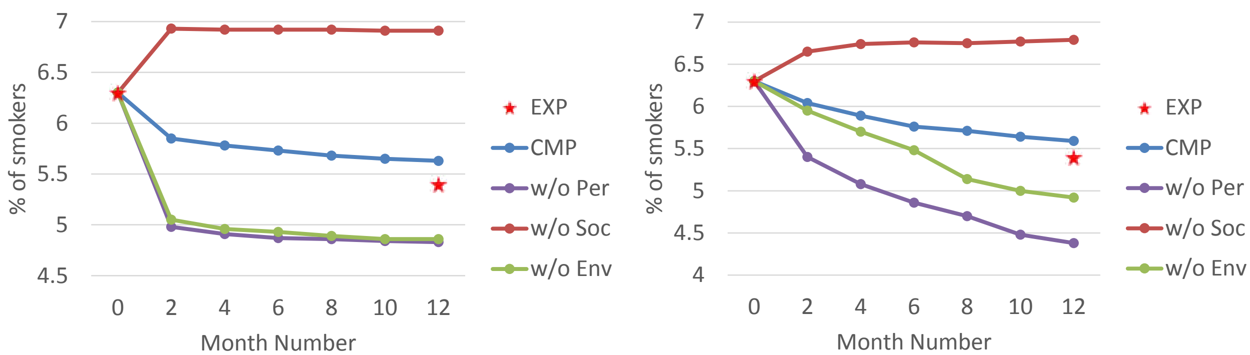


(b) CSL

Figure 7.8: Sensitivity analysis for the two threshold values in our models. Horizontal red lines show the current values used by our models.

By comparing the results shown in Figure 7.7a with 7.7b, and also 7.8a with 7.8b, we can observe that LNA seems to be more sensitive to parameter choice than CSL. By changing the coefficient values from 0 to 6, the maximum change in the percentage of smokers is close to 4 for LNA, and less than 2 for CSL. For the two threshold values (shown in Figure 7.8) LNA's results vary across a range of 3.5, while CSL's range is less than 2.5. Overall, the sensitivity of the model's output to the set of input values is low, and because of type of equation used by the model, the output range for different values remains linear.

We also study ablated versions of the CSL model that lack one of the three smoking elements (social, environmental, or personal). The results for alternate months during the year of simulation are reported in Figure 7.9. The reported results are, again, averaged over ten runs, and in all cases the initialization configuration is based on the survey data. In Figures 7.9a and 7.9b the left red star shows the starting value which is the empirically measured value, and is the same for all four experiments. Without the personal and environmental components, the model tends to underestimate results in comparison to the final empirical results. Without the social part, the model overestimates smoking behavior. Based on the size of differences between the empirical results and the other experiments for CSL, it can be concluded that the personal values are the major predictors in determining smoking behaviors. Environmental factors had the lowest impact on predicting smoking behaviors.



(a) LNA (b) CSL

Figure 7.9: The percentage of smoker students in LNA (a), and in CSL (b) during the first year simulation period. The numbers from the survey data are marked by the red star icons at the beginning and end of the simulation period (Experimental/EXP). The figure shows the predictions of the proposed model (complete/CMP), the model without the personal values, without the social aspect, and without environmental influence. There is a close match between the predicted values of the complete model and the survey data.

7.2.3 Discussion

The LNA architecture presents a fairly simple normative structure. This structure is very similar to many of normative structures that are currently being employed for building normative multi-agent systems. The reader can refer to [131] for details, and a review of normative architectures. On the other hand, CSL is representative of the group of architectures that employ richer structures for normative reasoning. CSL integrates internal cognitive structures with social interaction mechanisms.

The results shown from the experiments on these two models demonstrate that CSL can produce more realistic results. This indicates that we should anticipate agent design to become more complex as our problems become more complicated, and the number of variables in the problem increases. Specifically, when it comes to modeling the intricacies of humans' behaviors – like the correlation between the unacceptability of smoking in a society and quitting intention – simple agent architectures may be inaccurate building accurate models. Additionally, simpler design structures are potentially more sensitive to parameter choice. As we observed in the case of our two models, LNA was more sensitive to input values than CSL.

7.3 Conclusion

Normative multi-agent systems are a promising computational mechanism for representing group influences on human social behavior and creating large-scale social simulations for a variety of interesting public policy questions. This chapter presents a normative architecture, Cognitive Social Learners, that bridges the gap between two lines of research on norms. We benchmarked our architecture against three other models (NBDI, SL, and LNA) at predicting the adoption of sustainable practices. Performance of the CSL architecture was evaluated on the smoking case study that was presented in the previous chapter. Our results indicate that the CSL architecture is more robust than models that rely exclusively on internal or external processes at modeling norm emergence in complex real-world scenarios.

CHAPTER 8: MODELING TIPPING POINT THEORY USING NORMATIVE MULTI-AGENT SYSTEMS

Human societies are simultaneously frustratingly unchanging and yet susceptible to "epidemics" that sweep across the social fabric causing people to adopt previously rare practices. Tipping point theories attempt to explain the subtle triggers behind these social processes. In 2000, Malcolm Gladwell [125] produced a popular science book summarizing three key factors which trigger tipping points: 1) scale-free networks (the Law of the Few); 2) effective messaging (the Stickiness Factor) and 3) environmental influences (the Power of Context). This section relates tipping point theory to the process of norm emergence in multi-agent systems; we propose that normative agent architectures can serve an excellent computational model for expressing many contagious social phenomena, including tipping points and information cascades.

As was discussed in previous chapters, social norms are known to be a major factor governing humans' behavior; unbeknownst to us, many of our everyday behaviors are influenced by these implicit standards. Various normative architectures have been proposed for designing normative multi-agent systems (NorMAS) capable of reasoning about norm adoption. Some of these systems have been grounded in social science theory, but the aim of many architectures is simply to effectively address standard multi-agent system challenges, including agreement formation, coordination and conflict resolution.

Despite recent research progress in the area, the complete life-cycle of norms is far from fully understood. The complex nature of human decision-making makes comprehending the rationale behind social interactions difficult, since people are notoriously bad at self-reporting their motivations. The field of agent-based modeling aims to create agents in the image of humans. These agents typically have cognitively-inspired decision-making components, and are situated in life-like scenarios. In both standard multi-agent systems and cognitively-inspired models, existing social theories have been employed toward the construction of normative models. Various stages of the norm life-cycle including recognition, adoption, compliance and emergence are often modeled on similar concepts in social sciences.

This section proposes a unified model of how norm emergence in networked agent societies can be used to predict the effects of common tipping point triggers [28]. Previous work on norm emergence in networks has investigated the effects of social network topology in static [164, 154] and also dynamic networks [149]. Yu et al. [172] presented an evaluation of different learning methods on norm emergence in networked systems. In our work, we simply employ network structures as a medium to apply ideas from tipping point theory relating to the Law of the Few. Therefore, the structure of agents' network is not of interest by itself, other than making it congruent with human social networks.

The main purpose of this part of dissertation is showing the role and significance of tipping point principles in normative multi-agent systems (NorMAS), and evaluating the potential impact of this model on NorMAS design. Here, the impact of Gladwell's three factors on norm emergence in agent-based normative systems is studied and practical ways to apply this versatile theory is demonstrated. These three factors are the role of a few members of society, stickiness of message that is being passed and the role of environment. This is done through a set of experiments on a driving case study. The details about the experiments are described in the next section.

8.1 Experimental Setup

For the experiments, the classic scenario of rules of the road is employed, that is frequently used to study normative behavior in multi-agent systems. In this scenario, there exists a population of agents that do not have any preference toward driving on the left or right side of a two-way road. No rules or higher enforcement exist to determine the preferred side. This scenario represents a two-action stage game that models the situation where agents need to agree on one of several equally desirable alternatives. The societal norms that evolve in this domain are either driving on the left or driving on the right [155].

In this scenario agents receive a fixed value reward and punishment based on the following payoff matrix shown in Table 8.1.

Table 8.1: Payoff matrix for rules of the road scenario

	left	right
left	1,1	-1,-1
right	-1,-1	1,1

As Yu et al. [172] note, although this payoff matrix appears simple, the coordination game poses a very challenging puzzle for human beings to solve efficiently. The game has two pure Nash-equilibria: both agents drive left or both agents drive right. Classical game theory, however, does not give a coherent account of how people would play a game like this. The conundrum is that there is nothing in the structure of the game itself that allows the players (even purely rational players) to infer what they ought to do. In reality, people can play such games because they can rely on some contextual cues to agree on a particular equilibrium [171].

In similar studies on normative systems, usually the cumulative payoff (reward) of the whole population of agents is used as a measure of comparing various methods (see [155] and [172] for examples). Instead, the norm emergence time for each method is used as an evaluation method here. This is functionally equivalent since the payoff received by all agents post norm emergence is the same, hence a method which leads to faster norm emergence will also yield the higher cumulative payoff.

8.2 Key Few Members

In this section, the effect of key members of an agent society on the rate of norm emergence is studied. These key members are selected using standard heuristics for measuring influence within a network; the performance of three centrality measures: degree, closeness, and betweenness is evaluated. Degree centrality measures the number of edges connected to a node. Closeness is calculated based on the total distance to all other nodes. Nodes with a high betweenness centrality fall on a large proportion of the shortest paths (geodesics) in the graph.

To model the characteristics of a real social network, the same algorithm from Section 6.3.2 (originally introduced in [165]) is employed to create a synthetic network which follows a power law degree distribution and exhibits homophily, a greater number of link connections between similar nodes. The network generator uses link density (*ld*) and homophily (*dh*) to govern network formation. A simplified version of the pseudo-code for this method is shown in Figure 6.2. Predefined values for *ld* and *dh* are assumed. The nodes of the graph represent the individuals (agents) in the simulation, who can be considered as car drivers.

A weighted voting approach (also known as a structure based method) to determine an agent's decision with regard to its neighbors is used. The weight for each of an agent's neighbors is computed using a normalized value of that neighbor's centrality value as shown in Equation 8.1.

$$\text{weight}_{i,j} = \frac{C_j}{\Sigma_{k=1}^{Deg_i} C_k} \tag{8.1}$$

This equation shows the weight of the link connecting neighbor j to node i. C refers to the corresponding centrality value (degree, betweenness and closeness). Also, Deg_i denotes the number of neighbors for node i. The top 10 percent of the population of agents with the greatest centrality values are assumed to be the key elements of a society. At the beginning of the experiments, all of the agents follow a single norm; in other words, all of them have learned (through social learning [155]) to always drive on one side of the road. Each agent has a utility value defined for each of four possible cases: Up-Left, Up-Right, Down-Left and Down-Right, where Up and Down determine the section of road, and Left and Right determine the direction an agent drives. Figure 8.1 shows a snapshot of designed agents. These values are updated while receiving payoffs based on the matrix shown in Table 8.1.

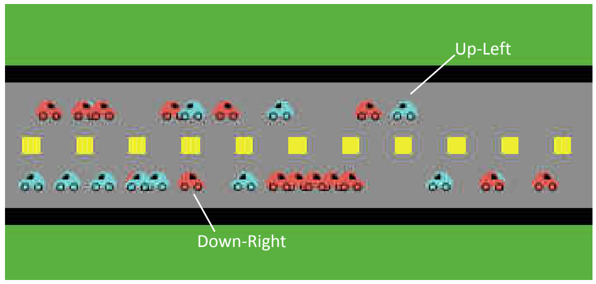


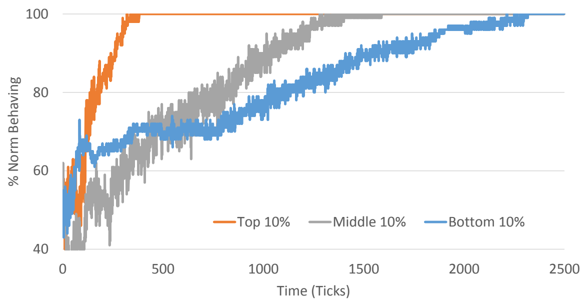
Figure 8.1: A simple graphical view of the agents designed for the *rules of the road* scenario.

In these experiments, the penetration of norm changing behaviors that emanate from key members of a society is compared vs. other cases. Emanation from the top is compared to emanation from the middle and bottom 10 percent of the population. At the beginning of the simulation, the agents (nodes) are ranked based on their centrality value to determine the top, middle and bottom agents. The utility value of these agents is kept fixed. Neighbors of these agents continue updating their behavior until a new norm emerges in the system. Figure 8.2a, Figure 8.2b, and Figure 8.2c show the number of iterations required for each case to converge. The population of agents contained 100 agents, and the reported results show the average values over 20 runs.

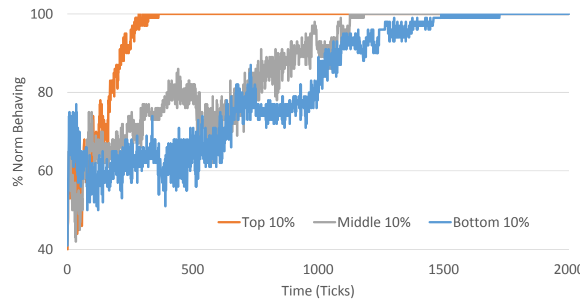
The pattern observed in all of three cases was very similar. When the norm propagation starts from the top 10% of the population, the norm emerges much faster compared to the other cases. Moreover, there is a fairly sizable difference among top, middle and bottom agents. The magnitude of difference between the top and middle 10% is more than the difference between the middle and bottom. These results are consistent with the role of connectors in tipping point theory.

8.3 Stickiness Factor

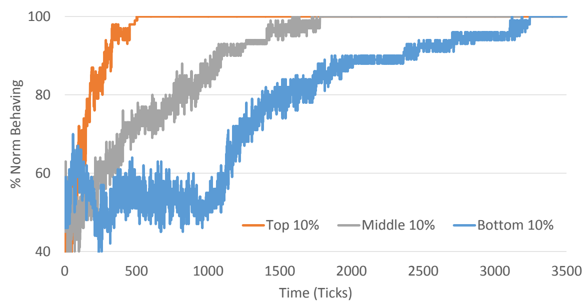
According to the tipping point theory, the extent and rate of emerging social norms in a society is not only related to the members of the society, but also related to the content of the message. An effective message needs to be interesting or "sticky" enough to remain in agents' minds. This factor is almost completely independent of the society and its structure, and is a property of the idea.



(a) degree



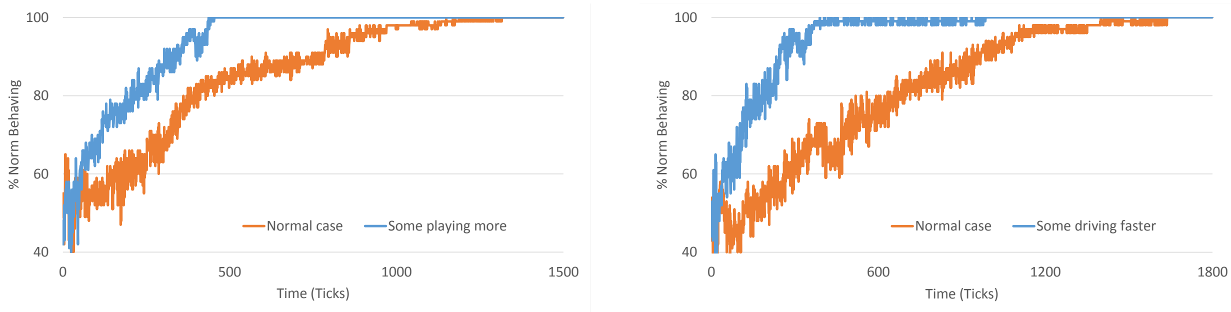
(b) betweenness



(c) closeness

Figure 8.2: Average number of iterations until the emergence of a norm in the population, when using degree centrality (a), betweenness centrality (b) and closeness centrality (c) to determine key agents.

As Gladwell [88] points out, it is potentially very complicated to determine if a certain message has the necessary stickiness or not, but one characteristic that is usually common to sticky ideas is that it frequently returns to a person's mind. This could be in the form of a desire to sit and watch a popular TV show every night, or in a more extreme case, a clinical addiction to smoking or gambling. Conventional marketing and advertising domains refer to this phenomenon as the *rule of 27*. According to this rule, a message (advertisement) should be seen at least 27 times, if the message is going to stick [138].



(a) (b)

Figure 8.3: Average number of iterations until the emergence of one norm, when 2 out of 4 agents with fixed utility values play twice with each agent that they encounter (a), and when 2 out of 4 agents with fixed utility values go (drive) faster (b).

In order to model this property, it's assumed that the stickiness is represented by the number of games that an agent plays with another agent. Therefore a higher number of games will result in the same effect as a stickier belief. In the experiments, this idea is evaluated in two different ways. The first way is to increase the number of games that a certain set of agents play. The second way is to have a certain number of agents driving faster than other agents to be exposed to more cars.

Figures 8.3a and 8.3b show results related to these two cases. In both cases, original 100 agents exist plus a group of 2 agents which have a fixed preference to drive on either the left or right. In the first scenario, one group of agents plays two games each time it encounters another agent. In the second scenario, one group of agents moves faster. Both of these scenarios lead to the same effect: increasing the number of times that an agent is exposed to an idea. This simulates the property of frequently returning to a person's mind. In both cases, when the stickiness factor is implemented, the entire system converges to a single norm faster.

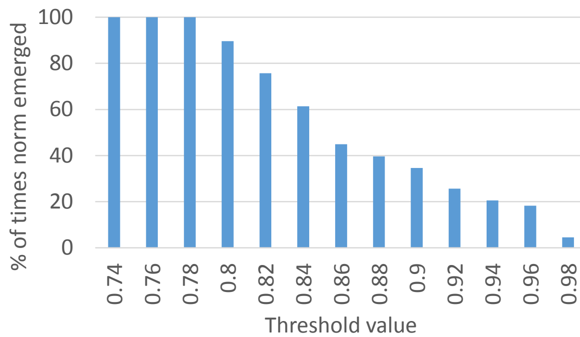


Figure 8.4: Percentage of times that a norm emerges in the population, when agents have different threshold values for activating.

8.4 Power of Context

The third element of the tipping point theory refers to the power of context. As Gladwell points out: it is possible to be a better person on a clean street or in a clean subway, than in one littered with trash and graffiti [88]. The idea is mostly based on what's known in criminology as the theory of *broken windows* [170]. According to this theory, slight changes in the environment could result in tipping effects over the whole society.

In order to apply this part of the tipping point theory, some ideas from the a set of techniques for studying fads and cascading effects in networks [167] are used. First, a network is built using the same approach described in Section 8.2. Then, a threshold value is assigned for each agent. Similar to the probabilistic information cascade models, if the cumulative value of the perceived cascade is less than the threshold, nothing will change. If it's higher, the agent will change its current behavior, which in our scenario would result in driving on the other side of the road. Figure 8.4 shows the percentage of times that a norm emerged in the system for a set of threshold values. The columns show the average results over 20 runs. Agents were selected randomly as a source of a small initial shock in the network, which results in negating the current payoff values for driving on each side of the road. The frequency of shocks is determined randomly. The system runs until it reaches some fixed iteration number (50,000), unless a different norm is observed. This experiment illustrates how minor shocks can shape a population fad, resulting in a population-level behavior change. The shocks (pulses) in this model can be viewed as any of the small changes that tipping point theory predicts can result in large changes in the whole society. According to the results presented in Figure 8.4, thresholds as big as 0.98 (as small as 2% percent activation chance) can lead to the emergence of norms in the system in almost 5 percent of the experiments. The computed values for each agent are compared to its tipping point value (normalized between 0 and 1).

There is a second aspect to the power of context, which refers to the number of people in groups. The Rule of 150 says that the size of groups is a subtle contextual factor that makes a big difference. This number is referred as *Dunbar's number* [68], after the anthropologist who originally proposed the idea. In groups with fewer than 150 members, people will cooperate relatively easily and rapidly become infected with the community ethos. Once that threshold is crossed, people begin to behave very differently. 150 is our *social channel capacity* as determined on the basis of personal loyalties and 1-on-1 contacts. Beyond the tipping point of 150 the group dynamics simply become too complex. For the average person there are just too many relationships to manage. The group then becomes divided and alienated, and usually splits into two. Smaller groups have been shown to be more effective at tasks than larger groups. This may be due to biological limitations of humans which make it very difficult for them to handle a larger community.

With the growth of virtual social media sites and the spread of online groups, there has been renewed interest in evaluating the importance of this limit on Facebook [69], Myspace [85] and within massively multiplayer online role-playing games (MMORPGs). The pivotal issue here is that a person cannot maintain a close relationship with all of the members of a larger group which ultimately sabotages its success. Having a direct connection with each member of the group is a necessary component to having a positive social relationship.

A clique structure is proposed to be used to illustrate this idea. In a clique each node has a direct edge to all of other nodes. There are $n(n-1)$ edges in the resulting graph. A directed graph is used here, as that seems to be the general assumption for friendship networks. The emergence process of driving norms in a network generated using the synthetic network generator described in Section 8.2 is studied. It should be noted that having more edges does not necessarily result in faster convergence. More connections makes the diffusion of ideas easier, while it makes it harder for the agents to find an idea that all agents like. In a clique structure, the major voting approach and the weighted voting approach (using the number of edges) are the effectively same; so neither of them elicits earlier norm emergence. Figure 8.5 shows the number of iterations that were required on average for the two cases to reach norm emergence. The one-side driving norm emerged faster in the case of clique structure than in the power-law degree distribution network. This shows the potential benefit of such a structure in constructing agent systems, at least for ideal cases.

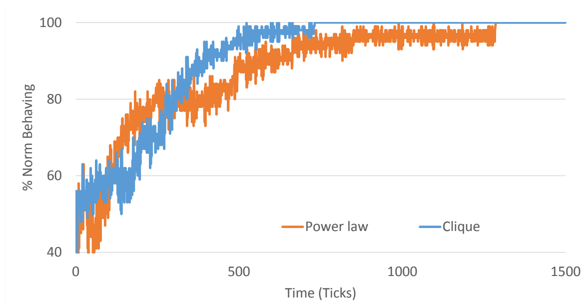


Figure 8.5: Average number of iterations until the emergence of one norm, when the network structure of agents follows a power-law distribution and when the network is a complete clique.

8.5 Conclusion

Norms are complex social behaviors that have been extensively studied in sociology, psychology, and other related fields. Most normative architectures draw upon theories from the social sciences. The theory of tipping points has inspired much research in different disciplines. For this section, some of the well-known elements of this theory are modeled, as applied to networked agent populations. It is illustrated that how three of the principal ideas including key few members, stickiness factor, and the role of environment can affect the process of norm emergence. These experiments are an attempt to illustrate the value of tipping point theory concepts to the NorMAS community.

CHAPTER 9: CONCLUSION AND FUTURE WORK

The overarching aim of my research is to create a general purpose normative agent-based modeling and simulation system for studying the effects of public policy decisions on a large range of social phenomena, including personal health decisions, sustainability behaviors, and opinion formation. Norms are an important part of human social systems, governing many aspects of group decision-making. Discovering the details about how social norms emerge in societies, and how they affect human activities enables us to have a better understanding of human behaviors in general. Specifically, constructing normative structures that can be employed in designing life-like simulations has many applications in domains such as public policy management, clinical health promotion and advertising.

The main contribution of this dissertation is introducing a new normative architecture, Cognitive Social Learners (CSL), that models bottom-up norm emergence through a social learning mechanism, while using BDI (Belief/Desire/Intention) reasoning to handle adoption and compliance. We demonstrate that the proposed architecture can be used to create a predictive model of the effect of UCF's smoke-free campus initiative on student smoking cessation trends.

At the beginning of this research, a detailed online survey about the transportation preferences of UCF students was designed, which was distributed to all the students via email. Using the collected data in combination with other sources, an agent-based model which simulates student transportation patterns was created. Agents in this model represent the students. Each simulated agent has a unique profile determining its actions. These profiles have statistically the same features as the collected datasets. The model can be employed to estimate statistics about UCF campus, including parking usage, car traffic and buildings' occupancy rate.

This model was extended by using the obtained samples from the agent-based model as an input for a Markov Chain Monte Carlo (MCMC) based component. This method was used to construct a more accurate model. In addition, the idea of merging agent-based modeling and

MCMC was a novel contribution of this dissertation. The mathematical logic for hybridizing these two methods was shown using category theory.

Once a reliable model for simulating transportation patterns of UCF students was built, it was merged into a detailed simulation of smoking cessation trends on campus. This model was validated with some independently collected data by Health Services at UCF. The purpose of building the model was to study students' smoking behaviors after the university started to become a smoke-free campus. In addition to following the general structure of CSL architecture, the proposed normative architecture for smoking includes three sets of factors: personal, social and environmental factors. The goal was to build an all-inclusive structure that contains the factors that can potentially affect one's smoking behavior, and implement this structure in a way that follows the general phases that are defined for norm emergence in computational normative studies. Using the introduced factors, three phases of recognition, adoption and compliance are mapped to the agents' smoking behaviors. This model employs a range of techniques from different AI domains including game theory, machine learning and social networks. This, itself, was part of a broader goal to build an effective way of simulating social norms in realistic scenarios, which are known to be complex by nature.

The theory of tipping points refers to a set of ideas in social sciences that describes how social phenomena like fads emerge in human societies. In order to expand the theoretical basis of the proposed normative architecture, some elements of this theory were used. These were the three elements popularized by Malcom Gladwell in his relevant book, which are role of key people, stickiness of messages and role of environment. Techniques from social network analysis such as centrality measures were used to implement tipping point theory ideas in normative models.

There are a number of possibilities for future work. One would be integrating tipping point theories with CSL. Another would be to apply CSL to more scenarios, such as modeling recycling behaviors of UCF students.

The model that was presented in this dissertation presents a cohesive structure for studying a set of complex human social behaviors. One can view this model as a novel way of studying social phenomena compared to lab-based or theoretical approaches currently used in these domains. Additionally, a procedure for constructing social simulations using survey data was introduced in this work. This process starts with initializing an agent-based model based on the survey data. Then, the agents that follow the proposed realistic normative architecture are built. The model runs for a certain period of time, and a population of desired samples is obtained from the agent-based model. These samples can be improved using the hybrid ABM-MCMC method. With these samples, it is possible to compute any statistical quantity of interest about the model.

In addition to presenting theoretical ideas and algorithms in this work, the presented ideas were successfully applied to several real applications which could be beneficial to the whole UCF community.

APPENDIX : IRB APPROVAL

University of Central Florida Institutional Review Board
Office of Research & Commercialization
12201 Research Parkway, Suite 501
Orlando, Florida 32826-3246
Telephone: 407-823-2901 or 407-882-2276
www.research.ucf.edu/compliance/irb.html

Approval of Exempt Human Research

From: **UCF Institutional Review Board #1**
FWA00000351, IRB00001138

To: **Gita Reese Sukthankar**

Date: **March 14, 2012**

Dear Researcher:

On 3/14/2012, the IRB approved the following activity as human participant research that is exempt from regulation:

Type of Review:	UCF Initial Review Submission Form
Project Title:	Learning Models of Human Activity and Transportation Patterns
Investigator:	Gita Reese Sukthankar
IRB Number:	SBE-12-08282
Funding Agency:	None

This determination applies only to the activities described in the IRB submission and does not apply should any changes be made. If changes are made and there are questions about whether these changes affect the exempt status of the human research, please contact the IRB. <u>When you have completed your research, please submit a Study Closure request in iRIS so that IRB records will be accurate</u>.

In the conduct of this research, you are responsible to follow the requirements of the <u>Investigator Manual</u>.

On behalf of Sophia Dziegielewski, Ph.D., L.C.S.W., UCF IRB Chair, this letter is signed by:

Signature applied by Janice Turchin on 03/14/2012 10:05:20 AM EST

IRB Coordinator

LIST OF REFERENCES

[1] National Social Norms Institute. `http://www.socialnorms.org/index.php`.

[2] R. L. Akers and G. Lee. A longitudinal test of social learning theory: Adolescent smoking. *Journal of Drug Issues*, 26:317–343, 1996.

[3] R. Akhavian and A. H. Behzadan. Construction activity recognition for simulation input modeling using machine learning classifiers. In *Proceedings of the 2014 Winter Simulation Conference*, WSC '14, pages 3296–3307, 2014.

[4] R. Akhavian and A. H. Behzadan. Construction equipment activity recognition for simulation input modeling using mobile sensors and machine learning classifiers. *Advanced Engineering Informatics*, 2015.

[5] M. Aminnayeri, E. A. Torkamani, M. Davodi, and M. Ramtin. Short-run process control based on non-conformity degree. In *Proc. World Congress in Engineering*, volume 3, pages 2273–2276, 2010.

[6] C. Andrieu, N. De Freitas, A. Doucet, and M. Jordan. An introduction to MCMC for machine learning. *Machine learning*, 50(1):5–43, 2003.

[7] C. Andrieu and É. Moulines. On the ergodicity properties of some adaptive MCMC algorithms. *The Annals of Applied Probability*, 16(3):1462–1505, 2006.

[8] G. Andrighetto, M. Campenní, F. Cecconi, and R. Conte. How agents find out norms: A simulation based model of norm innovation. In *International Workshop on Normative Multi-agent Systems (NorMAS)*, pages 16–30, 2008.

[9] G. Andrighetto, M. Campennì, R. Conte, and M. Paolucci. On the immergence of norms: a normative agent architecture. In *Proceedings of AAAI symposium, social and organizational aspects of intelligence*, 2007.

[10] G. Andrighetto and R. Conte. Cognitive dynamics of norm compliance. from norm adoption to flexible automated conformity. *Artificial Intelligence and Law*, 20(4):359–381, 2012.

[11] G. Andrighetto, R. Conte, P. Turrini, and M. Paolucci. Emergence in the loop: Simulating the two way dynamics of norm innovation. *Normative multi-agent systems*, 7122, 2007.

[12] G. Andrighetto, S. Cranefield, R. Conte, M. Purvis, M. Purvis, B. T. R. Savarimuthu, and D. Villatoro. (Social) norms and agent-based simulation. In S. Ossowski, editor, *Agreement Technologies*, volume 8, pages 181–189. Springer, 2013.

[13] G. Andrighetto and D. Villatoro. Beyond the carrot and stick approach to enforcement: An agent-based model. *European Perspectives on Cognitive Science*, 2011.

[14] G. Andrighetto, D. Villatoro, and R. Conte. Norm internalization in artificial societies. *AI Communications*, 23(4):325–339, 2010.

[15] S. Awodey. *Category theory*. Oxford University Press, USA, 2006.

[16] R. Axtell. Economics as distributed computation. In *Meeting the Challenge of Social Problems via Agent-Based Simulation*, pages 3–23. Springer Japan, 2003.

[17] J. C. Baez and A. Lauda. A prehistory of n-categorical physics. *Deep Beauty*, page 13, 2009.

[18] S. Balbi and C. Giupponi. Reviewing agent-based modelling of socio-ecosystems: a methodology for the analysis of climate change adaptation and sustainability. *University Ca'Foscari of Venice, Dept. of Economics Research Paper Series*, (15_09), 2009. Working Paper No. 15/WP/2009.

[19] T. Balke, T. Roberts, M. Xenitidou, and N. Gilbert. Modelling energy-consuming social practices as agents. In *Social Simulation Conference*, Barcelona, 2014.

[20] O. Barreteau, E. Sauquet, J. Riaux, N. Gailliard, and R. Barbier. Agent based simulation of drought management policy in practice. In *International Workshop on Agent-based Mod-*

eling for Policy Engineering (AMPLE 2012), European Conference on AI, pages 29–44, September 2012.

[21] F. M. Bass. Comments on a new product growth for model consumer durables the Bass model. *Management Science*, 50(12):1833–1840, 2004.

[22] R. Beckman, C. Kuhlman, A. Marathe, E. Nsoesie, and S. Swarup. Modeling the spread of smoking in adolescent social networks. In *Proceedings of the Fall Research Conference of the Association for Public Policy Analysis and Management*, Washington DC, 2011.

[23] R. Beheshti, A. M. Ali, and G. Sukthankar. Cognitive social learners: An architecture for modeling normative behavior. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 2017–2023, Austin, TX, January 2015.

[24] R. Beheshti and G. Sukthankar. Extracting agent-based models of human transportation patterns. In *Proceedings of the ASE/IEEE International Conference on Social Informatics*, pages 157–164, Washington, D.C., dec 2012.

[25] R. Beheshti and G. Sukthankar. Analyzing agent-based models using category theory. In *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 280–286, Atlanta, GA, nov 2013.

[26] R. Beheshti and G. Sukthankar. Improving Markov Chain Monte Carlo estimation with agent-based models. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, volume 7812, pages 495–502. Springer, 2013.

[27] R. Beheshti and G. Sukthankar. A normative agent-based model for predicting smoking cessation trends. In *Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems*, pages 557–564, 2014.

[28] R. Beheshti and G. Sukthankar. Modeling tipping point theory using normative multi-agent systems (extended abstract). In *Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems*, Istanbul, Turkey, May 2015. (to appear).

[29] I. Benenson, P. Torrens, W. Europe, and J. Portugali. Geosimulation: automata-based modeling of urban phenomena. *Environment and Planning B: Planning and Design*, 31(4):589–613, 2004.

[30] C. Bicchieri. *The Grammar of Society: The Nature and Dynamics of Social Norms*. Cambridge University Press, 2006.

[31] G. Boella and L. van der Torre. Norm governed multiagent systems: The delegation of control to autonomous agents. In *Intelligent Agent Technology, 2003. IAT 2003. IEEE/WIC International Conference on*, pages 329–335. IEEE, 2003.

[32] G. Boella and L. van der Torre. An architecture of a normative system: counts-as conditionals, obligations and permissions. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 229–231. ACM, 2006.

[33] G. Boella, L. Van Der Torre, and H. Verhagen. Introduction to the special issue on normative multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 17(1):1–10, 2008.

[34] F. Borceux. *Handbook of Categorical Algebra: Volume 3, Sheaf Theory*, volume 1-3. Cambridge University Press, 1994.

[35] J. Broersen, M. Dastani, J. Hulstijn, Z. Huang, and L. van der Torre. The BOID architecture: conflicts between beliefs, obligations, intentions and desires. In *Proceedings of International Conference on Autonomous Agents and Multiagent Systems*, pages 9–16, 2001.

[36] D. Brown, R. Riolo, D. Robinson, M. North, and W. Rand. Spatial process and data models: Toward integration of agent-based models and GIS. *Journal of Geographical Systems*, 7(1):25–47, 2005.

[37] B. Burgemeestre, J. Hulstijn, and Y.-H. Tan. Towards an architecture for self-regulating agents: a case study in international trade. In *Coordination, Organizations, Institutions and Norms in Agent Systems V*, pages 320–333. Springer, 2010.

[38] D. M. Burns, J. M. Major, and T. G. Shanks. Changes in number of cigarettes smoked per day: Crosssectional and birth cohort analyses using NHIS. *Smoking and Tobacco Control Monograph*, (15):83–99, 2003. NIH publication no. 03-5370.

[39] C. Camerer. *Behavioral game theory: Experiments in strategic interaction*. Princeton University Press, 2003.

[40] P. Carbonetto, M. King, and F. Hamze. A stochastic approximation method for inference in probabilistic graphical models. In *NIPS*, volume 22, pages 216–224, 2009.

[41] K. M. Carley. On the evolution of social and organizational networks. *Research in the Sociology of Organizations*, 16:3–30, 1999.

[42] A. Casali, L. Godo, and C. Sierra. A logical framework to represent and reason about graded preferences and intentions. In *Eleventh International Conference on Principles of Knowledge Representation and Reasoning*, pages 27–37, 2008.

[43] C. Castelfranchi. *Cognitive and social action*. Routledge, 1995.

[44] C. Castelfranchi, F. Dignum, C. M. Jonker, and J. Treur. Deliberative normative agents: Principles and architecture. In *Intelligent Agents VI. Agent Theories, Architectures, and Languages*, pages 364–378. Springer, 2000.

[45] S. Cauchemez, F. Carrat, C. Viboud, A. J. Valleron, and P. Y. Bolle. A Bayesian MCMC approach to study transmission of influenza: application to household longitudinal data. *Statistics in Medicine*, 23(22):3469–3487, 2004.

[46] B. Chen and H. H. Cheng. A review of the applications of agent technology in traffic and transportation systems. *Intelligent Transportation Systems, IEEE Transactions on*, 11(2):485–497, 2010.

[47] X. Chen. Agent-based simulation of evacuation strategies under different road network structures. *University Consortium of Geographic Information Science*, 2003.

[48] X. Chen, F. Lupi, L. An, R. Sheely, A. Via, and J. Liu. Agent-based modeling of the effects of social norms on enrollment in payments for ecosystem services. *Ecological Modelling*, 229(0):16 – 24, 2012. Modeling Human Decisions.

[49] A. E. Clark. Unemployment as a social norm: Psychological evidence from panel data. *Journal of Labor Economics*, 21(2):323–351, 2003.

[50] B. Coecke. Introducing categories to the practicing physicist. In *What is category theory*, pages 45–74, 2006.

[51] M. Conner, T. Sandberg, B. McMillan, and A. Higgins. Role of anticipated regret, intentions and intention stability in adolescent smoking initiation. *British Journal of Health Psychology*, 11(1):85–101, 2006.

[52] R. Conte, G. Andrighetto, and M. Campennl. *Minding norms: Mechanisms and dynamics of social order in agent societies*. Oxford University Press, 2013.

[53] R. Conte and C. Castelfranchi. From conventions to prescriptions. towards an integrated view of norms. *Artificial intelligence and Law*, 7(4):323–340, 1999.

[54] R. Conte, C. Castelfranchi, and F. Dignum. *Autonomous norm acceptance*. Springer, 1999.

[55] N. Criado, E. Argente, and V. Botti. Normative deliberation in graded bdi agents. In *Multi-agent System Technologies*, pages 52–63. Springer, 2010.

[56] N. Criado, E. Argente, P. Noriega, and V. Botti. Human-inspired model for norm compliance decision making. *Information Sciences*, 245(0):218 – 239, 2013. Statistics with Imperfect Data.

[57] N. Criado, E. Argente, P. Noriega, and V. J. Botti. Towards a normative BDI architecture for norm compliance. In *COIN 2010*, volume 6541, pages 1–20, 2010.

[58] P. J. Dacunto, K.-C. Cheng, V. Acevedo-Bolton, N. E. Klepeis, J. L. Repace, W. R. Ott, and L. M. Hildemann. Identifying and quantifying secondhand smoke in multiunit homes with tobacco smoke odor complaints. *Atmospheric Environment*, 2013.

[59] E. Davami and G. Sukthankar. Evaluating trust-based fusion models for participatory sensing applications (extended abstract). In *Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems*, pages 1377–1378, Paris, France, may 2014.

[60] N. De Freitas, P. Højen-Sørensen, M. Jordan, and S. Russell. Variational MCMC. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 120–127. Morgan Kaufmann Publishers Inc., 2001.

[61] S. de Marchi and S. E. Page. Agent-based modeling. *Annual Review of Political Science*, 17(1), 2014.

[62] A. P. De Pinninck, C. Sierra, and M. Schorlemmer. Distributed norm enforcement via ostracism. In *Coordination, organizations, institutions, and norms in agent systems III*, pages 301–315. Springer, 2008.

[63] F. Dechesne, G. Di Tosto, V. Dignum, and F. Dignum. No smoking here: values, norms and culture in multi-agent systems. *Artificial Intelligence and Law*, 21(1):79–107, 2013.

[64] H. Dia. An agent-based approach to modelling driver route choice behaviour under the influence of real-time information. *Transportation Research Part C: Emerging Technologies*, 10(5-6):331–349, 2002.

[65] F. Dignum. Autonomous agents with norms. *Artificial Intelligence and Law*, 7(1):69–79, 1999.

[66] F. Dignum, D. Morley, E. A. Sonenberg, and L. Cavedon. Towards socially sophisticated BDI agents. In *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*, pages 111–118. IEEE, 2000.

[67] A. Dotinga, C. T. Schrijvers, A. J. Voorham, and J. P. Mackenbach. Correlates of stages of change of smoking among inhabitants of deprived neighbourhoods. *The European Journal of Public Health*, 15(2):152–159, 2005.

[68] R. I. Dunbar. Neocortex size as a constraint on group size in primates. *Journal of Human Evolution*, 22(6):469–493, 1992.

[69] R. I. Dunbar. Primates on facebook. 2009. Retrieved from: http://econ.st/1qgCrDL.

[70] N. Eagle and A. Pentland. Reality mining: sensing complex social systems. *Pervasive and Ubiquitious Computing*, 10:255–368, 2006.

[71] D. Easley and J. Kleinberg. *Networks, crowds, and markets*, volume 8. Cambridge Univ Press, 2010.

[72] D. Eaton and K. Murphy. Bayesian structure learning using dynamic programming and MCMC. In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence (UAI2007)*, pages 101–108, 2007.

[73] P. Edwards. Global climate science, uncertainty, and politics: Data-laden models, model-filtered data. *Science as Culture*, 8:437–472, 1999.

[74] S. Eilenberg and S. MacLane. General theory of natural equivalences. *Transactions of the American Mathematical Society*, 58(2):231–294, 1945.

[75] J. Eisenberg. *Group Cohesiveness*, pages pp.386–388. Sage, 2007.

[76] C. Elsenbroich and N. Gilbert. Internalisation and social norms. In *Modelling Norms*, pages 133–142. Springer, 2014.

[77] C. Elsenbroich and N. Gilbert. *Modelling Norms*. Springer Netherlands, 2014.

[78] M. S. Fagundes, H. Billhardt, and S. Ossowski. Reasoning about norm compliance with rational agents. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 1027–1028, 2010.

[79] R. W. Floyd. Algorithm 97: shortest path. *Communications of the ACM*, 5(6):345, 1962.

[80] N. Gailliard, B. Olivier, and R.-F. Audrey. A conceptual model of participatory policy making in practice: water governance and boundary workers. In *International Workshop on Agent-based Modeling for Policy Engineering (AMPLE 2012)*, European Conference on AI, pages 90–104, September 2012.

[81] A. García-Camino, J.-A. Rodríguez-Aguilar, C. Sierra, and W. Vasconcelos. Norm-oriented programming of electronic institutions: A rule-based approach. In *Coordination, Organizations, Institutions, and Norms in Agent Systems II*, pages 177–193. Springer, 2007.

[82] M. Garlick and M. Chli. The effect of social influence and curfews on civil violence. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, pages 1335–1336, 2009.

[83] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):721–741, 1984.

[84] M. Gerst, P. Wang, A. Roventini, G. Fagiolo, G. Dosi, R. Howarth, and M. Borsuk. Agent-based modeling of climate policy: An introduction to the ENGAGE multi-level model framework. *Environmental Modelling and Software*, 2012.

[85] J. W. Gibbons. *Modeling Content Lifespan in Online Social Networks Using Data Mining*. PhD thesis, University of Kansas, 2014.

[86] W. Gilks, S. Richardson, and D. Spiegelhalter. *Markov Chain Monte Carlo in practice: interdisciplinary statistics*, volume 2. Chapman & Hall/CRC, 1995.

[87] H. Gimblett. *Integrating geographic information systems and agent-based modeling techniques for simulating social and ecological processes*. Oxford University Press, USA, 2002.

[88] M. Gladwell. *The tipping point: How little things can make a big difference*. Hachette Digital, Inc., 2006.

[89] M. Granovetter. Threshold models of collective behavior. *American Journal of Sociology*, 83(6), 1978.

[90] M. Grodzins. Metropolitan segregation. *Scientific American*, 1957.

[91] A. B. Hailegiorgis, W. G. Kennedy, M. Roleau, J. Bassett, M. Coletti, G. Balan, and T. Gulden. An agent based model of climate change and conflict among pastoralists in east Africa. In *Proceedings of the International Congress on Environmental Modelling and Software*, 2010.

[92] D. Hales. Group reputation supports beneficent norms. *Journal of Artificial Societies and Social Simulation*, 5(4), 2002.

[93] D. Hammond, G. T. Fong, M. P. Zanna, J. F. Thrasher, and R. Borland. Tobacco denormalization and industry beliefs among smokers from four countries. *American Journal of Preventive Medicine*, 31(3):225–232, 2006.

[94] E. Hatfield, J. Cacioppo, and R. Rapson. *Emotional contagion*. Cambridge University Press, 1994.

[95] M. J. Healy, R. D. Olinger, R. J. Young, S. E. Taylor, T. Caudell, and K. W. Larson. Applying category theory to improve the performance of a neural architecture. *Neurocomputing*, 72(13):3158–3173, 2009.

[96] B. Heath, R. Hill, and F. Ciarallo. A survey of agent-based modeling practices (January 1998 to July 2008). *Journal of Artificial Societies and Social Simulation*, 12(4):9, 2009.

[97] F. Hinkelmann, D. Murrugarra, A. S. Jarrah, and R. Laubenbacher. A mathematical framework for agent based models of complex biological networks. *Bulletin of Mathematical Biology*, 73(7):1583–1602, 2011.

[98] C. D. Hollander and A. S. Wu. The current state of normative agent-based systems. *Journal of Artificial Societies and Social Simulation*, 14(2):6, 2011.

[99] J. B. Homer and G. B. Hirsch. System dynamics modeling for public health: background and opportunities. *American journal of public health*, 96(3):452–458, 2006.

[100] A. Ilachinski. Modelling insurgent and terrorist networks as self-organised complex adaptive systems. *International Journal of Parallel, Emergent and Distributed Systems*, 27(1):45–77, 2012.

[101] D. Jan and D. Traum. Dynamic movement and positioning of embodied agents in multiparty conversations. In *Proceedings of International Conference on Autonmous Agents and Multi-agent Systems*, 2008.

[102] X. Jin and L. Jie. A study of multi-agent based models for urban intelligent transport systems. *International Journal of Advancements in Computing Technology*, 4(6):126–134, April 2012.

[103] X. Jin and R. White. An agent-based model of the influence of neighbourhood design on daily trip patterns. *Computers, Environment and Urban Systems*, 36(5):398 – 411, 2012.

[104] A. J. Jones and M. Sergot. A formal characterisation of institutionalised power. *Logic Journal of IGPL*, 4(3):427–443, 1996.

[105] R. Jordan, M. Birkin, and A. Evans. Agent-based modelling of residential mobility, housing choice and regeneration. In *Agent-Based Models of Geographical Systems*, pages 511–524. Springer Netherlands, 2012.

[106] M. Kao, N. Fillion, and J. Bell. Jean-Pierre Marquis. From a Geometrical Point of View: A Study of the History and Philosophy of Category Theory. *Philosophia Mathematica*, 18(2):227–234, 2010.

[107] D. Kempe, J. Kleinberg, and É. Tardos. Influential nodes in a diffusion model for social networks. *Automata, Languages and Programming*, pages 1127–1138, 2005.

[108] A. Kirsch. *An introduction to the mathematical theory of inverse problems*, volume 120. Springer Science+Business Media, 2011.

[109] F. Klügl and A. L. Bazzan. Agent-based modeling and simulation. *AI Magazine*, 33(3):29, 2012.

[110] T. A. Kohler, R. K. Bocinsky, D. Cockburn, S. A. Crabtree, M. D. Varien, K. E. Kolm, S. Smith, S. G. Ortman, and Z. Kobti. Modelling prehispanic Pueblo societies in their ecosystems. *Ecological Modelling*, 241:30–41, 2012.

[111] M. M. Kokar, J. A. Tomasik, and J. Weyman. Data vs. decision fusion in the category theory framework. In *Proceedings of the International Conference on Information Fusion (FUSION)*, 2001.

[112] M. Laine, 2013. MCMC toolbox for Matlab, Finnish meteorological institute. Retrieved from: http://helios.fmi.fi/ lainema/mcmc/.

[113] D. T. Levy, L. Nikolayev, and E. Mumford. Recent trends in smoking and the role of public policies: results from the SimSmoke tobacco control policy simulation model. *Addiction*, 100(10):1526–1536, 2005.

[114] R. Liu, J. Tao, N. Shi, and X. He. Bayesian analysis of the patterns of biological susceptibility via reversible jump MCMC sampling. *Computational Statistics & Data Analysis*, 55(3):1498–1508, 2011.

[115] Y. Liu, Q. Wang, J. Liu, and T. Wark. MCMC-based indoor localization with a smart phone and sparse WiFi access points. In *IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 247–252, March 2012.

[116] K. A. Lloyd. Category theory for agent-based modeling and simulation. 2010. Obtained from: http://bit.ly/12eHkhw.

[117] F. Lopez and A. A. Marquez. An architecture for autonomous normative agents. In *Computer Science, 2004. ENC 2004. Proceedings of the Fifth Mexican International Conference in*, pages 96–103. IEEE, 2004.

[118] A. López-Paredes, D. Saurí, and J. M. Galán. Urban water management with artificial societies of agents: The FIRMABAR simulator. *Simulation*, 81(3):189–199, 2005.

[119] U. Lotzmann, M. Möhring, and K. G. Troitzsch. Simulating the emergence of norms in different scenarios. *Artificial Intelligence and Law*, 21(1):109–138, 2013.

[120] M. Luck, S. Mahmoud, F. Meneguzzi, M. Kollingbaum, T. J. Norman, N. Criado, and M. S. Fagundes. Normative agents. In S. Ossowski, editor, *Agreement Technologies*, volume 8, pages 209–220. Springer, 2013.

[121] S. X. Luo, L. S. Covey, M.-C. Hu, F. R. Levin, E. V. Nunes, and T. M. Winhusen. Toward personalized smoking-cessation treatment: Using a predictive modeling approach to guide decisions regarding stimulant medication treatment of attention-deficit/hyperactivity disorder (adhd) in smokers. *The American Journal on Addictions*, 2015.

[122] S. Mac Lane and I. Moerdijk. *Sheaves in geometry and logic: A first introduction to topos theory*. Springer Verlag, 1992.

[123] C. Macal and M. North. Tutorial on agent-based modelling and simulation. *Journal of Simulation*, 4(3):151–162, 2010.

[124] M. Maghami and G. Sukthankar. Identifying influential agents for advertising in multi-agent markets. In *Proceedings of International Conference on Autonomous Agents and Multi-Agent Systems*, pages 687–694, Valencia, Spain, June 2012.

[125] G. Malcolm. *The tipping point*. Little Brown, New York, 2000.

[126] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

[127] K. Mengersen and R. Tweedie. Rates of convergence of the Hastings and Metropolis algorithms. *The Annals of Statistics*, 24(1):101–121, 1996.

[128] S. Merler, M. Ajelli, L. Fumanelli, M. F. Gomes, A. P. y Piontti, L. Rossi, D. L. Chao, I. M. Longini, M. E. Halloran, and A. Vespignani. Spatiotemporal spread of the 2014 outbreak of ebola virus disease in Liberia and the effectiveness of non-pharmaceutical interventions: a computational modelling analysis. *The Lancet Infectious Diseases*, 2015.

[129] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1093, 1953.

[130] F. Miller, A. Vandome, and J. McBrewster. *Internalisation (Sociology)*. International Book Marketing Service Limited, 2012.

[131] M. Neumann. A classification of normative architectures. In *Simulating Interacting Agents and Social Phenomena*, pages 3–18. Springer, 2010.

[132] M. Neumann. Norm internalisation in human and artificial intelligence. *Journal of Artificial Societies and Social Simulation*, 13(1):12, 2010.

[133] M. A. Niazi, A. Hussain, and M. Kolberg. Verification & validation of agent based simulations using the VOMAS (Virtual Overlay Multi-agent System) approach. In *MAS&S at Multi-Agent Logics, Languages, and Organisations Federated Workshops (MALLOW)*, 2009.

[134] J. Oakes. Invited commentary: Rescuing Robinson Crusoe. *American Journal of Epidemiology*, 8(1):9–12, 2008.

[135] N. Oren, A. Rotolo, L. van der Torre, and S. Villata. Norms and argumentation. In S. Ossowski, editor, *Agreement Technologies*, volume 8, pages 233–249. Springer, 2013.

[136] X. Pan, C. S. Han, K. Dauber, and K. H. Law. A multi-agent based framework for the simulation of human and social behaviors during emergency evacuations. *AI & Society*, 22(2):113–132, 2007.

[137] S. Panagiotidi, J. Vzquez-Salceda, and F. Dignum. Reasoning over norm compliance via planning. In H. Aldewereld and J. Sichman, editors, *Coordination, Organizations, Institutions, and Norms in Agent Systems VIII*, volume 7756 of *Lecture Notes in Computer Science*, pages 35–52. Springer Berlin Heidelberg, 2013.

[138] D. Pepper. *How to Start and Grow Your Lawn Care Maintenance Business*. Lulu Enterprises Incorporated, 2008.

[139] H. Perkins and A. Berkowitz. Perceiving the community norms of alcohol use among students: Some research implications for campus alcohol education programming. *International Journal of the Addictions*, 21:961–976, 1986.

[140] W. Piper, M. Marrache, R. Lacroix, A. Richardson, and B. Jones. Cohesion as a basic bond in groups. *Human Relations*, 36, 1983.

[141] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 2007.

[142] F. Ramtin, M. Abolhasanpour, H. Hojabri, A. Hemmati, and A. Jaafari. Optimal multi floor facility layout. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, volume 3. Citeseer, 2010.

[143] F. Ramtin and J. A. Pazour. Analytical models for an automated storage and retrieval system with multiple in-the-aisle pick positions. *IIE Transactions*, 46(9):968–986, 2014.

[144] W. Rand and R. T. Rust. Agent-based modeling in marketing: Guidelines for rigor. *International Journal of Research in Marketing*, 28(3):181–193, 2011.

[145] J. C. Reynolds. Using category theory to design programming languages. In *Programming Languages and Systems*, pages 62–63. Springer, 2009.

[146] F. Sadri, K. Stathis, and F. Toni. Normative kgp agents. *Computational & Mathematical Organization Theory*, 12(2-3):101–126, 2006.

[147] D. L. Sallach. In press work, http://1.usa.gov/13C2FlA.

[148] B. T. R. Savarimuthu and S. Cranefield. Norm creation, spreading and emergence: A survey of simulation models of norms in multi-agent systems. *Multiagent and Grid Systems*, 7(1):21–54, 2011.

[149] B. T. R. Savarimuthu, S. Cranefield, M. K. Purvis, and M. A. Purvis. Norm emergence in agent societies formed by dynamically changing networks. *Web Intelligence and Agent Systems*, 7(3):223–232, 2009.

[150] B. T. R. Savarimuthu, M. Purvis, M. Purvis, and S. Cranefield. Social norm emergence in virtual agent societies. In *Declarative Agent Languages and Technologies VI*, pages 18–28. Springer, 2009.

[151] T. C. Schelling. *Micromotives and macrobehavior*. WW Norton & Company, 2006.

[152] P. W. Schultz, R. J. Bator, L. B. Large, C. M. Bruni, and J. J. Tabanico. Littering in context personal and environmental predictors of littering behavior. *Environment and Behavior*, 45(1):35–59, 2013.

[153] S. H. Schwartz. A theory of cultural value orientations: Explication and applications. *International Studies in Sociology and Social Anthropology*, 104:33, 2006.

[154] O. Sen and S. Sen. Effects of social network topology and options on norm emergence. In *Coordination, Organizations, Institutions and Norms in Agent Systems V*, pages 211–222. Springer, 2010.

[155] S. Sen and S. Airiau. Emergence of norms through social learning. In *Proceedings of the International Joint Conference on Artifical Intelligence*, pages 1507–1512, 2007.

[156] C. S. Sripada and S. Stich. A framework for the psychology of norms. *The Innate Mind: Culture and Cognition*, pages 280–301, 2005.

[157] R. E. Streit and D. Borenstein. An agent-based simulation model for analyzing the governance of the Brazilian financial system. *Expert Systems with Applications*, 36(9):11489–11501, 2009.

[158] S. Swarup, A. Apolloni, and Z. Fagyal. A model of norm emergence and innovation in language change. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 693–700. International Foundation for Autonomous Agents and Multiagent Systems, 2011.

[159] M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:83–124, 1997.

[160] R. D. C. Team. R: A language and environment for statistical computing, 2012. Vienna, Austria: R Foundation for Stat istical Computing.

[161] K. P. Timms, D. E. Rivera, L. M. Collins, and M. E. Piper. System identification modeling of a smoking cessation intervention. In *16th IFAC Symposium on System Identification, Brussels, Belgium*, pages 11–13, 2012.

[162] J. Verella and A. Wardak. Modeling public opinion and voting as a complex system with agent-based simulations. In *IEEE Systems and Information Engineering Design Symposium*, pages 261–266, April 2008.

[163] D. Villatoro, G. Andrighetto, J. Sabater-Mir, and R. Conte. Dynamic sanctioning for robust and cost-efficient norm compliance. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume One*, pages 414–419. AAAI Press, 2011.

[164] D. Villatoro, S. Sen, and J. Sabater-Mir. Topology and memory effect on convention emergence. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 02*, pages 233–240. IEEE Computer Society, 2009.

[165] X. Wang, M. Maghami, and G. Sukthankar. Leveraging network properties for trust evaluation in multi-agent systems. In *Proceedings of the IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 02*, pages 288–295. IEEE Computer Society, 2011.

[166] M. D. Ward and K. S. Gleditsch. Location, location, location: An MCMC approach to modeling the spatial context of war and peace. *Political Analysis*, 10(3):244–260, 2002.

[167] D. J. Watts. A simple model of fads and cascading failures on sparse switching networks. In *Economics with Heterogeneous Interacting Agents*, pages 13–25. Springer, 2001.

[168] U. Wilensky, 1999. NetLogo. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University. Retrieved from: http://ccl.northwestern.edu/netlogo/.

[169] U. Wilensky and W. Rand. Making models match: Replicating an agent-based model. *Journal of Artificial Societies and Social Simulation*, 10(4):2, 2007.

[170] J. Q. Wilson and G. L. Kelling. Broken windows. *Atlantic Monthly*, 249(3):29–38, 1982.

[171] H. P. Young. The economics of convention. *The Journal of Economic Perspectives*, pages 105–122, 1996.

[172] C. Yu, M. Zhang, F. Ren, and X. Luo. Emergence of social norms through collective learning in networked agent societies. In *Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems*, pages 475–482, 2013.