

**AGENT-BASED SIMULATION TUTORIAL -
SIMULATION OF EMERGENT BEHAVIOR AND DIFFERENCES BETWEEN AGENT-BASED
SIMULATION AND DISCRETE-EVENT SIMULATION**

Wai Kin Victor Chan

Young-Jun Son

Rensselaer Polytechnic Institute
Department of Industrial and Systems Engineering
Troy, NY 12180, USA

The University of Arizona
Department of Systems and Industrial Engineering
Tucson, AZ 85721, USA

Charles M. Macal

Argonne National Laboratory
Center for Complex Adaptive Systems Simulation, Decision & Information Sciences Division
Argonne, IL 60439 USA

ABSTRACT

This tutorial demonstrates the use of agent-based simulation (ABS) in modeling emergent behaviors. We first introduce key concepts of ABS by using two simple examples: the Game of Life and the Boids models. We illustrate agent-based modeling issues and simulation of emergent behaviors by using examples in social networks, auction-type markets, emergency evacuation, crowd behavior under normal situations, biology, material science, chemistry, and archaeology. Finally, we discuss the relationship between ABS and other simulation methodologies and outline some research challenges in ABS.

1 INTRODUCTION

Agent-based simulation (ABS) is a rather new approach for simulating systems with interacting autonomous agents. While the precise definition of ABS varies across fields (or even in the same field), the philosophy and usages of ABS are similar—simulating interactions of autonomous objects (called agents) to identify, explain, generate, and design emergent behaviors.

This tutorial focuses on illustrating the use of ABS in modeling emergent behaviors. A great amount of work has been done in ABS over the past decade. Accompanying the development of ABS are introductory tutorials and surveys of ABS, which have further broadened the impact of ABS in various fields (Heath, Hill, and Ciarallo 2009; Macal and North 2005, 2006, 2009; Railsback, Lytinen, and Jackson 2006; Wooldridge and Jennings 1995; Zhou, Chan, and Chow 2009). The present tutorial complements existing ones by providing examples to demonstrate the use of ABS in modeling emergent behaviors. To be self-contained, this tutorial will also introduce basic concepts of ABS by using two well-known ABS models—the Game of Life model and the Boids model. To make the two models more interesting, we will extend them to demonstrate how a small variation in rules leads to new emergent behaviors, thus showing the capability of ABS in obtaining known and unknown emergent behaviors. Other examples to be discussed in this paper are listed in Table 1. In addition to these examples, we also highlight several representative studies in biology, material science, chemistry, and archaeology.

Table 1: List of ABS Examples

Model	Purposes
Game of Life	Introduce concepts of agents, interaction rules; Show how simple rules lead to various emergent patterns; Draw a connection between ABS and cellular automata.
Boids	Show how simple rules lead to flocking behavior; Demonstrate the effects of different rules on flocking behavior.
Social networks	Show how real-world data is injected into a model to drive agents' behavior; Demonstrate how a model is scaled up by means of ABS.
Electricity markets	Illustrate the incorporation of learning algorithms into agent decision making; Show how a simple learning algorithm leads to collaborative and competing behavior.
Emergency evacuation	Illustrate how to enhance agent cognition by incorporating the extended Belief-Desire-Intention framework and conducting human-in-the-loop experiments in the CAVE.
Pedestrian behavior at a mall	Illustrate how to integrate tactical human decision-making (via the Extended Field Theory (EFT)) and physical interactions and congestions (via the Social Force model) at a shopping mall.

A brief introduction to ABS should be provided before introducing these examples. We first comment on the relationship between ABS and other simulation approaches (a more detailed discussion is deferred to Section 3). Whether an ABS model is a discrete-event model or a hybrid of discrete and continuous depends on its state variables. If all state variables change only at discrete time instants, it is a discrete-event model. If in addition to discrete-time state variables it also contains state variables that vary in time, then it is a hybrid model. For example, a power grid ABS model can have continuous state variables representing the states of generators (e.g., phase angle) and transmission lines and discrete state variables modeling the decisions of generation companies.

ABS is, however, different from classic discrete-event simulation due to the nature of agents, which are proactive, autonomous, and intelligent. This leads to the following definition of an ABS model: Generally speaking, an ABS model is *a hybrid discrete-continuous simulation model with proactive, autonomous, and intelligent entities*. Proactiveness refers to the ability of taking initiative. Agents can initiate actions, communicate with other agents, and make decisions on their own. In contrast, entities in a discrete-event simulation model are rather simple, reactive, and limited in capabilities. Entities in most discrete-event simulations rely on some central mechanism—e.g., the event scheduling function—to invoke actions to change entity states, and entities have no learning or cognitive reasoning. For example, consider a queueing simulation model in which the entities in a queue determine whether to stay or leave by communicating and gathering waiting time information from nearby entities. It is not straightforward to model this system by using discrete-event simulation, which is demonstrated in (Chan 2010). It is rather easy to model this system by using the ABS approach.

One caveat is that agents' proactiveness diminishes when the model is executed in a single-CPU computer. The reason is that the update of all agents' states must be performed one agent-by-one agent (i.e., pooling)—because there is only one CPU. This one-by-one execution creates misconception that agents are reactive and similar to entities in a discrete-event simulation; But in fact, they are quite different. Full advantage of the proactiveness of the ABS framework can be achieved by using a parallel-CPU system or a network of computers.

The fields of *artificial intelligence* and *social science* embrace a more humanized notion (e.g., behavioral or emotional) in the description of agents. Agents can perceive, reason, react, memorize, and take initiatives based on their knowledge, past experience, and pre-defined rules—all according to the modeler's specification (for a complete list of properties of agents, see (Wooldridge and Jennings 1995)). In this sense, an agent can be thought of as an avatar of a human or a representation of an intelligent creature

or object. Indeed, agents are computer objects programmed to perform pre-defined actions. In many cases, agents, while operating based on their own strategies, cooperate or compete with each other in accordance with certain mutual rules or agreements. The behavioral complexity of an agent is quite flexible and ranges from simple binary decisions (yes or no) to complicated human behavior or intelligence.

The virtual space (environment) with which the agents interact can be in the form of ring, lattice, or random networks, 2D or 3D space, or based on a map such as GIS (Geographic Information System) map. Agents can move freely in the environment. This makes it applicable for modeling and visualizing complex behaviors in physical systems, such as simulations of traffic, evacuations, biological systems, mechanical systems, and so forth.

However, what makes ABS powerful is not the intelligence or rich notions of the agents; it is the simulation of the interactions of these avatars that creates opportunities for people to better understand their nature. ABS allows us to simulate cascading effects arising from possibly minor local interactions, experimentally examine tipping points, identify and explain beneficial or malicious emergent behaviors, and more importantly design mechanisms to grow beneficial (and discourage malicious) behaviors. All of these are facilitated by the advantages of simulation, which is low cost and repeatable. An ABS can be driven by real data and can sweep through the parameter space (given enough time, power, and memory) to answer what-if questions effectively.

Indeed, many existing ABS models are actually quite simple with no complicated agent architecture or sophisticated interaction rules. Yet, these simple models are able to produce various emergent behaviors owing to the modeling power generated through the interactions of a large number of simple agents (see examples in Section 2).

Emergent behavior is commonly defined as macroscopic coherent regularities, such as identifiable distributions, coherent patterns (spatial, temporal, or behavioral), equilibria, and so forth. While stochastic and chaotic patterns are irregular, so long as they are of interest to modelers or users, one can use ABS to simulate them and design rules to generate them. Therefore, in this paper we define emergent behavior as manifested discernible regular formations and interesting irregular phenomena of actions or patterns; they can be static or dynamic, deterministic or stochastic, or ordered, chaotic, or in-between ordered and chaotic so long as they are of interest to modelers or users. They can also emerge from non-coordinated interactions or competitions.

The popularity of ABS is also driven by its capability of handling the increasing complexity of real-world systems, which often contain a large number of interacting agents that are autonomous, goal-driven, and adaptive. Consider the following example: A large-scale healthcare network consists of a number of units from large facilities such as hospitals, medical supply warehouses, and blood banks to small service units such as ambulances, operating rooms, and personnel such as doctors and nurses to patients and other service providers. All these units can be considered as autonomous agents. For instance, warehouses react to demands by providing inventory based on certain ordering rules, ambulances operate in accordance with some dispatch policies constrained by time and space conditions, and patients try to intelligently select doctors. It is natural to model these intelligent units as computer agents and simulate their interactions.

ABS has been widely used in various fields, including military, biology, social science, economics, and business. The theoretical basis of ABS lies mainly in *cellular automata* (Beyer, Sellers, and Waterman 1985; Von Neumann 1966; Wolfram 1984), *complex system modeling*, *artificial life*, and *swarm intelligence* (Bonabeau, Dorigo, and Theraulaz 1999). However, the ABS community has grown very large, and research in ABS has incorporated essential ideas from these and many other fields, rendering it a multidisciplinary science integrating computer science, cognitive and social sciences, and simulation.

The algorithm for executing an ABS model varies greatly in different ABS platforms. A general scheme is outlined below. This algorithm has a high computational requirement because it loops through all agents in each time step. Methods have been proposed to avoid looping through all agents, for example, only updating those agents whose state changes matter. Another approach is to employ the discrete-

event simulation algorithm to advance the clock based on events and only update the continuous variables in every time step (Page, Knaak, and Kruse 2007). Different hardware (such as parallel CPUs or graphics processing units—GPUs) have also been used to improve the efficiency of ABS models. It has been shown that execution speed can be improved by thousands of times when using GPUs for some cases (Lysenko and D'Souza 2008).

Agent-Based Simulation Execution Algorithm:

- Initialize
- Do until stop condition is satisfied
 - Update all continuous state variables
 - For each agent (select an agent randomly or based on a specified order)
 - Perform actions: change state, send or fetch messages to (or from) other agents or the environment, etc.
 - Advance clock by a given amount of time (e.g., a tick count)

2 ILLUSTRATIVE EXAMPLES

2.1 Conway's Game of Life (Emergent Patterns)

The first example is Conway's Game of Life model (Berlekamp, Conway, and Guy 2003). Imagine that there is a two-dimensional grid of size $n \times n$ where n is the number of rows (or columns) in the grid. A cell lives in each entry of the grid. The cells cannot move. Each cell has only two states: alive and dead. Each cell changes its state based on the following three simple rules that describe the interactions between a cell and its eight neighbors (up, down, left, right, and four diagonal cells). The left panel of Figure 1 shows the initial state of the simulation. A red dot represents a live cell. Dead cells are uncolored and therefore, appear as white in the figure. A snapshot of the model after one hundred iterations is given in the right panel of Figure 1, which shows some clear patterns (one iteration is one execution of the outer loop of the ABS algorithm). See (Rendell 2002) for an extensive discussion of various patterns.

1. A live cell with exactly 2 or 3 live neighbors will remain alive in the next time step.
2. A dead cell with exactly 3 live neighbors will come to life in the next time step.
3. Otherwise, the cell will die either of loneliness or overcrowding in the next time step.

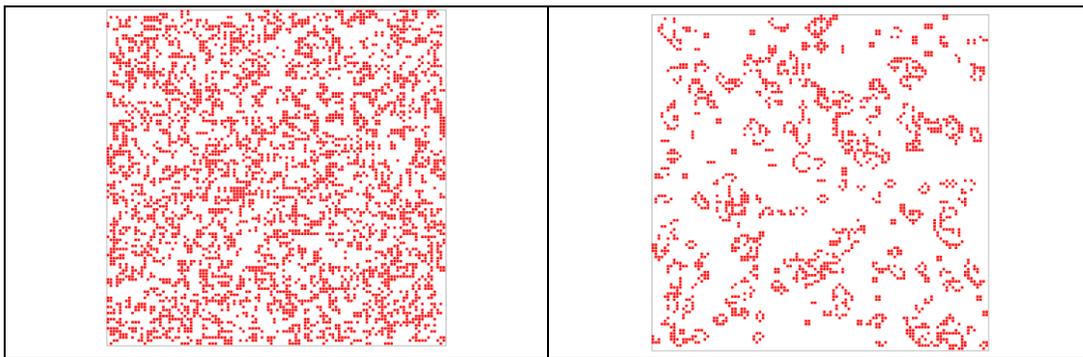


Figure 1: Agent-based Simulation Model of Conway's Game Of Life

The above example is quite simple because the agents have very limited capabilities and no spatial movement is allowed. This model is in fact a cellular automaton. A cellular automaton (CA) model considers a set of cells residing in a two-dimensional lattice. It produces complex behaviors emergent from some simple rules that govern the interactions of the cells. While this example is simple, it is has univer-

sal computational power (Rendell 2002). In addition, we note that all the cells update their states simultaneously at each iteration. CAs that use this kind of updating rules are called *synchronous CA*. On the other hand, CAs that update cells at a particular order are called *asynchronous CA*. If a synchronous CA is run in a single-CPU computer, an updating order should be specified (e.g., random order) if the order of updating matters to the outcome. This is because the cells must be updated one-by-one and cells updated later cannot affect the states of cells updated earlier in the same time step. For the example above, if a cell changes its state based on the states of its neighbors at the previous time step, then the order of updating does not matter. Otherwise, if the cell's state depends on its neighbors' states at the current time step, then the order of updating matters. Other situations in which the order of updating matters include cell competition and resource allocation. Although CA is known to be Turing complete, ABS allows agents to have a greater degree of flexibility and functionality (such as learning and adaptation) and can accommodate more complicated interaction rules.

The flexibility of this model in arriving at different emergent behaviors is demonstrated in (Chan 2010). There, the Game of Life model is generalized by simply changing the number of live neighbors in Rules #1 and 2 to X , Y , and Z , which are defined below. The emergent patterns are visualized by coloring the cells based on the number of live neighbors (see the legend of Figure 2).

Variation of John Conway's Game of Life Model:

1. A live cell with at least X and at most Y live neighbors will remain alive in the next time step.
 2. A dead cell with exactly Z live neighbors will come to life in the next time step.
 3. Otherwise, the cell will die either of loneliness or crowdedness in the next time step.
- where $0 \leq X, Y, Z \leq 8$ and $X < Y$.

This variation has a total of 405 possible combinations of rules, (X, Y, Z) . Some of them are trivial but some produce clear emergent patterns that evolve in time. One can also use a series of rules to obtain different combinations of patterns. Figure 2 shows the patterns obtained by first initiating all cells to be alive and then applying this series of rules: $(0, 7, 8) \rightarrow (0, 7, 3) \rightarrow (0, 7, 6) \rightarrow (0, 7, 5) \rightarrow (0, 7, 4) \rightarrow (0, 5, 4) \rightarrow (3, 5, 4)$.

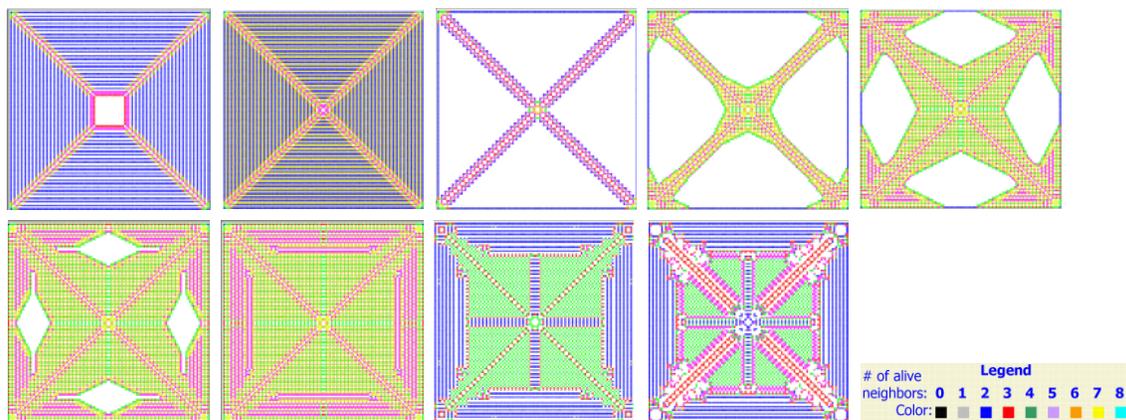


Figure 2: Emergent Patterns of Variation of Conway's Game Of Life Model

2.2 Boids Model (Flocking, Schooling, Herding)

In this section, we consider the famous Boids model introduced by (Reynolds 1987). This model simulates the flocking behavior of birds (or schooling behavior of fishes) emergent from leaderless interactions among self-propelled agents (i.e., fishes or birds). The emergent flocking behavior is also similar to grouping behavior of other creatures, such as schooling behavior in chub mackerel, *Scomber japonicus*, and herding behavior in animals and crowds.

In the Boids model, each agent follows three simple rules:

1. Separation: steer to avoid crowding local flockmates
2. Alignment: steer towards the average heading of local flockmates
3. Cohesion: steer towards the average position of local flockmates

The pseudocode of the model is given in the left panel of Figure 3. Implementing the code in NetLogo (Wilensky 1999) and executing it results in the emergent flocking behavior shown in Figure 3(b).

Besides having simple rules, this model exhibits a somewhat counter-intuitive result. Intuitively, the alignment rule should have nothing to do with cohering the birds together. However, this rule after being "transformed" by the interactions among the boids becomes a mechanism that binds the birds together. This can be seen in Figure 3(c), where only the separation and alignment rules (without cohesion rule) are enforced. On the other hand, the cohesion rule does not seem to cohere the agents (but makes the birds circle) when applied alone as shown in Figure 3(d), where only the separation and cohesion rules (without alignment rule) are used. Flocking behavior is not observed. The alignment rule has a more global effect on "cohering" the Boids into big schools.

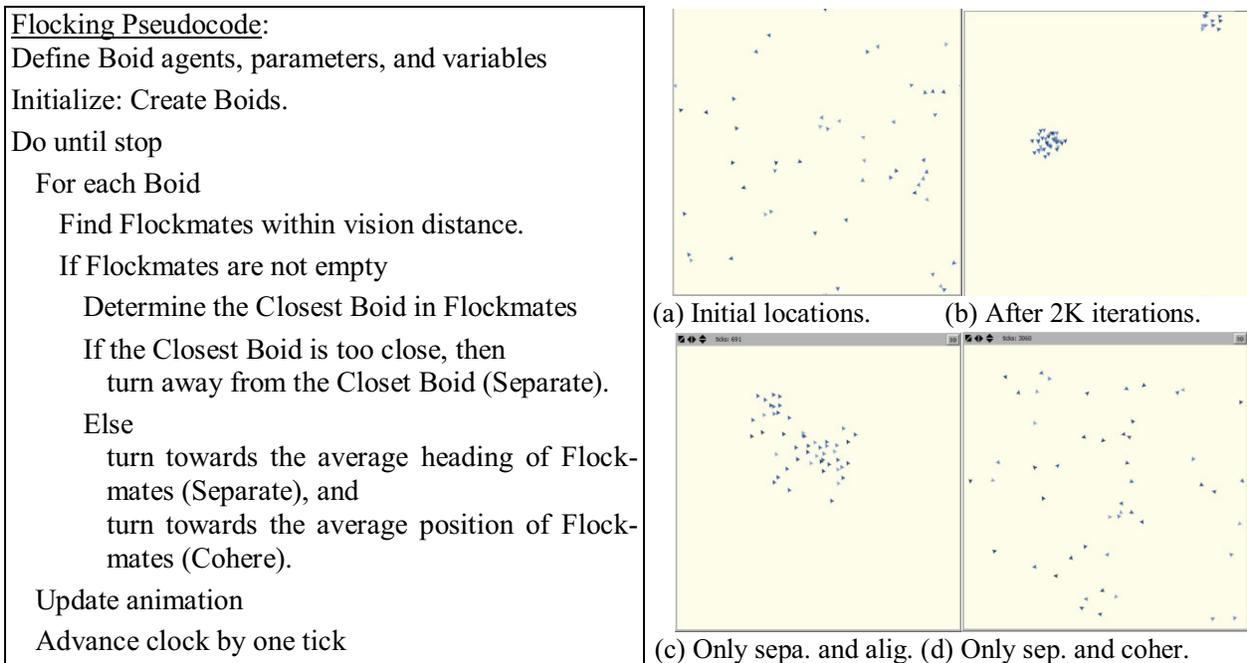


Figure 3. Emergent Flocking Behaviors.

2.3 Social Networks (Incorporating Data and Scaling Up)

In an ABS social network model, the agents are individuals (e.g., actors or participants). The network layout (i.e., locations of agents) can be a fixed layout (such as a ring or lattice), a random layout (i.e., distributed randomly in the environment), or a user-defined layout.

We take the model from (Chi et al. 2009) to show the practice of incorporating data and scaling up an ABS model. There, real-world communication data in a network of 30 students was obtained. This dataset includes messages exchanged between the students who participated in an online course. Despite being at a high-level of detail (with full content and attributes of each message), this dataset is limited in size. ABS was used to mitigate the limitation of using this dataset in influencing properties of large-scale networks.

The approach is to create a large network composed of a number of groups, each following a communication behavior similar to the real data. The details are as follows. First, various communication parameters, distributions, and correlations were estimated from the data. Approaches for handling no or few

data as suggested in (Law 2006) are employed. Specifically, common real-world network properties such as the small-world phenomenon and power-law distribution (i.e., preferential attachment) were employed in the agents' connectivity and communication decisions. Second, a small group of agents (such as 30) is created. These agents follow similar (but with random variation) communication behavior (i.e., parameters, distributions, etc.) as the real-world students. Then, this group of agents is duplicated as many times as needed (e.g., 100). There are random variations in the duplicates, but overall they still have the original communication behavior. Finally, the communication behavior across these groups is also set to follow the original communication behavior. This way, the network size is scaled up to 100 times with the original communication behavior preserved.

This example shows how data from a relatively small network can be used to drive a large-scale ABS model to provide insights about network behaviors of large networks composed of subnetworks that have similar properties. Figure 4 gives a snapshot of the simulation model with 50 groups, each arranged in a ring. One can also alter the communication behavior across groups or within groups, or sweep through the parameter space to examine how different communication behaviors influence network properties. Indeed, many real-world large-scale social networks are made up of many small socially or geographically-based subnetworks. For example, Facebook consisted of 55K subnetworks as of 2007 and surprisingly, 620M subnetworks as of 2010 (O'Neill 2010).

Conventional social network simulations are mostly Monte Carlo. In contrast, ABS of social networks allows decision making (autonomy) and learning by each participant. Another benefit of ABS over Monte Carlo simulation is its ease of visualization, which is also a key factor that makes ABS a popular approach in various fields.

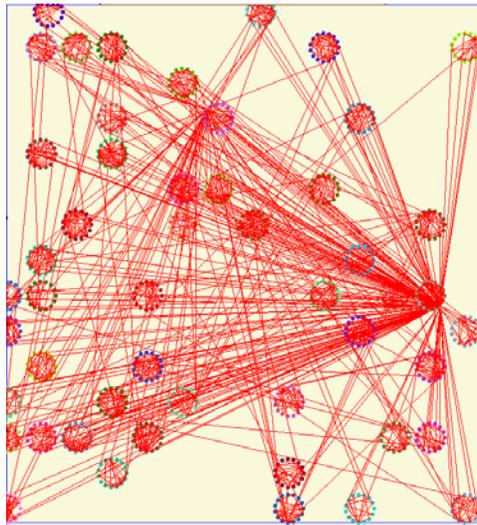


Figure 4: Snapshot of ABS Social Network Simulation.

2.4 Electricity Markets (Learning to Compete and Cooperate)

In this section, we provide an ABS example that incorporates learning capability into agents' decision making. This is an example of an electricity market. An electricity market is a complex system. It involves various types of intelligent market players (participants), including generation companies, transmission companies, utility companies (i.e., distribution or customer companies), consumers, and independent system operator (or regional transmission operator). Besides these human participants, a detailed simulation of an electricity market should also include simulation modules of the underlying physical transmission network (transmission lines and nodes) and other mechanisms that mimic situations encountered in real operations, for example, random events that disrupt the operation, and connections with other

energy markets, such as the CO₂ market, as well as extensions to the smart grid. All these components can be modeled as agents governed by certain physical laws or man-made regulations.

We take the example from (Zhou 2010) to illustrate how an agent-based approach is used to model a simple auction-type power market and to obtain collusive and competitive bidding behaviors. This example is a five-node power system shown in Figure 5(a). Each node contains multiple generators and a load-serving entity representing the load on the corresponding node. They are the agents in the model. In addition, there is an Independent System Operator (ISO) agent that is not shown in the figure. The market considered in this example is the day-ahead market.

Here, we focus on Zone 4, which is a zone with high demand and high production cost. There is a transmission limit on the amount of electricity imported to Zone 4. Situated at Zone 4 are two identical giant generator agents (G_1 and G_2) and four other small generator agents. The two giant generator agents, G_1 and G_2 , are strategic bidders and have equal generation capacity while the four small generator agents are simply marginal-cost bidders with no adaptation capability. The maximum total supply provided by the four small generator agents cannot meet the entire market demand, but G_1 (or G_2) alone can fulfill the entire demand. In other words, G_1 and G_2 have an equal market power much larger than the four marginal-cost bidder agents combined. This not only allows G_1 and G_2 to determine the market clearing price but also creates gaming opportunities for them to drive up or down the price as they bid to compete with each other. There is no competition between the small agents and the large agents because the four small agents always get dispatched due to their low bids. As a result, the two large agents, G_1 and G_2 , are competing for the residual demand, which is the total demand less the portion fulfilled by the four small generator agents. In addition, G_1 and G_2 , being identical, should equally share this residual market (i.e., 50% vs. 50%) if there is no manipulation of the market.

Both G_1 and G_2 have two objectives: maximize profit and reach the target market share. These two objectives are sometimes conflicting. For example, one agent can lower its bid (thus may lose profit) just to gain more market share. The learning algorithm used by the two agents to bid strategically is a simple online hill climbing algorithm. Specifically, there are three rules:

1. If the profit increases, then add a positive increment on the next bid. Otherwise, apply a negative increment.
2. If the market share decreases, then increase the amount of increment if the profit also decreases or decrease the amount of increment if the profit increases. Otherwise, make no adjustment if the market share increases.
3. If the current market share is less than the target market share, then increase the amount of increment if the profit decreases or decrease the amount of increment if the profit increases. Otherwise, make no adjustment if the current market share is larger than the target market share.

The two adjustments in Rules 2 and 3 are independent of each other and can be applied individually or simultaneously. By using this simple learning algorithm and setting a correct target market share of 50%, the two agents can learn to bid collusively to maximize their total profit and reach their target market share without coordination or knowledge of each other's information.

The results are more interesting when G_1 and G_2 are allowed to maliciously set a target market share higher or lower than their true market power (or simply mis-estimate their market power). Figure 5(b) shows the average profit of G_2 obtained under different target market share combinations between G_1 and G_2 . The figure for G_1 can be obtained by considering it symmetric to Figure 5(b). It can be seen that when both agents are conservative (i.e., target share less than 50%), both agents receive a relatively high profit. The worst case is a price war with zero profit, which occurs when both agents are aggressive (i.e., target share larger than 50%). If one agent is aggressive ($> 50\%$) and the other one is conservative ($< 50\%$), then the aggressive one wins. In addition, the higher its aggressiveness, the higher the profit, provided that the other agent is conservative. This result in fact mimics a Chicken game (Rapoport and Chammah 1966), which is common in many real-world competitions—the winner is the braver, but when both are brave,

both lose. The lessons learned from this example are that being accurate in estimating your market power is good and being aggressive is only good if your opponent is conservative.

Finally, this ABS model is different from conventional discrete-event, continuous, or Monte-Carlo simulation models or pure object-oriented programs in that it is a hybrid discrete-continuous simulation model with both discrete (e.g., decisions) and continuous (e.g., states of generators and transmission lines) state variables and factors that are stochastic (random number), dynamic (time), and adaptive (learning). ABS allows us to integrate different simulation modeling techniques and computer programming approaches to form a holistic methodology to tackle challenging problems and complex real-world systems such as the modeling and analysis of a smart grid.

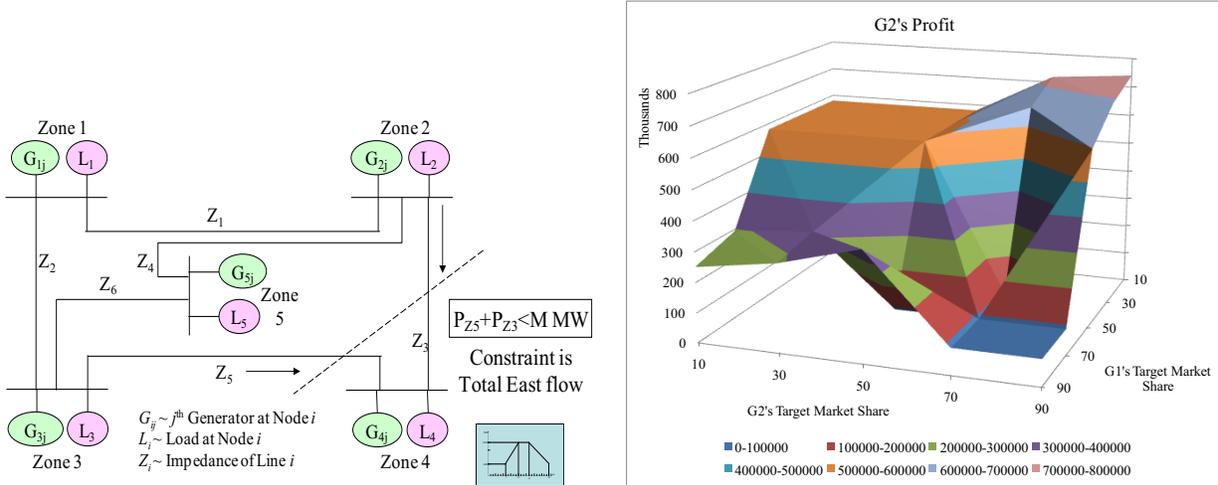


Figure 5: (a) Schematic of a Five-Zone Agent-based Power Simulator; (b) Average Profit of G2 Obtained under Different Target market Share Combinations.

2.5 Emergency Evacuation Simulation based on Integrated BDI Framework

This section describes an integrated human decision making model for evacuation scenarios under a Belief-Desire-Intention framework (see Figure 6(a)) based on work by (Lee and Son 2008) and (Lee, Son, and Jin 2010). The submodules of the integrated BDI framework (Lee and Son 2008) are based on a Bayesian belief network (BBN), Decision-Field-Theory (DFT), and a probabilistic depth first search (PDFS) technique. To mimic realistic human behaviors, attributes of the BDI framework have been reverse-engineered from human-in-the-loop experiments conducted in the Cave Automatic Virtual Environment (CAVE) available at The University of Arizona.

The modeling framework proposed by (Lee and Son 2008) has been demonstrated for a person's evacuation behaviors in response to a terrorist bomb attack in the National Mall area in Washington, D.C. Given the scenario, different types of agents were characterized based on 1) familiarity with the area, 2) risk-taking behavior, 3) confidence index (affecting the moving speed of an agent and leader/follower behavior), and 4) guidance by police. The scenario begins when an explosion occurs, the police are informed of it via radio transmission, and the police ask people around them to evacuate the area. Agents' evacuation behaviors vary according to their characteristics. For example, those who are familiar with the area (commuters) invoke the multi-horizon planning algorithm to develop their evacuation plan. On the other hand, those who are not familiar with the area (novice agents) move from intersection to intersection (i.e., invoke the single-horizon planning algorithm) and may be guided by a police officer or a commuter agent to the nearest exit. Once the agents reach an exit point, they are discarded from the simulation.

To develop the above mentioned emergency evacuation simulation, a human-in-the-loop experiment was conducted, where each subject was asked to evaluate the risk and the evacuation time of three availa-

ble paths (i.e., right, forward, and left) depending on the various environmental observations (i.e., fire, smoke, police, and crowd) at each intersection. Also, each subject was asked to select one of the three available paths. The data collected on the relationship between the environment and the subject’s evaluation was used to construct a BBN in the form of a conditional probability distribution. The constructed BBN infers 1) subjective evaluations for each attribute (e.g., risk and time) of each given option and 2) subjective weights of attention corresponding to each attribute, and the DFT calculates preference values of the options based on those matrices of evaluations and weights.

The simulated environment and agents conforming to the proposed BDI framework have been implemented in AnyLogic® software, where each agent calls external Netica BBN software to perform its perceptual processing function and Soar software to perform its real-time planning and decision-execution functions (see Figure 6(b)). The constructed simulation has been used to test the impact of several factors (e.g., demographics, number of police officers, information sharing via speakers) on evacuation performance (e.g., average evacuation time, percentage of casualties).

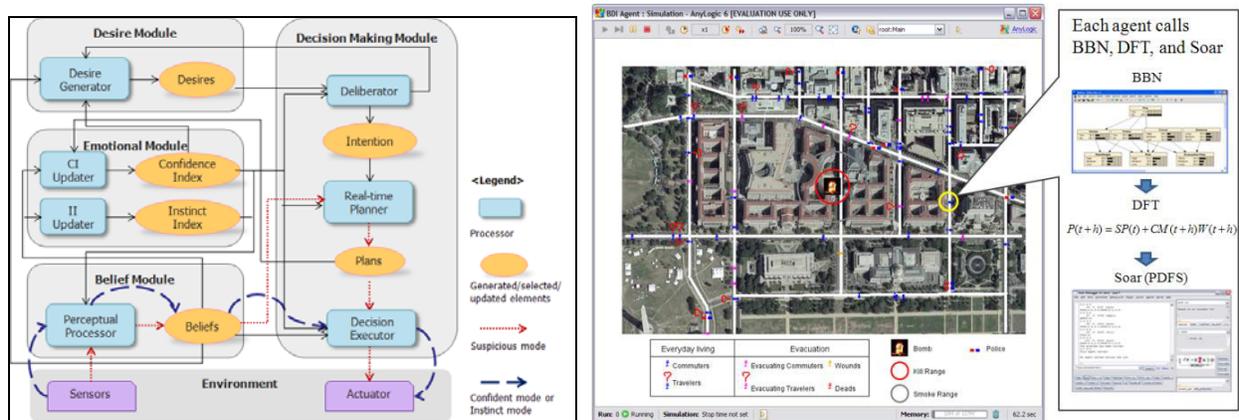


Figure 6: (a) Components of extended BDI framework (Lee and Son 2008); (b) Snapshot of emergency evacuation simulation interacting with BBN, DFT and Soar (Lee and Son 2008).

2.6 An Integrated Pedestrian Behavior Model Based on Extended DFT and Social Force Model

While Section 2.5 focuses on human decision-making under an emergency evacuation scenario, this section focuses on integration of human decision-making with physical interactions and congestion under a normal situation at a shopping mall based on the work by (Xi and Son 2010). While physical interactions and congestion among the people and the environment may not be a major concern in a spacious area (e.g. street), they become a significant factor in congested areas (e.g. building).

The pedestrian behavior model proposed by (Xi and Son 2010) integrates 1) Extended Decision Field Theory (EDFT) for tactical level human decision-making and 2) Social Force model (SFM) to represent physical interactions and congestions among the people and the environment. First, decisions on selecting from alternatives (e.g. destinations and movement directions) are made based on EDFT (Lee, Son, and Jin 2008). The original DFT is a psychology-based model and has been widely used for mimicking human deliberation processes in making decisions under uncertainty (Busemeyer and Townsend 1993; Lee, Son, and Jin 2008) extended it to cope with a dynamically changing environment. In the shopping mall scenario considered by (Xi and Son 2010), the environmental conditions (e.g. crowd density and destinations) change dynamically for individuals, and the EDFT in the pedestrian model mimics pedestrians’ deliberation on direction changes. Second, the Social Force Model introduced by (Helbing, Farkas, and Vicsek 2000) has been enhanced with the vision of each individual, and both individual behaviors as well as group behaviors are considered. In the original SFM, pedestrian motives and their reactions to a crowd are represented by a combination of physical and psychological forces, which are translated into the acceleration equation. (Xi and Son 2010) extended the original SFM with two aspects. First, the original SFM

computes a force impact between every pair of agents in the environment. In other words, there will be a force even between agents who are significantly far away from each other, which is not realistic. Second, the social force between agents is always positive implying that all agents are psychologically repulsed by each other. However, this is not the case for friends or family members, who usually stay close to each other.

The proposed model is illustrated and demonstrated with a shopping mall scenario as it provides us with various environmental conditions (e.g. different kind of shops, obstacles, promotions on the shops) and populations variations (e.g. gender, age, preference, schedule, and grouping). Consideration of a rich set of attributes for the environment as well as people will allow us to mimic a real shopping mall environment closely. A survey and observations have been conducted at the mall for data collection and partial validation of the proposed model. A simulation model of the considered shopping mall was then constructed using AnyLogic® software, where each individual in the simulation executes 1) a planning algorithm to select a destination, 2) EDFT for choosing a direction, and 3) the extended SFM to adjust its velocity (see Figure 7(a)). Using the constructed crowd simulation model (see Figure 7(b)), several experiments have been conducted to test the impact of various factors (e.g. consideration of human’s vision, group shopping behavior, arrangement of stores, complexity of the model) on several metrics such as 1) the average distance among neighboring shoppers, 2) the movement speed of pedestrians, 3) the profit of the shopping mall, and 4) scalability.

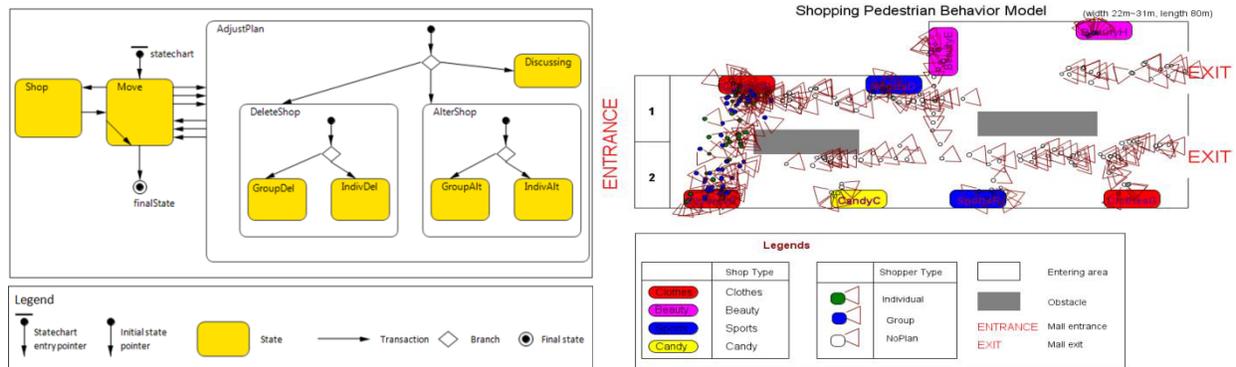


Figure 7: (a) State charts for the shopper’s behavior (Xi and Son 2010); (b) Snapshot of a shopping mall simulation (Xi and Son 2010).

2.7 Examples in Physical Sciences: Biology, Material Science, Chemistry, and Archaeology

Agent-based modeling is finding many applications in the physical sciences. Here agents are particles or molecules, and agent interactions are based on physically-based principals such as chemical kinetics, velocity alignment and cohesive and repulsive forces between agents. (Troisi, Wong, and Ratner 2005) applied agent-based simulation to model molecular self-assembly. Agents consist of individual molecules, and agent “behaviors” consist of the physical laws of molecular interaction. Similar agent-based modeling approaches have also found use in the physical sciences in modeling the self-assembly of nano-materials and in modeling the patterns that develop in agitated mixtures of granular particles. Emergent patterns in nano-materials form as a result of a self-organizing process (Cerro-Prada et al. 2009). For example, holes of uniform sizes form at regularly spaced intervals throughout a material when the proper proportions of materials are mixed together under the right physical conditions. These self-organizing patterns show great regularity in their structure. Granular particles of different sizes and densities when mixed together and slowly rotated cause the materials to segregate from each other or intermingle in regular patterns (Grossman, Aranson, and Ben Jacob 2008). (Grossman, Aranson, and Ben Jacob 2008) demonstrate a variety of emergent behaviors arising in a model of granular materials. They show that collective behaviors

such as swarm migration and vortex formation can emerge using a minimal number of assumptions for isotropic agents due solely to inelastic collisions among the agents.

In biology, agent-based modeling is used extensively to model emergent processes. (Kauffman 1993) has studied gene expression as an emergent process. (Macal 2009) describes the emergent processes investigated through computational agent-based modeling by the field of Artificial Life, including morphogenesis, the emergence of form in biological systems, and the emergence of life itself. General biology applications include modeling cell behavior and cell interaction, the system-wide workings of the immune system, morphogenesis and tissue growth, and the progression of disease processes in the human body. Generally, authors contend that agent-based modeling offers benefits beyond traditional modeling approaches for the problems studied and use the agent-based models as electronic laboratories to augment data collection in their traditional experimental laboratories. Cellular automata are a natural application for modeling cellular systems (Alber et al. 2003). In one modeling approach, the cellular automata grid is used to model structures of stationary cells comprising a tissue matrix. Mobile cells, consisting of pathogens and antibodies, are agents that diffuse through adjacent tissue and co-locate with tissue and other mobile cells. In these models, the concept of the CA is generalized in that the CA assumption that only one agent occupies each cell in the CA grid is relaxed. Some agent-based applications specifically focus on the emergence of and the spread of cancer cells throughout the body (Preziosi 2003). Aspatial models in the biological sciences include The Basic Immune Simulator, which is built on a general agent-based framework to model the interactions between the cells of the innate and adaptive immune system (Folcik, An, and Orosz 2007). Approaches for modeling the immune system have also inspired several agent-based models of intrusion detection for computer networks (Boukerche et al. 2007). Multi-scale modeling applications are also natural approaches for agent-based modeling. (Emonet et al. 2005) developed an agent-based simulator *AgentCell* for modeling the chemotaxis processes for motile behavior of the *E. Coli* bacteria. In this multi-scale simulation, agents are modeled at two distinct levels: as individual molecules and as whole cells. The motile behaviors of a population of bacteria cells as these behaviors arise from the underlying chemistry of the cells is emergent in *AgentCell*. The model is used to study how the range of natural cell diversity at the molecular level is responsible for the observed range of behaviors across a population of bacteria cells.

Archaeology and anthropology are making use of large-scale agent-based modeling by providing an experimental virtual laboratory for long-vanished civilizations. (Kohler, Gumerman, and Reynolds 2005) employed large-scale agent-based simulations based on archaeological evidence to understand the social and cultural factors responsible for the disappearance of the ancient Pueblo in the southwestern U.S. (Wilkinson et al. 2007) used agent-based modeling to understand the growth and decline of ancient Mesopotamian society. (Griffin and Stanish 2007) used agent-based modeling to gain insights into prehistoric regional political consolidation, an emergent property, for the Lake Titicaca basin of Peru and Bolivia for the late prehistoric period circa 2500 BC to AD 1000.

3 ABS AND OTHER SIMULATION APPROACHES

Both ABS and discrete-event simulation are Turing complete (see, e.g., (Epstein 2007; Peterson 1981; Savage, Schruben, and Yucesan 2005; Wolfram 1984)). Therefore, their theoretical modeling power is the same given enough memory and computational time. What differentiates them are their flexibility and efficiency in modeling different types of systems. ABS is suitable for systems with entities that interact frequently with each other. Discrete-event simulation has various world-views (e.g., event-scheduling, process interaction, activity scanning, state machines, and other formalisms) that vary greatly in modeling flexibility and analytical power (Kiviat 1969). The underlying software platforms also play an important role in shaping the development and popularity of discrete-event simulation and the corresponding world-views. But in general, discrete-event simulation focuses on simulating events and their relationships of the underlying discrete-event dynamic system. The events here can be general; for example, the action taken by each agent at each time step in an ABS can be considered as an event. In this sense, ABS is no differ-

ent from a discrete-event simulation (provided that the calculation of all continuous state variables are also discretized). We note that the “interactions” in a process-interaction discrete-event simulation model are those among processes—e.g., arrival process, service process, break-down process, etc.—rather than the entities, which should be distinguished from those “interactions” among agents in an ABS model. Of course, in the most extreme case, one can model each entity as a process in a process-interaction simulation model. However, most existing discrete-event simulation platforms are not designed to handle such a model as it can exponentially increase the number of events, making the model inefficient and hard to analyze.

Actions and interactions of agents in an ABS model usually have direct physical interpretations that are suitable for animation. As a result, the animation capability (which is also enhanced by underlying software platforms) is a major factor that makes ABS appealing to both experts and non-experts. All these capabilities and characteristics make ABS to be one of the most popular simulation modeling approaches (and its popularity is still increasing). We should point out that animation is not a necessary component of an ABS model. An ABS model can run perfectly well without any animation and still provide desired statistical outputs. Animation is a way of visualizing the dynamics and interactions of agents, and is also a powerful way of verifying, validating, and explaining the model.

The development of most simulation paradigms typically involves multiple branches (or stages): modeling techniques, analysis methodologies, algorithms, theoretical foundation, optimization techniques, validation and accreditation methods, and the combination of them, with a ultimate goal of supporting decision making (probably critical and in near real time) in various science and engineering domains. Other simulation paradigms, such as discrete-event simulation, system dynamics, or finite-element-based simulation have already developed a rich set of theories and practices. In contrast, ABS is still at its infancy stage of developing modeling techniques. More work in analytical development, optimization, and validation is needed.

4 CONCLUSIONS

Enough interest in ABS has accumulated to induce a strong momentum for more research to enhance the theoretical foundation of ABS and to improve its efficiency. There are undoubtedly a large number of challenges that need to be solved. Issues and challenges identified in the literature include the development of ABS-specific modeling and analysis approaches distinguished from those of other simulation methods, open access to reported ABS models for purposes of validation and comparison, more credible arguments (examples or theories) supporting the use of ABS compared to other simulation approaches, understanding how to generate emergent behaviors in addition to the identification of their emergence, and more rigorous and effective methods of quantifying emergent behaviors (Bankes 2002; Epstein 2007; Heath, Hill, and Ciarallo 2009).

These issues are in fact fundamental to most computational tools including ABS, which are mostly used to provide deductive reasoning and results. Therefore, the real questions here are how to enhance the theoretical foundation of ABS, how to validate and accredit ABS models and their results, and how to effectively and securely use ABS in different science and engineering domains.

We feel that a viable solution should be a multidisciplinary approach, in which simulationists, computer scientists, cognitive scientists, mathematicians, and social scientists all work together to develop a holistic solution to these challenging problems and issues.

ACKNOWLEDGMENTS

Some of the work presented in this paper was partially supported by the National Science Foundation through grant CMMI-0644959. This work was also partially supported by the U.S. Department of Energy under contract number DE-AC02-06CH11357. We thank Dr. Averill Law for discussions on the relationship between ABS and other simulation approaches.

REFERENCES

- Alber, M. S., M. A. Kiskowski, J. A. Glazier, and Y. Jiang. 2003. On cellular automaton approaches to modeling biological cells. In *Mathematical systems theory in biology, communication, and finance, ima*. eds. J. Rosenthal, and D. S. Gilliam. 134:1-39. New York: Springer.
- Bankes, S. C. 2002. Agent-based modeling: A revolution? *Proceedings of the National Academy of Sciences* 99(Suppl 3):7199-7200.
- Berlekamp, E. R., J. H. Conway, and R. K. Guy. 2003. *Winning ways for your mathematical plays*. 2nd. Natick, Massachusetts: AK Peters, Ltd.
- Beyer, W. A., P. H. Sellers, and M. S. Waterman. 1985. Stanislaw m. Ulam's contributions to theoretical theory. *Letters in Mathematical Physics* 10:231-242.
- Bonabeau, E. A., M. A. Dorigo, and G. A. Theraulaz. 1999. *Swarm intelligence: From natural to artificial systems*. New York: Oxford University Press.
- Boukerche, A., R. B. Machado, K. R. L. Juc, J. O. B. M. Sobral, and M. S. M. A. Notare. 2007. An agent based and biological inspired real-time intrusion detection and security model for computer network operations. *Computer Communications* 30(13):2649-2660.
- Busemeyer, J. R., and J. T. Townsend. 1993. Decision field-theory - a dynamic cognitive approach to decision-making in an uncertain environment. *Psychological Review* 100(3):432-459.
- Cerro-Prada, E., M. J. Vázquez-Gallo, J. Alonso-Trigueros, and A. L. Romera-Zarza. 2009. Nanoscale agent based modelling for nanostructure development of cement. In *Nanotechnology in construction 3, Proceedings of the NICOM3*. 175-180. Springer.
- Chan, W. K. V. 2010. Foundations of simulation modeling. In *Encyclopedia of operations research and management science*. ed. J. J. Cochran. New York: Wiley.
- Chi, L., W. K. Chan, G. Seow, and K. Tam. 2009. Transplanting social capital to the online world: Insights from two experimental studies. *Journal of Organizational Computing and Electronic Commerce* 19(3):214-236.
- Emonet, T., C. Macal, M. North, C. Wickersham, and P. Cluzel. 2005. Agentcell: A digital single-cell assay for bacterial chemotaxis. *Bioinformatics* 21(11):2714-2721.
- Epstein, J. M. 2007. *Generative social science: Studies in agent-based computational modeling (princeton studies in complexity)*. Princeton University Press.
- Folcik, V. A., G. C. An, and C. G. Orosz. 2007. The basic immune simulator: An agent-based model to study the interactions between innate and adaptive immunity. *Theoretical Biological and Medical Modeling* 4(39).
- Griffin, A. F., and C. Stanish. 2007. An agent-based model of prehistoric settlement patterns and political consolidation in the lake titicaca basin of peru and bolivia. *Structure and Dynamics: eJournal of Anthropological and Related Sciences* 2(2).
- Grossman, D., I. S. Aranson, and E. Ben Jacob. 2008. Emergence of agent swarm migration and vortex formation through inelastic collisions. *New Journal of Physics* 10(2):023036.
- Heath, B., R. Hill, and F. Ciarallo. 2009. A survey of agent-based modeling practices (january 1998 to july 2008). *Journal of Artificial Societies and Social Simulation* 12(4):9.
- Helbing, D., I. Farkas, and T. Vicsek. 2000. Simulating dynamical features of escape panic. *Nature* 407(6803):487-490.
- Kauffman, S. A. 1993. *The origins of order: Self-organization and selection in evolution*. Oxford, UK: Oxford University Press.
- Kiviat, P. J. 1969. *Digital computer simulation: Computer programming languages*. Memorandum RM-5883-PR. The Rand Corporation Santa Monica, CA.
- Kohler, T. A., G. J. Gumerman, and R. G. Reynolds. 2005. Simulating ancient societies. *Scientific American* 293(1):77-84.
- Law, A. 2006. *Simulation modeling and analysis*. 4th. Boston: McGraw-Hill.

- Lee, S., and Y.-J. Son. 2008. Integrated human decision making model under belief-desire-intention framework for crowd simulation. In *Proceedings of the 2008 Winter Simulation Conference*. eds. S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, and J. W. Fowler. 886-894. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Lee, S., Y.-J. Son, and J. Jin. 2008. Decision field theory extensions for behavior modeling in dynamic environment using bayesian belief network. *Information Sciences* 178(10):2297-2314.
- Lee, S., Y.-J. Son, and J. Jin. 2010. Integrated human decision making and planning model under extended belief-desire-intention framework. *ACM Transactions on Modeling and Computer Simulation* (accepted).
- Lysenko, M., and R. M. D'Souza. 2008. A framework for megascale agent based model simulations on graphics processing units. *Journal of Artificial Societies and Social Simulation* 11(4):10.
- Macal, C. M. 2009. Agent based modeling and artificial life. In *Encyclopedia of complexity and systems science*. ed. R. Meyers:112-131. New York: Springer.
- Macal, C. M., and M. J. North. 2005. Tutorial on agent-based modeling and simulation. In *Proceedings of the 2005 Winter Simulation Conference*. eds. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines. 2-15. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Macal, C. M., and M. J. North. 2006. Tutorial on agent-based modeling and simulation part 2: How to model with agents. In *Proceedings of the 2006 Winter Simulation Conference*. eds. L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto. 73-83. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- Macal, C. M., and M. J. North. 2009. Agent-based modeling and simulation. In *Proceedings of the 2009 Winter Simulation Conference*. eds. M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls. 86-98. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.
- O'Neill, N. 2010. Google now indexes 620 million facebook groups. Available via <<http://www.allfacebook.com/google-now-indexes-620-million-facebook-groups-2010-02>> [accessed October 10, 2010].
- Page, B., N. Knaak, and A. Kruse. 2007. A discrete event simulation framework for agent-based modelling of logistic systems. In *INFORMATIK 2007 Informatik trifft Logistik Band 1 Beiträge der 37. Jahrestagung der Gesellschaft für Informatik e.V. (GI)*. eds. R. Koschke, O. Herzog, K.-H. Rodiger, and M. Ronthaler. 397.
- Peterson, J. L. 1981. *Petri net theory and the modeling of systems*. New Jersey: Prentice-Hall.
- Preziosi, L. 2003. *Cancer modelling and simulation*. Boca Raton, FL: Chapman and Hall/CRC.
- Railsback, S. F., S. L. Lytinen, and S. K. Jackson. 2006. Agent-based simulation platforms: Review and development recommendations. *Simulation* 82(9):609-623.
- Rapoport, A., and A. M. Chammah. 1966. The game of chicken. *American Behavioral Scientist* 10(3):10-28.
- Rendell, P. 2002. Turing universality of the game of life. In *Collision-based computing*:513-539: Springer-Verlag.
- Reynolds, C. W. 1987. Flocks, herds, and schools: A distributed behavioral model. *SIGGRAPH '87, Computer Graphics (ACM)* 21(4):25-34.
- Savage, E. L., L. W. Schruben, and E. Yucesan. 2005. On the generality of event graph models. *INFORMS Journal on Computing* 17(1):3-9.
- Troisi, A., V. Wong, and M. Ratner. 2005. An agent-based approach for modeling molecular self-organization. *Proceedings of the National Academy Sciences* 102(2):255-260.
- Von Neumann, J. 1966. *Theory of self-reproducing automata*. Champaign, IL: University of Illinois Press.
- Wilensky, U. 1999. Netlogo. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University. Available via <<http://ccl.northwestern.edu/netlogo/>> [accessed October 10, 2010].

- Wilkinson, T. J., M. Gibson, J. H. Christiansen, M. Widell, D. Schloen, N. Kouchoukos, C. Woods, J. Sanders, K.-L. Simunich, M. Altaweel, J. A. Ur, C. Hritz, J. Lauinger, T. Paulette, and J. Tenney. 2007. Modeling settlement systems in a dynamic environment. In *The model-based archaeology of socionatural systems*. eds. T. A. Kohler, and S. v. d. Leeuw:175-208. Santa Fe, NM: School for Advanced Research Press.
- Wolfram, S. 1984. Universality and complexity in cellular automata. *Physica D: Nonlinear Phenomena* 10D:1-35.
- Wooldridge, M., and N. R. Jennings. 1995. Agent theories, architectures, and languages: A survey. In *ECAI-94 Workshop on Agent Theories, Architectures, and Languages*. 1-32. Springer-Verlag.
- Xi, H., and Y. Son. 2010. An integrated pedestrian behavior model based on extended decision field theory and social force model. In *Human-in-the-loop simulation: Methods and practice*. eds. L. Rothrock, and S. Narayanan: Springer (accepted).
- Zhou, Z. 2010. Agent-based simulation and optimization with learning and applications on electricity market. Department of Decision Sciences & Engineering Systems. Troy, NY, Rensselaer Polytechnic Institute. Ph.D.
- Zhou, Z., W. K. V. Chan, and J. Chow. 2009. Agent-based simulation of electricity markets: A survey of tools. *Artificial Intelligence Review* 28(4):305-342.

AUTHOR BIOGRAPHIES

WAI KIN (VICTOR) CHAN is an Assistant Professor of the Department of Industrial and Systems Engineering at the Rensselaer Polytechnic Institute, Troy, NY. He holds a Ph.D. in industrial engineering and operations research from University of California, Berkeley. He has received the 2006 Pritsker best Ph.D. thesis award, the 2007 NSF CAREER Award, the 2007 and 2008 IEEE CASE Best Paper Award finalists, and the 2010 INFORMS Service Science Best Paper Award. His research interests include discrete-event simulation, agent-based simulation, and their applications in energy markets, social networks, service systems, transportation networks, and manufacturing. His e-mail address is chanw@rpi.edu.

YOUNG-JUN SON is a Professor of Systems and Industrial Engineering at The University of Arizona. He is an associate editor of the *International Journal of Modeling and Simulation* and the *International Journal of Simulation and Process Modeling*. He has received the SME 2004 Outstanding Young Manufacturing Engineer Award, IIE 2005 Outstanding Young Industrial Engineer Award, IERC Best Paper Awards (2005 in modeling and simulation; 2008 in homeland security; 2009 in modeling and simulation), and the Best Paper of the Year Award in 2007 from *International Journal of Industrial Engineering*. He can be reached by email at son@sie.arizona.edu.

CHARLES M. MACAL, Ph.D., P.E., is the Director of the Center for Complex Adaptive Agent Systems Simulation, Argonne National Laboratory, Senior Fellow in the joint Computation Institute of the University of Chicago and Argonne, and Adjunct Professor in the Graham School of the University of Chicago. Dr. Macal has a Ph.D. in Industrial Engineering & Management Sciences (Operations Research) from Northwestern University and a Master's Degree in Industrial Engineering (Computer Simulation) from Purdue. He is also a Registered Professional Engineer. His email address is macal@anl.gov.