Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 44 (2015) 363 – 372
Procedia Computer Science

2015 Conference on Systems Engineering Research

Simulation Approaches for System of Systems: Events-Based versus Agent Based Modeling

W. Clifton Baldwin[a], Brian Sauser[b], and Robert Cloutier[c]*

[a]*Federal Aviation Administration, Altantic City, NJ, USA*
[b]*University of North Texas, Denton, TX, USA*
[c]*Stevens Institute of Technology, Hoboken, NJ, USA*

Abstract

In order to understand how we may engineer system of systems (SoS), we will have to rely significantly on our abilities in modeling and simulation. While there are some models of non-specific SoS, few attempts have been made to demonstrate these models with simulation. Simulations of SoS are usually specific cases. Likewise, in the description of a SoS, most of the approaches have focused on their characterization, yet this characterization has not been greatly utilized as an underlying feature in the modeling and simulation of SoS. We review different modeling techniques and use two converse techniques, i.e. agent-based and event-based modeling, to run a simulation of hypothetical systems collaborating into a SoS. The results of the empirical comparison indicate an agent-based modeling approach would achieve a characterization model better with validation achieved through an event-based approach.

Peer-review under responsibility of the Stevens Institute of Technology.

Keywords: Modeling, Simulation, System of Systems, Systems engineering

1. Introduction

There would be no argument that modeling and simulation plays a vital role in our ability to engineer the integration of systems into a system of systems (SoS) [1]. Consequently, there have and continue to be many efforts on modeling SoS, yet few attempts at demonstrating these models with simulation. Likewise, a review of the

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000 .
E-mail address: author@institute.xxx

Peer-review under responsibility of the Stevens Institute of Technology.
doi:10.1016/j.procs.2015.03.032

literature reveals many endeavors to define a SoS by their features or characteristics [2], while only a few of these models have greatly utilized these underlying features or characteristics. We contend that simulating a SoS via the characteristics is one way to validate the attributes that produce the objective SoS.

Consequently, we are seeking to simulate the behavioral characteristics that a systems engineer may be able to modify in order to influence certain aspects of a SoS. Specifically, we want to determine how altering the degree of specified characteristics could impact the formation of SoS [3]. Equipped with this knowledge, a systems engineer may modify the chances of forming a SoS or at least have some predictor for the likelihood of the formation.

SoS has become a readily accepted term to classify an arrangement of independent and interdependent systems that delivers unique capabilities. There have been many attempts at defining and depicting these complex systems based on experience, and while many of the definitions designate SoS as a new entity, the definitions vary [4,5]. Gorod, et al. [2] revealed many endeavors to define and characterize SoS and concluded that a characterization is a more optimal approach to understanding them. Given the diverse descriptions of SoS, the characterizations are usually a list of various features that appear mostly anecdotal.

We are seeking to understand SoS and their properties through modeling and simulation. In this paper we will present a review of the literature regarding modeling and simulating. To determine the optimal simulation platform for a non-specific SoS, we contrast the major modeling paradigms from the literature and perform a comparison of agent-based modeling (ABM) versus event-based modeling (EBM) (also known as discrete modeling). Agent-based modeling, or individual based modeling as it is known also, has been extensively used in ecology where the researcher can program particular behaviors for entities and then watch the simulation produce interactions among many entities [6]. Event-based modeling is a common approach for systems engineering since the researcher can program different states a system undergoes in order to learn something about its underlying behavior. In section 4, we present an experiment to evaluate ABM and EBM to determine the utility of each modeling and simulation approach to SoS given the accepted characterization. We conclude with a discussion of the implications of this study.

2. Modeling and Simulation of SoS

Many studies in the literature examine SoS to understand them better [4], but we would like to learn more than passive observation allows. First we adopt a description of a SoS as a composite system composed of autonomous, diverse constituent systems that are dynamically connected and belong through contributions to the goals of the SoS [7]. Therefore, we argue that a SoS must have at least four basic attributes: autonomy, belonging, connectivity, and diversity. Furthermore a SoS exhibits emergence [8], although we defer exploring this characteristic to future work. To briefly summarize the definitions of these attributes, autonomy is the ability of a constituent system to complete its own goals within limits and without the control of another entity. The system's goal is the reason the constituent system exists [4]. Belonging is the ability of a constituent system to choose to contribute value to the goals of another system in exchange for value to its own goals [9], and diversity ensures the different systems have different goals [7,8]. Finally, connectivity is more than just having a connection but refers to a dynamical nature of information flow between constituent systems.

Given the behavioral characteristics that have been associated with SoS, modeling and simulation is one approach to validate the characteristics and simultaneously search for additional properties. By definition, a model is a simpler representation of some system of interest, and a simulation is the operation of the model for usefully inferring behavior [10]. More specifically, a model reproduces the characteristics of interest in order to observe specific behaviors [11]. This model should contribute to the understanding of the SoS. Although there have been various attempts at modeling SoS, few examples for simulating SoS from the literature are presented below.

One EBM approach to simulating a SoS involves the Discrete Event System Specification (DEVS) formalism. DEVS models interoperability events using architectural modeling techniques, such as the Unified Modeling Language (UML) or the Department of Defense Architectural Framework, which is commonly referenced as DoDAF. Once the interactions of the systems are understood, the messages that pass between systems are captured using the eXtensible Markup Language (XML). Basic models known as atomic models simulate each system in the SoS. The DEVS simulator imitates the SoS by taking the XML data as input to atomic models and outputting processed XML data [12]. A similar technique has been used to model the Global Earth Observation System of

Systems or GEOSS using the Systems Modeling Language (SysML), a modified UML. In this case, the various diagrams are simulated using Coloured Petri Nets, a graphically-oriented computer language [13]. Coloured Petri Nets allow developers to describe states and actions of the system and observe representations via simulations. Along the same lines, a report out of Sandia National laboratories provides a modified state chart approach where any event can occur at any point in time. Custom software implements the model as a simulation [14]. Further support for SysML as an appropriate tool is the modeling of the fictitious system of systems FireSAT, albeit there was no intention for a simulation [15]. Phantom System Models (PSM) is a modeling methodology that focuses on the relationships between and among any two systems within a system of systems and uses traditional systems engineering tools to support the methodology [16]. The PSM includes elements of ABM for handling the interrelationships.

ABM replicates independent agents in order to study their interdependencies. This approach can generate information for policy-makers by revealing the policy's effects on player interactions [17]. Nobel laureate Thomas Schelling applied cellular automata to study housing segregation in one of the first ABM studies. Schelling's neighborhood model includes multiple entities constructed from two distinct agent types [11]. The individual preferences of each agent lead to a segregated SoS [18]. Admitting that ABM is a tool for complex systems, Lerman, Galstyan, and Hogg [19] and Laubenbacher et al. [20] present some mathematical frameworks for ABM. Examples include stochastic systems such as a finite difference equation [19] and the mathematics for cellular automata, Hopfield networks, communicating finite state machines, and finite dynamical systems [20].

In many if not most cases, the simulated specific object is more important than the model. For example, applying an ABM approach simulates a Navy Warfighter ship to explore the impacts of inserting wireless technology [21], a supply chain for distributed manufacturing environment in order to plan a successful manufacturing operation [22], and dynamic job allocation among resources in a manufacturing shop floor in order to improve scheduling and allocation of machines [23]. Lewe, DeLaurentis and Mavris [24] apply ABM to study the national transportation system, where agents represent entities that travel between various locales.

The use of game theory is a mathematical approach to modeling agents. Game theory has been applied to model the interactions of systems within a space situational awareness network SoS [25] and for an inferred SoS within an artificial world [26]. Interacting systems for dynamic shop floor routing are modeled as games of coordination [27]. This study uses repeated games in an ABM to determine if there is any convergence to a Nash Equilibrium. DiMario et al. [3] have applied elements of utility theory to model the workings of a SoS. They developed an approach for the collaboration of a SoS using satisficing game theory, which permits the direct consolidation of the SoS interests with its component systems' interests. This approach requires algorithms that collect information from multiple component systems in order to compute a decision. Therefore ABM was not attempted "due to the lack of visibility of how agents in available agent software actually communicate" [3]. This study models behavior of a SoS by applying the characteristics to the decision process but does not necessarily model the actual characteristics.

Natural phenomena provide many examples that can be considered SoS. Application of the Artificial Life framework models the multiple architectural levels of a generic SoS. The cognitive architecture level is tested in respect to physical, social and behavioral perspectives using ABM simulations [28]. Two biological organisms representing SoS are Escherichia coli (E. coli) and a flock of birds. Simulations using MatLab implement models of the swarming behavior of E. coli [29]. In addition to foraging behavior of the bacteria, the simulation demonstrates the swarming of organisms when stressed. Similar to a SoS, the organisms sacrifice independence and speed for the safety of a swarm. An earlier biological model called the boids simulation demonstrates how birds can flock using only three simple rules applicable to flying birds [30]. Again there is no mention of SoS, but we argue that a flock of birds is a biological SoS in that the flock acts as a single system yet is composed of individual birds. Referenced as an individual-based model, this modeling paradigm is a type of ABM.

A noticeable dearth in the literature is the modeling of the underlying characteristics in order to simulate engineered SoS. Conversely some biological SoS are simulated from their species-specific behaviors in nature, such as the boids [30], microbes [31], plants [32], and a forest ecosystem [33]. This observation is strange since many descriptions of non-biological SoS are based on generic behavior [4].

3. Comparison of Approaches

Most model types reduce into the perspectives of looking down at a system or up from the system. Various modeling approaches are available, such as discrete event or EBM, ABM, and mathematical equation modeling. EBM looks down at a system by modeling its encounters of expected events. Conversely, ABM considers the functionality of a system to see what actions result.

ABM is a computational tool that can produce system functionality by programming specific elements of the system [34–36]. A description of ABM states, "a system is modeled as a collection of autonomous decision-making entities called agents. Each agent individually assesses its situation and makes decisions on the basis of a set of rules" [34]. These agents are distinct parts of a software program and can interact based on each agent's rules [36].

An agent is a distinct software program that represents social actors, which may be people, animals, organizations, or any individual system. These agents must be flexible and able to function independently of their environment, at least within the situations of interest. In addition they can be heterogeneous, autonomous, and "boundedly rational." A boundedly rational agent acts within a set of rules limited by its available information [36]. In other words, the agent does not always act rationally since it may not have sufficient information to make an informed decision. Therefore boundedly rational agents are good choices to model realistic situations where decisions are almost always limited. An additional strength of ABM is the ability for the agents to interact [36].

As a complementary description, "Agent-based models provide computational demonstrations that a given microspecification is in fact sufficient to generate a macrostructure of interest" [35]. Consequently ABM can be used to test hypotheses regarding characteristics to see if the expected outcome results. ABM can reveal new testable hypotheses also. The literature may validate these predictions or the hypotheses can serve as the basis for new experiments [37].

The strengths of ABM have been applied for biological research, as one example. Primates are modeled using ABM, which resulted in new explanations for certain primate social structures between males and females. Some of the results were validated from the literature where available, while other results caused the researchers to reconsider the biological correlates of their model [37].

While ABM focuses on agents, EBM deals with events. "Basically, an event is an externally observable phenomenon, such as an environmental or a user stimulus, or a system response, punctuating different stages of system activity" [38]. In these discrete event simulations, a discrete occurrence causes an instantaneous response to the system under study [10]. Although ABM implementation can refer to a type of discrete event simulation [20], we differentiate EBM for our purposes as state events rather than just the time-step events of ABM.

EBM represent a collection of events that impact the system of interest. Basically, "[EBM do not] record past states of a system, but rather events that change the state" [39]. Therefore EBM is a good choice when information is known about how a SoS reacts to a given situation.

Some mathematical equation modeling is just a form of EBM. This approach uses equations to represent the system's states, and the simulation is the evaluation of the equations [40]. Individuals within the model represent entities that perform some function over time, and observables are measurable variables of interest to the researcher. For example differential equation models start with equations representing relationships among the mostly system-level observables and evolve over time. In contrast, ABM starts with behaviors of the individuals and allows them to interact. As the simulation advances, the states and relationships among the agents change, which produces system-level information [40].

An approach called equation-free macroscopic analysis combines ABM with mathematical equations. This approach has been applied on an automated industrial transport system. The vehicles in the system decide autonomously where to move in order to perform their missions. The requirements for autonomous systems influenced the choice of ABM, but the model included system-wide requirements also. Since these requirements were quantified mathematically, an equation-based model had some desirable elements as well. Therefore a combination of the two modeling techniques obtained the benefits of both. The approach used in the study was to generate initial conditions for an ABM based on observation. After simulating the model, numerical analysis produced new values for the observed variables, which were reapplied [41].

To create an ABM, the modeler must understand behavioral qualities regarding the systems of interest. Thus the modeler proposes the underlying rules that direct the system and converts those rules into algorithms for the agents.

These rules may be simple or very complicated as needed. For an EBM, the modeler must understand the different states of a system of interest and the events it may encounter. In EBM, "When an event occurs information about the event may be registered and activity may be initiated" [42]. In essence, the difference is one between programming action and programming reaction. Furthermore the results of the two modeling techniques differ also. By entering how an agent acts, the ABM simulation produces how the agent reacts and changes in response to its environment. An EBM appears to have the opposite result. The various rules of how a system will react to events results in a state change producing different output. This output can teach the modeler something about the system's behavior during the simulation. Although discussing digital circuits, Hall [43] argues that event-based simulation is a technique for predicting the behavior of certain systems. Nonetheless both modeling methods result in a better understanding of the system of interest and can support each other.

An advantage of EBM is its simplicity in creating the model [38]. The modeler can think of the EBM as a "black box" in which the internal workings are unknown. As such, the modeler codes the EBM to react according to actual observations. Another advantage of EBM is in its testability. "For [event-based] modeling, the approach merges the notions of state, input, and output to events. This event-based view is more convenient for a tester because he or she is not primarily interested in internal states of the system under test, but rather in events that can externally be observed, perceived and evaluated" [38]. These same events may be a disadvantage if the modeler does not have knowledge of their causes [44]. In other words the events and the reactions to the events must be well understood for a valid simulation.

In many cases, simple rules produce the desired actions in an ABM. For example, a few simple preference rules are sufficient to explain segregation in neighborhoods [11,18]. Simulation of a flock of birds requires three rules related to flying preferences [30]. Another advantage of ABM is its ability to serve as a scientific hypothesis testing tool where validation of the hypothesis is a validation of the ABM [37].

As summarized in Table 1, both approaches have their merits and shortfalls. Therefore the question is not a matter of the better modeling paradigm but rather a question of the better approach for our specific purposes.

Table 1. Comparison of EBM and ABM

Event-Based (Discrete) Modeling	Agent-Based Modeling
Macrospecifications reveal microstructures (top-down view)	Microspecifications generate macrostructure (bottom-up view)
Externally observable phenomenon (events)	Autonomous decision making entities (agents)
Programmed response to discrete events	Programmed functionality of agents
Events adhere to system-level observable information	Agents adhere to behavioral rules (boundedly rational)
System of interest changes state in response to events	Agents function independently and flexibly
Event impacts the entire entity	Agents interact as distinct parts of simulation
Simplicity in modeling inputs, state, and outputs	Simplicity in modeling rules
Internal behavior is unknown	Events emerge
Easy to test	Difficult to validate

4. Empirical Comparison

4.1. Scenario

Table 1 in the previous section presents a notional comparison of ABM to EBM. In this section, we present an empirical comparison based on a hypothetical SoS example. Since the experiment's goal is only to select the modeling paradigm appropriate for our purposes, the scenario is quite simple. Consequently a straightforward comparison is sufficient to evaluate the differences presented by each approach. The ABM is coded and executed in the multi-agent programmable modeling environment named NetLogo [45], and the EBM is run in GPSS/H, a Wolverine software product.

For the purposes of our fabricated example, suppose the research question concerns the collaboration of authors for an academic paper. Collaboration causes a team to form around a specific topic of interest to the authors.

Depending on the number of potential authors, multiple teams may represent different paper topics. Each author is an autonomous system yet the goal of publishing applies to the entire group. The authors' skills differ in that some are graduate students while others are professors with varying levels of tenure. The authors are connected usually from their mutual academic institution, but also they share a common interest related to the topic of their potential publication. Of course truly autonomous authors can work alone on their papers, but this action does not help the team. By joining forces, a paper will benefit in part from the experience of professors and the time and dedication of graduate students. In return, graduate students receive the recognition from publishing in a quality journal while professors are published despite their busy schedules. These actions adhere to the SoS characteristics proposed by Boardman and Sauser [8]. Therefore this contrived system of interest is a SoS with its constituent systems, which is the objective of the simulations. Furthermore for this scenario, each academic team can attempt to publish in one of three types of publication, which correspond to various levels of quality. Peer-reviewed journals refer to the highest quality with peer-reviewed conference papers the next level. A below-average paper is submitted to a trade journal. Corresponding to each level of publication is a specific probability of rejection.

4.2. Agent-Based Model

In the agent-based case, each agent represents an author with a specific skill level and topic of interest. Agents wander around the field searching for other agents that share their interest. As agents gather around a topic, they form groups for the purpose of writing a paper for publication. The authors are connected via their common topic. Each author is autonomous with individual skills and belongs to the group by contributing his or her skill to the paper. The authors' skills were randomly selected from a normal distribution. In addition to the skill levels, there is diversity in that some authors are full professors, some are assistant professors, and some are graduate students, each with the potential for unique properties. The collective skill levels of a SoS are summed over a specific period of time, which represents a cycle of writing papers. Therefore some papers represent a higher quality than others. Once a paper is produced, it is submitted for an appropriate level of publication. Depending on the publication level, there is a probability of rejection. Papers that are accepted are counted in the results of the simulation. Only papers with a quality rating at least one standard deviation above the mean qualified for the highest publication, and the next level required the paper to rank above the mean average.

The ABM simulation reset itself every 100 ticks of the counter, which was more than enough time to complete a cycle, and the simulation was run for 1300 cycles. Each cycle had 30 agents with five different topics. Therefore each group formed with an average of six authors, but the quality of papers was calculated without regard of the quantity of authors to allow equal opportunity for every group.

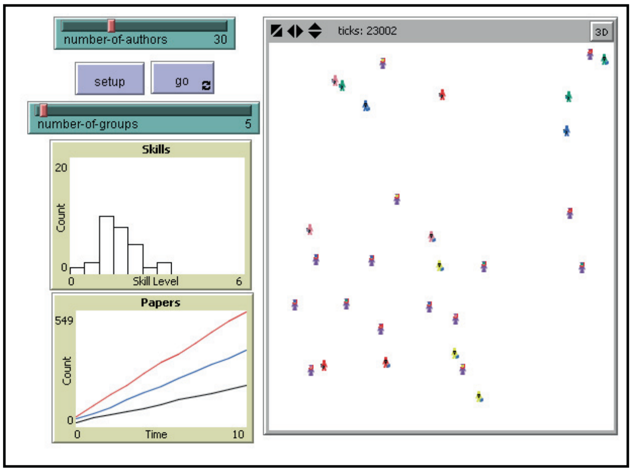

Fig. 1. Screen shot of ABM

Fig. 1 shows a screen shot of the simulation with controls and monitors on the left. The field on the right side shows the agents randomly placed. For aesthetics only, different colors represent different topics. As the simulation progresses, the agents gather by topic and writing commences as described. Output statistics are collected in a text file.

4.3. Event-Based Model

For the discrete case, an event signalled the formation of a group with a specific topic. Since the SoS formation does not consist of gathering systems, differentiation of topic is irrelevant. However a probabilistic distribution could select a topic from a set, if desirable. The time to form a SoS after a topic is selected is modeled followed by a set duration to model the writing. The group writes a paper and submits it for publication based on a probabilistic decision. The quality of paper and the chance of rejection are modeled as normal probabilities, which result in a paper count. In a normal distribution, approximately 15.9% should be greater than one standard deviation above the mean. Therefore 15.9% of all papers were submitted for the highest level of publication to correspond with the ABM. The group would enter a waiting period that completed with a rejection, acceptance, or request to rewrite action. A rewrite would only delay publication. The probability of rejection for each publication level was equivalent to the rejection rate in the ABM case, respectively. Descriptive statistics for published papers are collected in a text output file. The basic logic for the EBM is shown in Fig. 2. In order to gather results, the EBM simulation was run 100 separate times each with 100 GPSS/H tokens to represent SoS entities.

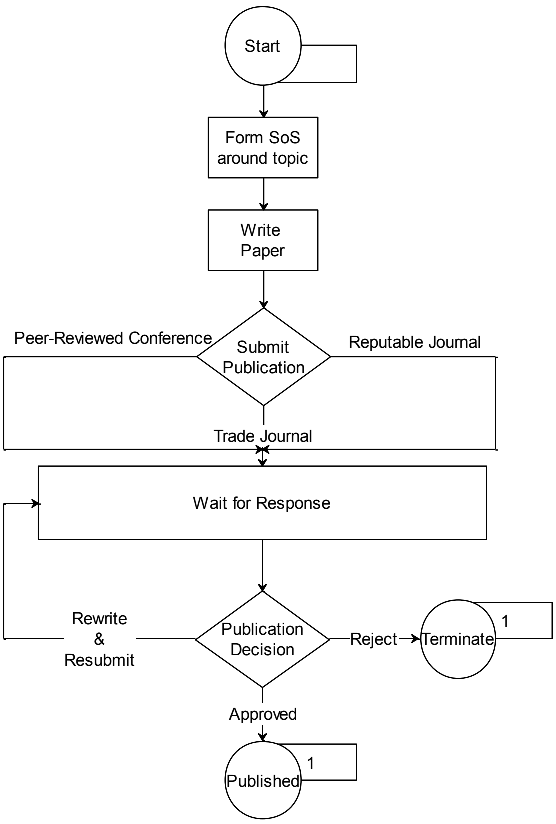

Fig. 2. Flow logic of EBM

4.4. Results

Comparing the results to expectations validates the SoS simulations introduced in the previous sections. While validation usually considers the desired outcome, we have modeled a toy system. Therefore, we compared the two models to each other as well as to mathematical predictions, which is a form of face validity. Furthermore trace validation follows the behaviors of the agents in the ABM to determine if the model's logic adheres to expectations [46]. The resulting statistics of papers are within the margin of error for both approaches with the mathematically expected values (Table 2). Therefore the real difference is the requirements of the modeling approaches to address a specific question. For example, each agent in the ABM has a skill variable that determines the contribution to the paper. On the other hand, the EBM has an assigned probability of producing a particular outcome to represent skill. Specifically, the skill in the ABM is set as a normally distributed variable, but the EBM has a probability of writing a peer-reviewed journal paper selected from the normal distribution table.

Table 2. Output Results of Simulations

	ABM	EBM	Expected
Prestigious journals published	8.9%	9.6%	9.5%
Prestigious journals attempted	14.9%	15.9%	15.9%
Conference papers published	24.8%	24.9%	25.6%
Conference papers attempted	32.6%	33.0%	34.1%
Trade journals published	41.7%	40.9%	40.0%
Trade journals attempted	52.5%	51.1%	50.0%

An advantage of the ABM is the ability to have diverse authors in the group. For example, the ABM simulation had varying proportions of professors, assistant professors, and graduate students, but the EBM simulated the SoS as one entity. Therefore each EBM SoS was individually coded to represent the group skill, but the ABM SoS developed its group skill level based on its constituent systems.

Both approaches provide a means to experiment with time and to model utilization of shared resources. However, SoS formation is modeled in the EBM while the ABM simulated the formation. In other words, we had to program the EBM to form a SoS at a specific point in time, but we coded rules into each agent on what to do upon encountering another agent in the ABM. One limitation of the ABM, or perhaps a limitation of our programming ability, was the lack of multiple agents in the form of a SoS to act as a single agent. We will have to consider this limitation if we proceed with ABM. This drawback of ABM is not an issue with EBM where the SoS is modeled as one entity.

These two simulations demonstrate each of the properties illustrated in Table 1 with one exception. The EBM responds to discrete events as expected, but it is possible to introduce some of these events into the ABM also. In the ABM scenario, the groups submitted papers for publication, which spurred the event of a response. This response took the form of accepting or rejecting the paper, and corresponded closely to the event in the EBM.

Given that we intend to model the common characteristics and simulate the formation of a SoS, the ABM paradigm appears preferred for our purposes. As was demonstrated, autonomy and diversity were modeled for each agent within the ABM. Although somewhat trivial, rules to model belonging and connectivity between agents were modeled in the ABM also. The result of the ABM simulation approximated a SoS. On the other hand, the EBM required modeling of each SoS based on a priori probabilities of topic selection representing the various belonging. There was no indication of the autonomous agents within the SoS and therefore no indication of their diversity. The connectivity was assumed, although another a priori probability could have modeled the connectivity. Hence it is apparent from this simple experiment that ABM is the obvious paradigm for our purpose of modeling the characteristics within and between constituent systems to simulate the formation of a SoS. With that stated, EBM may be useful also as a validation technique. While EBM handles a SoS as a single entity, the agents within an ABM must mimic that single entity. If the collective actions of the ABM differ from the expected actions of a SoS, then the ABM fails to be validated as a SoS model.

5. Conclusions and Future Research

When a SoS is simulated, the approach has focused on the information flow or discrete events of the system, and most simulations are specific systems. Few simulations were found in the literature where the basis for the model is the SoS behavior or the actions of the component systems. Furthermore no simulations were found where the SoS characteristics are directly modeled in order to produce the simulation. Therefore there appears to be a gap in the literature for modeling and simulating the behavior of SoS. We want to determine the influence of characteristics on the formation of SoS through modeling and simulation. Therefore we conducted a simple experiment to compare two of the major modeling paradigms. Our comparison indicates ABM appears better given our situation, and our experiments imply some of the limits of this approach. Of course the difficulty of validating an ABM must be addressed prior to pursuing this approach to completion.

In general, both ABM and EBM have their uses. EBM is preferred when the subject of the simulation involves a series of events, such as the spread of a fire through the forest or the movement of customers through a store. In other words, EBM is useful in examining the results of a system. The stochastic elements of an EBM contribute to an understanding of the system of interest, and the ease in testing provides a level of confidence in the simulation. When the agents are not going through a series of events but rather are responding in some way to their interactions, then ABM becomes the preferred modeling paradigm. Unlike agents represented in an EBM that react to certain events, agents in an ABM change their action and even their behavior based on reactions to other agents, that are also changing based on their interactions. Therefore ABM should be preferred when the modeler is interested in characteristic behavior of system of interest rather than the results of a system activity. While many systems of interest include these interactive behaviors, the complexity of ABM leads to increased scrutiny or decreased confidence. To overcome this shortfall, a thorough emphasis should be placed on validation. Finally, if the modeler is interested in determining characteristic behaviors as a system of interest reacts to certain events, a combination of ABM and EBM may be appropriate. We have demonstrated a situation where an EBM helps validate an ABM. However, when the two approaches are combined, the problem of validation may be exacerbated. Nonetheless a combined EBM and ABM approach may be very useful to answer many questions about systems.

This paper presents our findings from modeling SoS based on our adopted characterization. Although we cannot and do not intend to state there is only one way to model SoS, we hope our work provides insights for others in the field. We believe that modeling SoS will advance the body of knowledge by indicating changes a systems engineer can make to force dynamism in SoS form and function, whilst maintaining SoS control and resilience, minimizing vulnerability, and increasing SoS agility.

References

[1] M. Jamshidi, System of Systems Engineering, Wiley, Hoboken, NJ, 2008.
[2] A. Gorod, B. Sauser, J. Boardman, System-of-systems engineering management: A review of modern history and a path forward, IEEE Syst. J. 2 (2008) 484–499. doi:10.1109/JSYST.2008.2007163.
[3] M. DiMario, J. Boardman, B. Sauser, System of systems collaborative formation, IEEE Syst. J. 3 (2009) 360–368. doi:10.1109/JSYST.2009.2029661.
[4] W.C. Baldwin, B. Sauser, Modeling the characteristics of system of systems, in: 2009 IEEE Int. Conf. Syst. Syst. Eng. SoSE, IEEE, Albuquerque, NM, 2009: pp. 1–6.
[5] A.P. Sage, S.M. Biemer, Processes for system family architecting, design, and integration, Syst. J. IEEE. 1 (2007) 5–16.
[6] V. Grimm, S. Railsback, Agent-based models in ecology: Patterns and alternative theories of adaptive behaviour, in: Agent-Based Comput. Model., Physica-Verlag HD, 2006: pp. 139–152. http://dx.doi.org/10.1007/3-7908-1721-X_7 (accessed May 14, 2009).
[7] W.C. Baldwin, W.N. Felder, B.J. Sauser, Taxonomy of increasingly complex systems, Int. J. Ind. Syst. Eng. 9 (2011) 298–316.
[8] J. Boardman, B. Sauser, System of Systems - the meaning of of, in: Proc. 2006 IEEESMC Int. Conf. Syst. Syst. Eng., IEEE, Los Angeles, CA, 2006: pp. 118–123. doi:10.1109/SYSOSE.2006.1652284.
[9] W.C. Baldwin, T. Ben-Zvi, B.J. Sauser, Formation of collaborative system of systems through belonging choice mechanisms, IEEE Trans. Syst. Man Cybern.-Part Syst. Hum. 42 (2012) 793–801. doi:10.1109/TSMCA.2011.2164522.
[10] A. Maria, Introduction to modeling and simulation, in: S. Andradóttir, K.J. Healy, D.H. Withers, B.L. Nelson (Eds.), Proc. 1997 Winter Simul. Conf., Atlanta, GA, 1997: pp. 7–13. http://www.wintersim.org/prog97.htm (accessed February 16, 2008).
[11] T. Schelling, Micromotives and macrobehavior, W.W. Norton & Co., New York, 1978.
[12] S. Mittal, B.P. Zeigler, J.L.R. Martín, F. Sahin, M. Jamshidi, Modeling and simulation for systems of systems engineering, in: M. Jamshidi (Ed.), Syst. Syst. Innov. 21st Century, John Wiley & Sons, Hoboken, NJ, 2008.
[13] M. Rao, S. Ramakrishnan, C. Dagli, Modeling and simulation of net centric system of systems using systems modeling language and colored Petri-nets: A demonstration using the global earth observation system of systems, Syst. Eng. 11 (2008) 203–220.

[14] J.E. Campbell, D.E. Longsine, D. Shirah, D.J. Anderson, System of systems modeling and analysis, Sandia National Laboratories, Albuquerque, NM, 2005. http://www.prod.sandia.gov/cgi-bin/techlib/access-control.pl/2005/050020.pdf (accessed March 9, 2009).

[15] R. Cloutier, R. Griego, Applying object oriented systems engineering to complex systems, in: 2008 IEEE Int. Syst. Conf. Proc. SysCon 2008, IEEE, Montreal, QC, 2008: pp. 515–520. http://www.scopus.com/scopus/inward/record.url?eid=2-s2.0-49049090228&partnerID=40.

[16] Y.Y. Haimes, Modeling Complex Systems of Systems with Phantom System Models, Syst. Eng. 15 (2012) 333–346.

[17] A. Mostafavi, D.M. Abraham, D. DeLaurentis, J. Sinfield, Exploring the Dimensions of Systems of Innovation Analysis: A System of Systems Framework, IEEE Syst. J. 5 (2011) 256–265.

[18] C. Baldwin, J. Boardman, B. Sauser, Expanding a System of Systems Model with the Schelling Segregation Model, Syst. Res. Behav. Sci. 30 (2013) 65–75. doi:10.1002/sres.2115.

[19] K. Lerman, A. Galstyan, T. Hogg, Mathematical analysis of multi-agent systems, arXiv:cs/0404002. (2004) 1–42.

[20] R. Laubenbacher, A.S. Jarrah, H. Mortveit, S.S. Ravi, A mathematical formalism for agent-based modeling, ArXiv08010249v1 CsMA. (2007) 1–23.

[21] V. Mahulkar, S. McKay, D.E. Adams, A.R. Chaturvedi, System-of-systems modeling and simulation of a ship environment with wireless and intelligent maintenance technologies, Syst. Man Cybern. Part Syst. Hum. IEEE Trans. On. 39 (2009) 1255–1270. doi:10.1109/TSMCA.2009.2025140.

[22] V. Kumar, N. Mishra, A Multi-Agent Self Correcting Architecture for Distributed Manufacturing Supply Chain, Syst. J. IEEE. 5 (2011) 6–15. doi:10.1109/JSYST.2010.2100195.

[23] M. Owliya, M. Saadat, R. Anane, M. Goharian, A New Agents-Based Model for Dynamic Job Allocation in Manufacturing Shopfloors, Syst. J. IEEE. 6 (2012) 353–361. doi:10.1109/JSYST.2012.2188435.

[24] J.-H. Lewe, D.A. DeLaurentis, D.N. Mavris, Foundation for study of future transportation systems through agent-based simulation, in: 24th Int. Congr. Aeronaut. Sci., Yokohama, Japan, 2004: pp. 1–10.

[25] N.B. Shah, M.G. Richards, D.A. Broniatowski, J.R. Laracy, P.N. Springmann, D.E. Hastings, System of systems architecture: The case of space situational awareness, in: AIAA Space 2007, Long Beach, CA, 2007: pp. 1–12.

[26] S.J. Majeski, G. Linden, C. Linden, A. Spitzer, Agent mobility and the evolution of cooperative communities, Complexity. 5 (1999) 16–24. doi:10.1002/(SICI)1099-0526(199909/10)5:1<16::AID-CPLX3>3.0.CO;2-M.

[27] V.A. Cicirello, A game-theoretic analysis of multi-agent systems for shop floor routing, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 2001. http://www.cs.drexel.edu/~cicirello/publications.html (accessed March 24, 2009).

[28] N. Kilicay-Ergin, C. Dagli, Executable modeling for system of systems architecting: An artificial life framework, in: Syst. Conf. 2008 2nd Annu. IEEE, IEEE, Montreal, Canada, 2008: pp. 1–5. doi:10.1109/SYSTEMS.2008.4518983.

[29] K.M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, Control Syst. Mag. IEEE. 22 (2002) 52–67. doi:10.1109/MCS.2002.1004010.

[30] C.W. Reynolds, Flocks, herds and schools: A distributed behavioral model, ACM SIGGRAPH Comput. Graph. 21 (1987) 25–34. doi:10.1145/37402.37406.

[31] F.L. Hellweger, V. Bucci, A bunch of tiny individuals—Individual-based modeling for microbes, Ecol. Model. 220 (2009) 8–22. doi:10.1016/j.ecolmodel.2008.09.004.

[32] A.-K. Bittebiere, C. Mony, B. Clément, M. Garbey, Modeling competition between plants using an Individual Based Model: Methods and effects on the growth of two species with contrasted growth forms, Ecol. Model. 234 (2012) 38–50. doi:10.1016/j.ecolmodel.2011.05.028.

[33] R. Seidl, W. Rammer, R.M. Scheller, T.A. Spies, An individual-based process model to simulate landscape-scale forest ecosystem dynamics, Ecol. Model. 231 (2012) 87–100. doi:10.1016/j.ecolmodel.2012.02.015.

[34] E. Bonabeau, Agent-based modeling: Methods and techniques for simulating human systems, Proc. Natl. Acad. Sci. 99 (2002) 7280–7287.

[35] J.M. Epstein, Agent-based computational models and generative social science, Complexity. 4 (1999) 41–60.

[36] N. Gilbert, Agent-based models: Quantitative applications in the social sciences, Sage Publications, Inc., Thousand Oaks, CA, 2008.

[37] J.J. Bryson, Y. Ando, H. Lehmann, Agent-based modelling as scientific method: A case study analysing primate social behaviour, Philos. Trans. R. Soc. B Biol. Sci. 362 (2007) 1685–1699. doi:10.1098/rstb.2007.2061.

[38] F. Belli, C.J. Budnik, L. White, Event-based modelling, analysis and testing of user interactions: Approach and case study, Softw. Test. Verification Reliab. 16 (2006) 3–32.

[39] P. Plessers, O. De Troyer, S. Casteleyn, Event-based modeling of evolution for semantic-driven systems, in: CAIS 2005 Lect. Notes Comput. Sci., Porto, Portugal, 2005: pp. 63–76.

[40] H.V.D. Parunak, R. Savit, R.L. Riolo, Agent-based modeling vs. equation-based modeling: A case study and users' guide, in: Proc. Multi-Agent Syst. Agent-Based Simul., Springer, Paris, France, 1998: pp. 10–25.

[41] T. De Wolf, G. Samaey, T. Holvoet, D. Roose, Decentralised autonomic computing: Analysing self-organising emergent behaviour using advanced numerical methods, in: Proc. IEEE Int. Conf. Auton. Comput. ICAC05, IEEE Computer Society, Seattle, WA, 2005: pp. 52–63.

[42] L. Bækgaard, Event-based activity modeling, in: ALOIS04 Action Lang. Organ. Inf. Syst., Linköping, Sweden, 2004. http://www.baekgaard.biz/pdf/alois04.pdf (accessed October 17, 2008).

[43] R.J. Hall, A fully abstract semantics for event-based simulations, M.I.T. Artificial Intelligence Laboratory, Cambridge, MA, 1987.

[44] J. Granat, A framework for event based modeling and analysis, J. Telecommun. Inf. Technol. 4 (2006) 88–90.

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

[46] R.G. Sargent, Verification and validation of simulation models, in: B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, E. Yucesan (Eds.), 2010 Winter Simul. Conf., IEEE, Baltimore, MD, 2010: pp. 166–183. http://www.informs-sim.org/wsc10papers/016.pdf (accessed January 3, 2012).