Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

A TAXONOMY FOR CLASSIFYING TERMINOLOGIES THAT DESCRIBE SIMULATIONS WITH MULTIPLE MODELS

Christopher J. Lynch
Saikou Y. Diallo

Virginia Modeling, Analysis and Simulation Center
1030 University Blvd.
Suffolk, VA 23435, USA

ABSTRACT

Many terms exist within Modeling and Simulation that refer to models consisting of more than one modeling paradigm, more than one model, or more than one formalism. To provide some clarification this paper identifies nine terms from the Modeling and Simulation literature and compares them against a taxonomy of model characteristics including time representation, basis of value, behavior, expression, resolution, and execution in order to classify the various terminologies and allow for a discussion from a generalized perspective. Results show that all nine modeling terminologies share the characteristic of resolution, none of the terminologies deal with all six characteristics, and that many of the terminologies deal with only three or less of the characteristics. Finally, this paper explores challenges with using multiple models that contain competing characteristics that are not covered in the literature.

1 INTRODUCTION

A model is a representation of a real or imaginary system and is used in lieu of the real system in order to learn about that system (Fishwick 1995; Sokolowski and Banks 2009). Modeling is the process of solving a problem or answering a question about a real or imagined system through the use of abstraction or simplification (Bennett 1995; Tolk 2012). Models are approximations of the real system that describes specific aspects of that system (Sokolowski and Banks 2010). There are numerous ways to classify a model: physical models represent physical objects; mathematical models represent procedures, algorithms, and mathematical equations that can be solved discretely; and procedural models represent dynamic relationships of situations expressed through mathematical or logical processes (Tolk 2010). Other classifications for models rely on the representation of time, the appearance of randomness, and whether the states within the model change instantaneously or continuously (Fishwick 1995).

Within the Modeling and Simulation (M&S) literature many terms exist and are used interchangeably for describing models that use multiple paradigms, formalisms, or models. For instance, multi-paradigm modeling (MPM) applies when a model consists of more than one modeling paradigm (Vangheluwe, de Lara, and Mosterman 2002), hybrid modeling applies when a model contains more than one model (Mosterman 1999), and multi-method modeling applies when a model contains multiple modeling methods (Borshchev 2013). This causes a problem in clearly identifying the difference between the terms which makes it more difficult to differentiate between different types of models. Multi-paradigm modeling shares similarities with several of the other terminologies. Similar to hybrid modeling, multi-paradigm modeling can consist of both discrete and continuous elements. In this sense, hybrid models qualify as multi-paradigm models; however, not all multi-paradigm models qualify as hybrid models as a multi-paradigm model does not have to contain both discrete and continuous elements.

We propose to address the problem of describing models by clarifying the M&S modeling terminologies through the use of a taxonomy that classifies models with respect to the model characteristics of time representation, basis of value, behavior, expression, execution, and resolution. Additionally, we explore a set of challenges associated with simulations that use multiple models which is not addressed within the literature. This paper is structured as follows: Section 2 provides background information on modeling terminologies within M&S. Section 3 presents a taxonomy of model characteristics for classifying the modeling terminologies. Section 4 presents challenges involving simulations that consist of competing characteristics due to the use of multiple models. Section 5 presents conclusions and identifies future work.

2 BACKGROUND

Single models are not always sufficient for providing the level of depth needed to capture real world processes; whereas simulations that use multiple models can provide a greater depth into the problem (Yilmaz and Ören 2005). Many terminologies exist that provide the ability to capture a problem in greater depth and involve modeling with multiple paradigms, formalisms, or models. A *multimodel* is a model comprised of multiple models that collectively serve to represent the behavior of a system (Fishwick and Zeigler 1992; Fishwick 1995; Fishwick 1998; Tolk 2012) Conducting experimentation with multimodels allows for the simulation to represent several aspects of reality (Yilmaz and Ören 2005; Yilmaz et al. 2007). A *multi-paradigm model* uses two or more modeling paradigms to address a modeling question (MQ) (Lynch et al. 2014; Vangheluwe, de Laura, and Mosterman 2002; Villa and Costanza 2000). A *hybrid model* consists of both discrete and continuous elements (Mosterman 1999; Swinerd and McNaught 2012; Tolk 2012). *Multifacetted modeling* uses multiple models to answer a MQ (Zeigler, 1984). *Multi-resolution modeling* incorporates multiple models from different resolution levels to address the problem (Fishwick and Zeigler 1992; Tolk 2012). A *multi-formalism model* uses at least two formalisms (Vangheluwe, de Laura, and Mosterman 2002). A *coupled model* consists of independent models connected together through a network (Vangheluwe, de Laura, and Mosterman 2002). *Multi-method modeling* uses multiple modeling methods, such as system dynamics (SD) and agent-based modeling (ABM) to create a simulation (Borshchev 2013). *Composite modeling* applies a combination of simulation approaches to address a problem (Viana, et al. 2014). A further discussion on M&S terminologies is presented in Balaban, Hester, and Diallo (2014).

Embedded within the modeling terms are the concepts of methods, methodologies, formalisms, and paradigms. Paradigms, methodologies, and methods assist in transitioning from the real system to the model and each term plays a different role in this transition. A paradigm contains a set of "assumptions, concepts, values, and practices that constitutes a way of viewing reality" that is commonly shared within a community (McGregor and Murnane 2010 pg. 1). The selection of a paradigm "sets down the intent, motivation and expectations" that drives the research process (Mackenzie and Knipe 2006 pg. 2). Modeling paradigms reflect the ways of thinking about how to represent systems within M&S and contain the assumptions that generally accompany each way of thinking (Lorenz and Jost 2006). A paradigm does not need a standard interpretation or a full set of rules in order to guide research (Kuhn 1970). Within M&S, modeling paradigms encompass various ways to think about representing systems and are purposefully well equipped to address specific questions from the real system.

Methodologies and methods provide the connections between modeling paradigms and a model's construction. A methodology is an approach linking a paradigm to research (Mackenzie and Knipe 2006) and deals with the "philosophical assumptions that underlie any natural, social or human science" (McGregor and Murnane 2010 pg. 2). A method is the technique, procedure, or tool used to collect data, conduct research, or analyze data and is based upon the selected methodology (Mackenzie and Knipe 2006; McGregor and Murnane 2010). Formalisms are one method within M&S for implementing models into a computer executable simulation. Formalisms provide explicit representations of a model necessary for the model's implementation on a digital computer (Zeigler 1984). These concepts show a separation of focus on the way of thinking about the system (paradigms) and on how to implement a computer executable simulation (methods, methodologies, and formalisms). However, this does not provide a solid enough foundation for comparing the differences between MPM and the other modeling terminologies.

A taxonomy provides a means for creating a hierarchical classification of a system that is both exhaustive and mutually exclusive (Bailey 1994). Several taxonomies have been applied to M&S. Sulistio, Yeo, and Buyya (2004) provides a taxonomy for designing computer-based simulations with respect to parallel and distributed systems. Taxonomies exist to classify multimodel formalisms based on the structure and behavior of the models (Yilmaz and Ören 2004; Yilmaz and Ören 2005; Yilmaz and Tolk 2008) as well as to classify multimodels based on conceptual, declarative, functional, constraint, and spatial categories contained within the models (Fishwick 1998).

In particular, Sulistio, Yeo, and Buyya (2004) provides a *simulation* taxonomy that characterizes simulations in terms of presence of time, basis of value, and behavior and a *simulation execution* taxonomy that characterizes simulations in terms of execution. These characteristics provide a baseline for describing a model. Tolk, Turnitsa, Diallo and Winter (2006) describes models through their atomic, aggregated, and composite levels of resolution while other models are described in terms of their mathematical (Fishwick 1995; Sokolowski and Banks 2010) or logical representations (Woolridge & Jennings 1994). We combine these characteristics to describe models as follows:

- S*tatic* and *dynamic* time representation characteristics refer to the dependency between the progression of the model and the advancement of time (Birta and Arbez 2007; Law 2007; Ljung and Glad 1994; Sulistio, Yeo, and Buyya 2004). In a static model, the model's state does not depend upon a representation of time; whereas, a dynamic model's states are dependent upon the advancement of time.
- *Discrete* and *continuous* bases of value characteristics refer to the change in values that variables can take (Chung 2003; Law 2007; Tolk 2012). A discrete model produces variable values at specific points during the model's execution; whereas, a continuous model can produce values for the variables at any point during the model's execution (Bennett 1995; Sokolowski and Banks 2012). Therefore, discrete models can be envisioned as producing a finite number of values over a specified range while continuous models produce an infinite number of values over a specified range (Aburdene 1988; Sulistio, Yeo, and Buyya 2004).
- D*eterministic* and *stochastic* behavioral characteristics refer to the notion of uncertainty and randomness (Bennett 1995; Law 2007). A deterministic model always produces the same output for a given input and system state (Aburdene 1988). A stochastic model produces potentially many outcomes for a given input and system state; therefore, a specific output cannot be known with certainty in advance of running the model (Ljung and Glad 1994; North and Macal 2007; Sulistio, Yeo, and Buyya 2004; Tolk 2012).
- S*erial* and *parallel* execution characteristics refer to the execution of the model (Law 2007). A serial model is generally executed on a single processor and the simulation execution proceeds sequentially. Simultaneous events can still occur but each event is calculated in a sequential order (Fishwick 1995). Generally, a model constructed for serial execution cannot be executed in parallel fashion (Sulistio, Yeo, and Buyya 2004). Parallel models are executed over multiple processors either within a single computer or distributed across multiple computers (Fishwick 1995; Law 2007).
- M*athematical* and *logical* expression characteristics refer to the notional development of the model (Sokolowski and Banks 2012). Notional expressions help in expressing the structure of a language and establishes a "common ground" for assigning truth to content (Féry and Krifka 2008 pg. 124). The assumptions associated with communication form the common ground through which the communication process is enhanced (Stalnaker 1977). The perception of the

modeler forms the words and relationships that create the abstract representation of a system (Bennett 1995). Expressions (logical or mathematical) allow for the translation of an abstract representation into an executable simulation. Logical models capture causality and decision processes and result in defined sequences of events (Tolk 2012). Mathematical models use equations to provide quantitative or analytical representations of systems (Fiskwick 1995; Sokolowski and Banks 2010).

- A*tomic*, *aggregated*, and *composite* resolution characteristics refer to the level of detail, scale, or abstraction used by the model. Abstraction is the level of detail needed to construct the model assists in the modeling process by directing focus to features of the objects within the system being modeled that are relevant to addressing the problem (Fishwick 1995; Zeigler 1984). An atomic model cannot be decomposed into a smaller element by anything else within the model (Tolk et al. 2006). An aggregated model exists when a collection of individually represented components within a simulation are merged to form a higher level object (Tolk 2012). Composite models are comprised of elements at varied resolution levels. A composite simulation may consist of entities which can change resolution levels during a simulation run (Davis and Hillestad 1993). A model that contains multiple atomic level resolutions becomes a composite resolution model unless all of the atomic resolutions are the exact same atomic level. Figure 1 shows the adapted taxonomy.

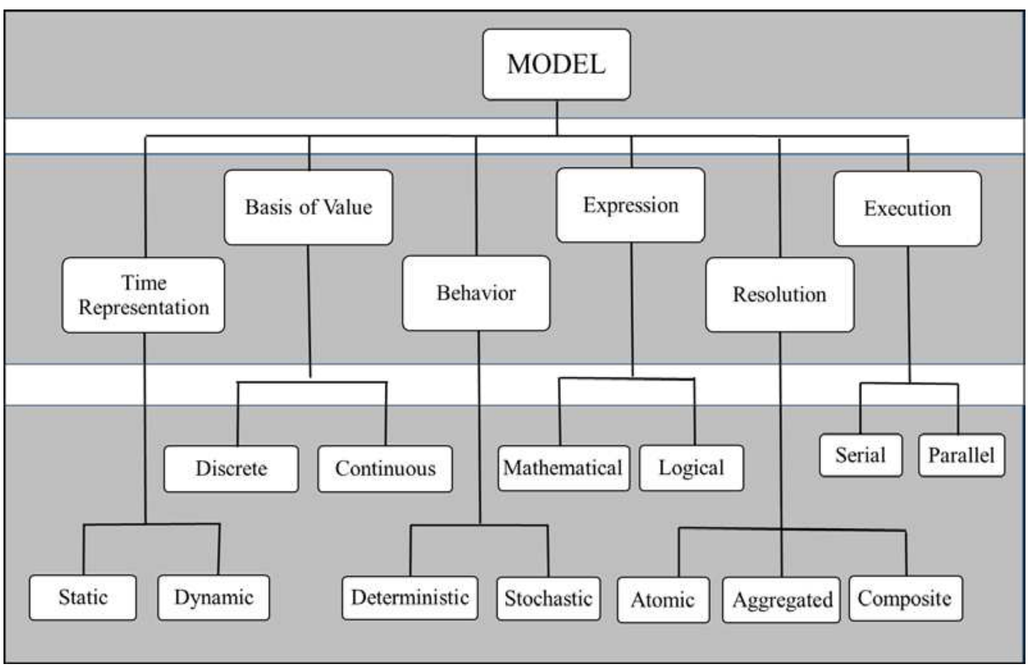

Figure 1: Taxonomy of Model Characteristics adapted from Sulistio, Yeo, and Buyya (2004). The characteristics of *time representation*, *basis of value*, *behavior*, *execution*, *expression*, and *resolution* provide a mutually exclusive description for categorizing models. The bottom-most level of this hierarchy provides an exhaustive description by further describing each of the six categories.

The addition of the expression and resolution categories to the four categories provided by Sulistio, Yeo, and Buyya's (2004) *simulation* and *simulation execution* taxonomies provides a mutually exclusive classification for describing models and the sub-categorization of each characteristic provides an exhaustive description of the model. The next section presents the classification of terms against the taxonomy.

3 CLASSIFYING THE M&S MODELING TERMINOLOGIES

We examine the literature to identify the characteristics that are explicitly mentioned with each of the modeling terminologies. Any characteristic that is referenced with respect to a terminology contains an "X" and any characteristic that is not specifically mentioned contains a blank space within Table 1. *Multimodels* capture multiple aspects of reality (resolution) (Fishwick and Zeigler 1992; Ören 1987; Yilmaz et al. 2007). Ören (1987) classifies multimodels in terms of continuous, discrete, and memoryless types; however, these classifications do not appear in the later multimodel taxonomy presented in (Yilmaz and Ören 2004; Yilmaz and Ören 2005) and basis of value is therefore not included. *Multi-paradigm models* deal with levels of abstraction (resolution) (Lorenz and Jost 2006; Vangheluwe, de Laura, and Mosterman 2002) as well as the characteristics included in the assumptions of the paradigm itself, such as differences in time representation, basis of value, behavior, and expression between the discrete-event simulation and system dynamics paradigms. *Hybrid* models contain feedback between simulation models (resolution) (Swinerd and McNaught 2012) and deal with continuous and discrete elements (basis of value) (Mosterman 1999). Multifacetted models consist of integrating multiple perspectives (resolution) to produce the whole picture of reality (Zeigler 1984; Zeigler and Ören 1986). Multi-resolution models allow for entities to be represented at different levels of resolution within the same simulation environment (Tolk 2012). *Multi-formalism models* can be grounded in predicate logic or mathematical theory (Balaban, Hester, and Diallo 2014) dealing with time representation, basis of value, and expression as well as handling parallel executions (Chow 1996). *Coupled models* consist of multiple models connected together (resolution) in a graph or network layout (Vangheluwe, de Laura, and Mosterman 2002). *Multi-method models* deal with implementing paradigms (therefore, contain the same characteristics as paradigms) and utilize couplings between models (resolution) (Balaban, Hester, and Diallo 2014). Composite models involve the use of multiple simulation methods or techniques (Viana et al. 2014) and contain the same characteristics as multi-method models. Table 1 provides a visual mapping of the terminologies to their characteristics.

Table 1: Mapping of model terminologies against the taxonomy of model characteristics. Column 1 provides the terminology. The remaining columns represent the characteristics from the taxonomy in Figure 1. Blank cells represent that the characteristic is not mentioned with respect to the terminology.

M&S Modeling Terminology	Time Representation	Basis of Value	Behavior	Expression	Execution	Resolution
Multimodel						X
Multi-paradigm Model	X	X	X	X		X
Hybrid Model		X				X
Multifacetted Model						X
Multi-resolution Model						X
Multi-formalism Model	X	X		X	X	X
Coupled Model						X
Multi-method Model	X	X	X	X		X
Composite Model	X	X	X	X		X

Resolution produces the greatest driver within the list of characteristics with all nine terminologies relying on it; therefore, a main motivator behind the use of multiple models, formalisms, or paradigms within a model appears to be the desire to capture multiple aspects of reality. A model fitting under any of these terminologies contains some combination of resolutions (atomic, composed, and aggregated), but does not necessarily contain all three at once. Multimodels, multifaceted models, multi-resolution models, and coupled models have the same characteristics and are only associated with resolution with the literature. Likewise, multi-method models, composite models, and multi-paradigm models have the same characteristics and are associated with all of the characteristics except execution. This leaves hybrid models and multi-formalism models in the position of having unique descriptions with respect to the characteristics. From the perspective of the characteristics, *execution* stands out as being the only one that maps to only a single terminology.

Due to the potentially many paradigms, models, or formalisms included within these models, there can be challenges in constructing and running the simulations due to competing child-characteristics within a single category of the taxonomy. For instance, multimodels allow for a system to be represented at multiple levels of resolution; however, this brings in the challenge of how to properly construct the model so that the simulation runs correctly when involving a combination of atomic, composed, and aggregated resolutions. We explore the challenges that arise with respect to each of the model characteristics in the following section.

4 CHALLENGES WITH SIMULATIONS CONTAINING COMPETING MODELING CHARACTERISTICS

The use of multiple modeling paradigms and models results in a number of challenges for simulations. Each category within the taxonomy provides a number of challenges to consider when constructing a model using any of these terminologies as there exists the possibility of having competing characteristics within the model. These challenges make the verification process crucial to the development of the model since the model's components can exist in potentially many specifications that need to be checked to ensure that they are correct regarding each formalism. A potential consequence of not verifying a multi-paradigm model is that an error produced during model validation may be a result of (1) a conceptual error (i.e. something missing from the model structure or model parameters) or (2) an implementation error (i.e. something is missing from the simulation that is supposed to be there). Without conducting verification first, the process of identifying whether the error was conceptual or implementation related is much harder to determine. Thinking that the model is programmed correctly when analyzing an unexpected outcome from a multi-paradigm model can lead to the conclusion that the new behavior is a result of the multi-paradigm model when the unexpected outcome is really caused by an implementation error. The following six items explore the challenges that pertain to each of the model characteristics within the taxonomy as well as challenges involving the use of more than one model.

- ***Challenges with Multiple Time Representations:*** Combining static and dynamic time representations requires determining how to integrate a dynamic time models with static time or event-driven models. Standards such as the HLA exist to deal with issues of time management (IEEE 1515-2010). Fujimoto and Weatherly (1996) identify that the simulations within a federation may deal with varied event ordering requirements, time flow mechanisms, real and scaled time data, and combinations thereof. The synchronization of events must account for differences in the use of time steps and events between simulations. Two simulations running in parallel may consider the use of a least common denominator in time-steps to ensure synchronization of time. The combination of continuous and discrete time requires a mapping of real numbers and integer numbers. The set of real numbers is uncountable whereas the set of integer numbers is countable; therefore, the set of integers used by a discrete time simulation is able to map all of its possible values to the set of real numbers that the continuous time simulation uses; however, this relationship does not work in reverse (Hein 2010). For simulations running at different time scales, there must be a way to ensure that the faster simulation can recover from changes in the slower simulation that occurs after the fast simulation has already passed that time step.

- ***Challenges with Multiple Bases of Values:*** Multiple bases of values provide challenges with respect to cardinality and computability. Bijective functions provide a means for observing the input and output relationship in a model. A function or model is bijective if there is a one-to-one mapping from the input set to the output set (Hein 2010). A countable set has a bijective relation to a subset of the natural numbers and a countable infinite set is bijective to the set of natural numbers. The union of countable sets is also countable; however, if both sets were bijective the union of the sets may be surjective (Hein 2010). Uncountable sets are not computable. Computability deals with the computational complexity or the amount of real time required to solve a problem. Issues involving computability deal with the ability to solve problems in polynomial (P) time using a one-tape deterministic Turing Machine or to solve problems in polynomial time on a nondeterministic (NP) Turing Machine (Karp 1972). Combining a model that is not solvable in non-deterministic polynomial (NP) time with a model that is solvable in P results in a NP problem.
- ***Challenges with Multiple Behaviors:*** Combining multiple behaviors in a model results in challenges of composition pertaining to the injective, bijective, and surjective nature of the models. Every model has a set of possible input values (domain) and a set of possible outputs values (codomain), in the same fashion as a function (Hein 2010). MPM produces a feedback setup between paradigms and as a result the output of a model (i.e. *codomain* M_A) can replace the input set of another model (i.e. *domain* M_B). The set of values comprising *codomain* M_A must match or be a subset of *domain* M_B or there will be an error in consistency for model M_B. If the set of input values is only a subset of the total input set that the model can have, then the input relationship may result in an injective model which restricts the total set of outputs that the model can produce. This effect can potential cascade through all of the models and alter the injective, bijective, and surjective functions of each model. Constraints may need to be added to explicitly reinitialize state variables when they are functions of the final values from another configuration (Mosterman 1999). Inconsistent units of measure between models can lead to contradictory outcomes if the conversion functions are not accounted for properly. Additionally, combining models with multiple behaviors can disrupt the homomorphism relationship of the model. Homomorphic functions can preserve the behaviors of the reference system by mapping lower level models to higher level models (Fishwick 1995).
- ***Challenges with Multiple Expressions**:* The combination of logical and mathematical expressions, specifically when the combination contains both discrete and continuous elements, can result in a very large state space of possible state changes within the model (Mosterman and Vangheluwe 2004). This can potentially lead to an issue in computability and cardinality (refer to challenges with multiple bases of value). Additionally, the works of Mosterman (1999) and Mosterman and Vangheluwe (2004) identify four potential issues that result from combining executable formalisms. *Event detection and location* deals with continuous variables that cause events to occur once they cross over specific thresholds and both the time of occurrence of the cross and the level of the threshold needs to be detected (Mosterman 1999; Mosterman and Vangheluwe 2004). *Sequences of discrete transitions* deals with known events that will occur due to time reaching a specific value and can be planned for in advance to help maintain synchronization of the simulation (Mosterman 1999; Mosterman and Vangheluwe 2004). *Consistent semantics of formalisms* deals with maintain consistency of meaning of elements contained within multiple formalisms to ensure that the formalisms communicate properly (Mosterman and Vangheluwe 2004). *Sensitivity to initial conditions* deals with the input

parameters of the model being sensitive to alterations when combining formalisms (refer to challenges with multiple behaviors for a discussion of model composition effects) (Mosterman and Vangheluwe 2004).

- ***Challenges with Multiple Executions:*** Multiple models executed in series need to be configured in a manner that events occur sequentially between all of the models. The field of Parallel and Distributed Simulation (PADS) deals with challenges in executing simulation in parallel. Fujimoto (1999) identifies several challenges with parallel simulation, including synchronization, local causality constraint (running a simulation in parallel should produce the exact results as running the simulation sequentially), and increased memory requirements for maintaining synchronization of the simulation. The crucial component of running a simulation in parallel or distributed over multiple computers is ensuring that all of the events within the simulation execute in the correct order (Fishwick 1995; Law 2007). Conservative and optimistic synchronizations seek to prevent violations of the local causality constraint or to provide a method for recovery to handle violations of the local causality constraint (Fujimoto and Weatherly 1996).
- ***Challenges with Multiple Resolutions:*** Some paradigms and their associated formalisms exist at specific levels of resolution, such as System Dynamics taking a high level view of a system while ABM takes a low level view of a system. Davis and Hillestad's (1993) work identifies a number of issues pertaining to multiple resolutions. Do the assumptions and operations hold across all levels of resolution? Is the representation of time maintained across all resolutions? Are spatial representations maintained across levels of resolution? Are aggregation and disaggregation relationships maintained? When combining models at different levels of resolution, a common information exchange model can establish a common view of entities and properties of the problem. In order to establish a common exchange between models, the higher resolution models need to aggregate their views or lower resolution models need to disaggregate their views (Tolk 2012). Inconsistencies can occur when transitions occur for an entity across varying levels of resolution such as a leading an entity into a state that it could not have reached through the normal time span of the model due to transitions between resolutions (Reynolds, Natrajan, and Srinivasan 1997). Challenges with multiple resolutions also occur when all of the models use the same resolution level. An example involves the use of ABM agents and DES entities where the model requires that the agents move through a DES process (Borshchev 2013). This requires that the agents and entities have semantic and syntactic compatibility to enable correct movement between the model components.
- ***Challenges with Multiple Models:*** Running multiple models in series results in increased time required to generate results. However, this time may not scale linearly as removing repeated functions between the models can serve to reduce the run time of the overall model (Mosterman 1999). Constructing a multi-paradigm model using multiple models involves challenges of communication between models and falls within the domain of interoperability. The models need to share relevant information and use the shared information (Diallo, Padilla, and Tolk 2010). A common goal of both interoperability and MPM is to achieve effectiveness, efficiency, and correctness and timeliness of exchanged information between systems (Tolk 2012). *Effectiveness* is achieved when all of the exchanged information is delivered to the correct simulation elements. *Efficiency* is achieved when only the required information is delivered to the target simulation element. *Correctness and timeliness* are achieved when the delivery of the information occurs at the correct time. Additionally, there can also be challenges pertaining to polymorphism (simulations interpret the same information differently) and encapsulation (hiding information within the simulation) pertaining to issues of data misalignment and misrepresentation (Diallo, Padilla, and Tolk 2010). Overall, all of the models need to maintain consistency and be non-contradictory with respect to each other and the reference system.

These challenges represent some of the main roadblocks that may occur in constructing and executing simulations that contain competing characteristics due to the use of multiple models. The following section presents areas for future work based on these challenges.

5 CONCLUSION AND FUTURE WORK

We construct a taxonomy to describe modeling terminologies with respect to the characteristics of their models. Interestingly, we find that none of the modeling terminologies explicitly deal with all six categories of model characteristics, that all of the terminologies deal with resolution, and that several of the terminologies only care about multiple resolutions. While multimodeling, MPM, hybrid modeling, multifacetted modeling, multi-resolution modeling, multi-formalism modeling, coupled modeling, multi-method modeling, and composite modeling are used by the M&S community, the challenges with building and verifying these models have not been addressed by the literature in an in-depth manner. The M&S community needs a verification framework to assist modelers in ensuring that the challenges associated with competing model characteristics do not cause errors within their simulations. This framework should be generalizable so that it can be applied to the models based on the model's characteristics which would allow for the framework to be applicable to any of the modeling terminologies. This taxonomy can potentially be applied to a problem during the conceptual modeling phase of the project in order to identify the characteristics that are needed to answer the MQ. This can help to identify the paradigms or types of models needed to address the problem.

Future works involves extending this research to tie the taxonomy of model characteristics into the simulation design phase in order to (1) guide modelers in the process of selecting paradigms or formalisms to use in constructing simulations that use multiple models, (2) illuminate possible challenges that may arise in constructing the simulation, (3) help in selecting the best tool to use for implementing the simulation, and (4) assist in verifying the simulation. Additionally, future work involves exploring different modeling formalisms and classifying them with respect to the taxonomy to provide another option for determining how to handle simulation implementation based on the desired characteristics for a simulation.

REFERENCES

Aburdene, M. 1988. *Computer Simulation of Dynamic Systems*. Dubuque, IA: Wm. C. Brown Publishers.

Bailey, K. 1994. *Typologies And Taxonomies: An Introduction To Classification Techniques* (Sage University Paper series no. 07-102). Thousand Oaks, CA: Sage.

Balaban, M., Hester, P., and Diallo, S. 2014. "Towards a Theory of Multi-method M&S Approach: Part I." In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Diallo, I. Ryzhov, L. Yilmaz, S. Buckley, and J. Miller, 1652-1663. Piscataway, NJ: IEEE Press.

Bennett, B. 1995. *Simulation Fundamentals*. New York, NY: Prentice Hall International.

Birta, L, and Arbez, G. 2007. *Modeling And Simulation: Exploring Dynamic System Behavior* (2nd Ed.). New York, NY: Springer.

Borshchev, A. 2013. "Multi-method Modeling." In *Proceedings of the 2013 Winter Simulation Conference: Simulation: Making Decisions in a Complex World*, edited by R. Pasupathy, S. Kim, R. Hill, and M. Kuhl, 4089-4100. Piscataway, NJ: Institute of Electrical and Electronics Engineers, Inc.

Chow, A. 1996. "Parallel DEVS: A Parallel, Hierarchical, Modular Modeling Formalism and its Distributed Simulator." *TRANSACTIONS of the Society for Computer Simulation* 13(2): 55-68.

Chung, C. A. (Ed.). 2003. *Simulation Modeling Handbook: A Practical Approach*. New York, NY: CRC Press.

Davis, P. K., and Hillestad, R. 1993. "Families of Models that Cross Levels of Resolution: Issues for Design, Calibration and Management." In *Proceedings of the 25th Conference on Winter Simulation*, edited by G. Evans, M. Mollaghasemi, E. Russell, and W. Biles, 1003-1012. New York, NY: ACM.

Diallo, S. Y., Padilla, J. J., and Tolk, A. 2010. “Why is Interoperability Bad: Towards a Paradigm Shift in Simulation Composition.” In *Proceedings of the Fall Simulation Interoperability Workshop*, 20-24. Orlando, FL.

Féry, C. and Krifka, M. 2008. “Notional Distinctions, Ways of Expression.” In *Unity and Diversity Of Languages*, edited by P. van Sterkenburg, 123-136. Amsterdam: John Benjamins.

Fishwick, P. A., and Zeigler, B. P. 1992. “A Multimodel Methodology for Qualitative Model Engineering.” *ACM Transactions on Modeling and Computer Simulation* (*TOMACS*) 2(1): 52-81.

Fishwick, P. A. 1995. *Simulation Model Design And Execution: Building Digital Worlds*. Upper Saddle River, NJ: Prentice Hall PTR.

Fishwick, P. A. 1998. “A Taxonomy for Simulation Modeling based on Programming Language Principles.” *IIE Transactions*, *30*(9), 811-820.

Fujimoto, R. M., and Weatherly, R. M. 1996. “Time Management in the DoD High Level Architecture.” In *ACM SIGSIM Simulation Digest* 26(1): 60-67. Piscataway, NJ: IEEE Computer Society.

Fujimoto, R. M. 1999. “Parallel and Distributed Simulation.” In *Proceedings of the 31st Conference On Winter Simulation: Simulation---a Bridge to the Future-Volume 1*, edited by P. Farrington, H. Nembhard, D. Sturrock, and G. Evans, 122-131. New York, NY: ACM.

Hein, J. L. 2010. *Discrete Structures, Logic, And Computability*. Boston, MA: Jones and Bartlett Publishers.

IEEE Std 1516-2010. 2010. “IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) – Framework and Rules.” Piscataway, NJ: IEEE Computer Society Press.

Karp, R. M. 1972. *Reducibility Among Combinatorial Problems*, 85-103. Springer.

Kuhn, T. S. 1970. *The Structure Of Scientific Revolutions (International Encyclopedia Of Unified Science, Vol. 2, No. 2)*. Chicago, IL: University of Chicago Press.

Law, A. M., Kelton, W. D., and Kelton, W. D. 2007. *Simulation Modeling And Analysis*. 4th ed. New York, NY: McGraw-Hill.

Ljung, L., and Glad, T. 1994. *Modeling of Dynamic Systems*. Eaglewood Cliffs, NJ: Prentice-Hall, Inc.

Lorenz, T., and Jost, A. 2006. “Towards an Orientation Framework in Multi-paradigm Modeling.” In *Proceedings of the 24th International Conference of the System Dynamics Society*, 1-18.

Lynch, C., Padilla, J., Diallo, S., Sokolowski, J., and Banks, C. 2014. “A Multi-paradigm Modeling Framework for Modeling and Simulating Problem Situations.” In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Diallo, I. Ryzhov, L. Yilmaz, S. Buckley, and J. Miller, 1688-1699. Piscataway, NJ: IEEE Press.

Mackenzie, N., and Knipe, S. 2006. “Research Dilemmas: Paradigms, Methods and Methodology.” *Issues in Educational Research* 16(2): 193-205.

McGregor, S. L., and Murnane, J. A. 2010. “Paradigm, Methodology and Method: Intellectual Integrity in Consumer Scholarship.” *International Journal of Consumer Studies* 34(4): 419-427.

Mosterman, P. J. 1999. “An Overview of Hybrid Simulation Phenomena and their Support by Simulation Packages.” In *Proceedings of the Second International Workshop of Hybrid Systems: Computation and Control (Lecture Notes in Computer Science 1569)*, edited by F. Vaandrager and J. van Schuppen, 165-177. New York, NY: Springer.

Mosterman, P. J., and Vangheluwe, H. 2004. “Computer Automated Multi-paradigm Modeling: An Introduction.” *Simulation* 80(9): 433-450.

North, M. J., and Macal, C. M. 2007. *Managing Business Complexity: Discovering Strategic Solutions with Agent-based Modeling and Simulation*. New York, NY: Oxford University Press.

Ören, T. 1987. “Model Update: A Model Specification Formalism with a Generalized View of Discontinuity.” In *Proceedings of the 1987 Summer Computer Simulation Conference*, 689-694.

Reynolds Jr, P. F., Natrajan, A., and Srinivasan, S. 1997. “Consistency Maintenance in Multiresolution Simulation.” *ACM Transactions on Modeling and Computer Simulation (TOMACS)*7(3): 368-392.

Sokolowski, J. A., and Banks, C. M. 2009. *Modeling and Simulation for Analyzing Global Events*. Hoboken, NJ: John Wiley and Sons.
Sokolowski, J. A., and Banks, C. M. 2010. *Modeling and Simulation Fundamentals: Theoretical Underpinnings and Practical Domains*. Hoboken, NJ: John Wiley and Sons.
Sokolowski, J. A., and Banks, C. M. (Eds.). 2012. *Handbook of Real-world Applications in Modeling and Simulation* (Vol. 2). Hoboken, NJ: John Wiley and Sons.
Stalnaker, R. 1977. "Pragmatic Presuppositions." In Milton K. Munitz and Peter K. Unger (eds.), *Semantics and Philosophy*, 197–214. New York, NY: University Press.
Sulistio, A., Yeo, C. S., and Buyya, R. 2004. "A Taxonomy of Computer-based Simulations and its Mapping to Parallel and Distributed Systems Simulation Tools." *Software: Practice and Experience* 34(7): 653-673.
Swinerd, C., and McNaught, K. R. 2012. "Design Classes for Hybrid Simulations involving Agent-based and System Dynamics Models." *Simulation Modelling Practice and Theory* 25:118-133.
Tolk, A. (Ed.). 2012. *Engineering Principles of Combat Modeling and Distributed Simulation*. Hoboken, NJ: John Wiley and Sons, Inc.
Tolk, A., Turnitsa, C. D., Diallo, S. Y., and Winters, L. S. 2006. "Composable M&S Web Services for Net-centric Applications." *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology* 3(1): 27-44.
Vangheluwe, H., de Lara, J., and Mosterman, P. J. 2002. "An Introduction to Multi-paradigm Modelling and Simulation." In *Proceedings of the AIS'2002 Conference (AI, Simulation and Planning in High Autonomy Systems)*, 9-20. *Lisboa, Portugal*.
Viana, J., Brailsford, S. C., Harindra, V., and Harper, P. R. 2014. "Combining Discrete-event Simulation and System Dynamics in a Healthcare Setting: A Composite Model for Chlamydia Infection." *European Journal of Operational Research* 237(1): 196-206.
Villa, F. and Costanza, R. 2000. "Design of Multi-paradigm Integrating Modelling Tools for Ecological Research." *Environmental Modelling and Software* 15(2): 169-177.
Woolridge, M. and Jennings, N 1994. "Agent Theories, Architectures, and Languages: A Survey." In *Proceedings of the European Conference on Artificial Intelligence* (*ECAI'94) Workshop on Agent Theories, Architectures, and Languages*. Amsterdam, Netherlands, 1-32.
Yilmaz, L., Lim, A., Bowen, S., and Ören, T. 2007. "Requirements and Design Principles for Multisimulation with Multiresolution, Multistage Multimodels." In *Proceedings of the 2007 Spring Simulation Multiconference*, 823-832. Piscataway, NJ: IEEE.
Yilmaz, L. and Ören, T. 2004. "Dynamic Model Updating in a Simulation with Multimodels: A Taxonomy and a Generic Agent-based Architecture." In *Proceedings of the 2004 Summer Simulation Conference*, 1-6.
Yilmaz, L. and Ören, T. 2005. "Discrete-event Multimodels and their Agent-supported Activation and Update." In *Proceedings of the 2005 Spring Simulation Multiconference*, 63-72.
Yilmaz, L. and Tolk, A. 2008. "A Unifying Multimodel Taxonomy and Agent-supported Multisimulation Strategy for Decision-support." In *Intelligent Decision Making: An AI-Based Approach*, 193-226. Springer.
Zeigler, B. P. 1984. *Multifacetted Modelling and Discrete Event Simulation*. Orlando, FL: Academic Press Prof.
Zeigler, B. and Ören, T. 1986. "Multifacetted, Multiparadigm Modelling Perspectives: Tools for the 90s." In *Proceedings of the 18th Conference on Winter Simulation*, edited by J. Wilson, J. Henriksen, and S. Roberts, 708-712. New York, NY: ACM.

AUTHOR BIOGRAPHIES

CHRISTOPHER LYNCH is a Senior Project Scientist at the Virginia Modeling, Analysis and Simulation Center. He received his M.S. in Modeling and Simulation from Old Dominion University in 2012 and a B.S. in Electrical Engineering from Old Dominion University in 2011. He is currently pursuing a Ph.D. in M&S from Old Dominion University. His research interests include multi-paradigm modeling, conceptual modeling, and verification of simulation models. His email address is cjlynch@odu.edu.

SAIKOU DIALLO is a Research Assistant Professor at the Virginia Modeling, Analysis and Simulation Center at Old Dominion University. He received a B.S. in Computer Engineering and a M.S. and a Ph.D. in Modeling and Simulation from Old Dominion University. His email is sdiallo@odu.edu.