systems

MDPI

Article

MBSE with/out Simulation: State of the Art and Way Forward

Bernard P. Zeigler [1,*], Saurabh Mittal [2] and Mamadou Kaba Traore [3]

1 Co-Director of the Arizona Center for Integrative Modeling and Simulation (ACIMS), University of Arizona and Chief Scientist, RTSync Corp., 6909 W. Ray Road, Chandler, AZ 85226, USA
2 MITRE Corporation, 7515 Colshire Dr., McLean, VA 22102, USA; smittal@mitre.org
3 IMS CNRS UMR 5218, University of Bordeaux, 33405 Talence, France; traore@isima.fr
* Correspondence: zeigler@rtsync.com

Received: 11 October 2018; Accepted: 9 November 2018; Published: 15 November 2018

Abstract: The limitations of model-based support for engineering complex systems include limited capability to develop multifaceted models as well as their analysis with robust reliable simulation engines. Lack of such Modeling and Simulation (M&S) infrastructure leads to knowledge gaps in engineering such complex systems and these gaps appear as epistemological emergent behaviors. In response, an initiative is underway to bring Model-Based Systems Engineering (MBSE) closer together with model-based simulation developments. M&S represents a core capability and is needed to address today's complex, adaptive, systems of systems engineering challenges. This paper considers the problems raised by MBSE taken as a modeling activity without the support of full strength integrated simulation capability and the potential for, and possible forms of, closer integration between the two streams. An example of a system engineering application, an unmanned vehicle fleet providing emergency ambulance service, is examined as an application of the kind of multifaceted M&S methodology required to effectively deal with such systems.

Keywords: modeling and simulation; discrete event system specification; DEVS; model-based system engineering; MBSE; internet of things; IoT; cyber physical systems; CPS; complex adaptive systems of systems

---

1. Introduction

Model-based engineering originated in the 1970s and with the foundational Systems Theory providing means and methods to incorporate simulation as integral mechanism to understand the abstractions and conceptual alignment between various constituent parts/systems. A. Wayne Wymore's book [1] is generally acknowledged as the first formulation of Model-based System Engineering (MBSE). Quite fittingly, Wymore is also one of the early System Theorists and the theory in his book [2] proves the basis for the Discrete Event System Specification (DEVS) to be discussed here. With the advent of Information Technology in the late 1990s, new modeling notations emerged that helped develop IT-enabled systems using traditional systems engineering practices. With IT now woven in every fabric of society, IT-enabled systems have grown complex and unmanageable. These are commonly known as sociotechnical systems [3].

To describe this new class of super complex systems in a man-made world, labels such as System of Systems (SoS), Cyber-Physical Systems (CPS), Complex Adaptive Systems (CAS), and Cyber CAS (CyCAS) are used interchangeably [4,5]. All of them are multi-agent systems, i.e., have large number of agents, are contextualized in an interactive environment and manifest emergent behavior. The constituting agents are goal-oriented with incomplete information at any given moment and interact among themselves and with the environment. SoS is characterized by the constituent systems under independent operational and managerial control, geographical separation between the constituent systems and independent evolutionary roadmap. CAS is an SoS where constituent systems can be construed as agents that interact and adapt to the dynamic environment. Cyber CAS is a CAS that exist in a netcentric environment (for example, Internet) that incorporates human elements where distributed communication between the systems and various elements is facilitated by agreed upon standards and protocols. CPS is an SoS wherein the constituent physical and embedded systems are remotely controlled through the constituent cyber components.

Model-based Systems Engineering employs model-based practices to engineer IT-enabled systems. While they still can be created using MBSE practices, the usage of such systems is far from it. In the sociotechnical era of Internet of Things (IoT), wherein multiple domains (for example, cyber, physical, and computational across various societal sectors) are involved, experimenting with the model to understand the model's functionality and engineer the resulting complex system is a challenging task. The existing toolsets lack the needed simulation analysis and experimentation capabilities leading to epistemological emergent behaviors, which is a characteristic defining property of any complex system [6]. These emergent behaviors can be both positive and negative. The negative emergent behaviors lead to cascaded failures while positive emergent behaviors may be sustainable and improve the complex systems' overall function [7]. To overcome such limitations requires extending MBSE for human machine interaction analysis and resilient system design [8].

The model-based paradigm has been successfully adopted by many disciplines when it comes to traditional systems engineering. However, for complex systems engineering, it is still in infancy as the tool-set support for engineering complex systems is limited [9]. These limitations range from developing multifaceted models (including comprehensive, highly detailed simulations and highly informative, analytic simplifications) to providing equally capable underlying simulation engines. Lack of such M&S infrastructure leads to knowledge gaps in engineering such complex systems and these gaps appear as epistemological emergent behaviors [10]. Many domains and disciplines are successfully employing simulation capabilities within their domains [11]. Furthermore, strong support for developing enterprise system of systems using architecture frameworks [12] such as Department of Defense Architecture Framework (DoDAF), Unified Architecture Framework (UAF), etc. is now widely available. However, these are nowhere close to addressing the requirements for multifaceted model development and its underlying simulation and experimentation infrastructure.

For the purpose of a unifying label, we will consider the term Complex Adaptive System of Systems (CASOS). This term emphasizes three aspects: complex, adaptive and SoS of complex sociotechnical systems. These three are distinct characteristics that require unique infrastructure (both hardware and software) for implementation. An SoS may be complicated but not complex i.e., SoS may employ traditional systems engineering. The adaptive aspect is brought in the mix through agent embodiment, situatedness and learning. These are made available in sociotechnical systems through Artificial Intelligence, Machine learning, algorithms, or the presence of human-in-the-loop (that complement the system functionality with guidance and participatory roles). We consider a class of examples of such CASOS in Appendix A to illustrate the particular advances in M&S required to support their systems engineering design.

CASOS present challenges that cannot be easily tackled using MBSE nor classical modeling, simulation, and optimization techniques. Recent model-based system engineering has proved inadequate due to lack of a full-strength M&S computational substrate [5]. Modeling and Simulation (M&S) methodology has been evolving to provide increasing capability to help systems engineers develop models of CASOS [4,11,13]. Such simulation models support design and testing of mechanisms with learning capabilities to coordinate the interactions of the operationally and managerially independent components. The design of such systems presents challenges to the currently employed independent use of simplified models for formal verification or brute-force simulations which are severely limited in the range of conditions they can test. M&S of CASOS must have a usable modeling environment that facilitates model validation from the end-user and a robust simulation infrastructure that can be formally verified to ensure correct model execution. Together, they enable exhaustive parameter evaluation and advanced experimentation. Model-based methods which support traditional systems engineering need to be augmented with simulation-based methodologies to ensure they support complex systems engineering that integrate discrete and continuous systems for complex hybrid systems. CASOS engineering will not become possible unless undesired emergent behaviors are completely removed from a computational environment or are known a priori so that they can be knowledgeably eliminated. A computational simulation-based environment provides experimentation opportunities to validate a CASOS model, such that it becomes predictable and eventually useful [5]. Ultimately, this is realizable in a Live, Virtual and Constructive environment with robust simulation infrastructure and human-in-the-loop undertakings [14].

The task of integrating various simulators to perform together as a composite simulation, termed as co-simulation, involves weaving the time series behavior and data exchanges accurately, the failure of which will yield inaccurate simulation results. As elaborated by Mittal and Zeigler [15], in the absence of a generic approach, every such hybrid system would require a dedicated effort to build a co-simulation environment. Bringing various simulators together is much more than a typical software engineering integration exercise.

In the following sections, we start with some background in M&S theory and the Discrete Event System Specification (DEVS) formalism to lay the basis for discussion of multifaceted modeling and the associated co-simulation infrastructure. This sets the table for considering how MBSE, DEVS and CASOS may be unified. Then, a generic architecture and workflow are proposed for M&S working within MBSE. This leads to a more in-depth discussion of multiobjective, multiperspective, and multiresolution families of models supporting simulation capabilities and tools. An example of a system engineering application, an autonomous unmanned vehicle fleet providing emergency ambulance service, is examined as an application of the kind of multifaceted M&S methodology required to effectively deal with such systems.

The paper is organized as follows: Section 2 describes the developments in M&S using DEVS formalism and its application to Complex Adaptive Systems, Section 3 describes the concepts in multifaceted M&S that facilitate development of multiresolution, multiperspective and multiobjective modeling and simulation. Section 4 provides an overview of the State of the Art of simulation tools. Section 5 discusses the way forward for both MBSE *with*, and *without*, simulation, followed by conclusions in Section 6.

2. Developments in Modeling and Simulation and DEVS

A solution to the problem of developing models and associated simulation capabilities that is gaining increased acceptance is offered by the DEVS formalism with a holistic construct called the Modeling and Simulation Framework (MSF). Briefly summarized, the framework defines the entities and their relationships of the enterprise of M&S and includes the relation between detailed models and their abstractions [16]. The framework is based on mathematical systems theory and recognizes that the complexity of a model can be measured objectively by its resource usage in time and space relative to a particular simulator, or class of simulators. Furthermore, properties intrinsic to the model are often strongly correlated with complexity independently of the underlying simulator. Successful modeling can then be seen *as valid simplification*, i.e., reduction of complexity to enable a model to be executed on resource-limited simulators and, at the same time, creating *morphisms that preserve behavior and/or structural properties, at some level of resolution, and within some experimental frame of interest*. Indeed, according to the framework, there is always at least one pair of models involved, which are called the *base* and *lumped* models, in such a relationship.

The DEVS formalism is formulated within MSF and formally specifies the internal behavior of the system as well as macro behavior of the overall system due to its closure under coupling property. This robustness in both structural and behavioral description ensures that the unwanted holistic behaviors, also known as negative emergent behaviors are explicitly avoided, along with the guaranteed manifestation of the desired (or positive) emergent behaviors [6,13,17]. The DEVS super-formalism provides a foundation [15] that specifies an abstract simulation protocol between the model and the simulator [16]. Thus, a requirement for M&S of CAS is to employ the principles of the Parallel DEVS simulation protocol (as illustrated by the hybrid approach of Camus et al. [18], for example) to support the required robust co-simulation.

2.1. MBSE, DEVS and CAS: Towards Unification

A major thrust of the MBSE community calls for formalized models to replace documents as the fundamental building blocks of systems engineering [19] Such models support a host of model-intensive activities such as improved communication about system artifacts among stakeholders as well as strengthened testing and verification (both at design as well as at runtime). Practical implementation of this thrust demands that such models eventually support *all* the activities typically associated with the simulation discipline. However, as suggested, current MBSE formalisms stop well short of this capability. One approach to bridging this gap is to enable mappings to be defined that precisely specify simulation models that realize the models' behaviors. Taken to a logical limit, this approach entails building more capability into a MBSE formalism so that it eventually replicates all capabilities associated with traditional simulation methodology. Although there are attempts to achieve this goal [20,21], there are also fundamental reasons why it is not attainable [22,23]. An approach that we hypothesize to work is to tie MBSE models with informal but well documented links. Furthermore, as experience grows with such cross-linkages, it might eventually become feasible to formalize these associations. Another complimentary approach is model-based interactive storytelling (MBIS) that enhances MBSE and interactive storytelling to increase stakeholder participation in the systems engineering process especially involving participants from multiple disciplines, and eventually transdisciplines (see below) [24].

2.2. Architecture and Workflow for M&S Working within MBSE

Software defined systems must increasingly operate on large, time-varying, heterogeneous data. Big Data enables and requires that these systems perform across an enormous variety of operating conditions presenting engineers with multi-dimensional, hierarchical, uncertain and critical control and decision challenges requiring transdisciplinary systems engineering [24]. Recent work has begun to address these challenges. Kavak et al. [25] offer a structured modeling approach to produce agents or parts thereof directly from data that focuses on individual-level data to generate agent behavioral rules and parameter values. Generalizing from the approach that recently enabled the AlphaGo program to defeat the world's top ranked human Go player, Wang et al. [26] envision an AlphaGo-like computation platform to enable artificial systems to model and evaluate complex systems, and through the virtual-real system interaction, realize effective control and management over the complex systems.

The architecture of Figure 1 offers a generic workflow that supports Wang et al. [26] vision of AlphaGo-like computational strength for future M&S-based systems engineering and management. The Modeling and Simulation Framework [16], and, in particular, its system specification hierarchy for acquiring levels of knowledge about an observed system, provides a solid basis for inference of structures from the volumes of Big Data envisioned by Kavak et al. [25].

Figure 1's architecture and workflow for M&S offers a vision of model production for use in MBSE. The process starts with the development (or reuse) of a System Entity Structure (SES) that organizes a family of simulation models for the current application of interest [27]. SES is an ontology, a language with syntax and semantics to represent declarative knowledge [28]. The SES representation scheme structures the search for a subset of models that are of particular concern under criteria that relate essentially to their behavior and can't be defined in the first instance by their structural properties. Indeed, the behavior generated under simulation is observed within the Experimental Frame that characterizes the criteria defining the subset of interest. Roughly, an experimental frame (EF), as defined within the MSF, is a specification of the conditions under which the system is observed or experimented with. As such, experimental frames are the operational formulation of the criteria that motivate the M&S-based pursuit of the models of interest. The SES includes coupling information that directs the compositions of hierarchical models from components in the model-base. The combination of selection from specializations and aspects leads to a very high combinatorial search space. Since an SES describes a number of system configurations, the SES tree needs to be pruned to get one particular configuration, which is called Pruned Entity Structure (PES). Pruning operations factor out a particular model specification which can then be transformed automatically into a coupled model with components from the model base. Such components are either DEVS models or have been wrapped in a DEVS interface for DEVS compliance and amenability to the coupling specified by the SES. Simulation of such a model, eventually on a high-performance platform using parallel simulations of multiple models under testing for reasonable execution times, generates the behavior of the model and produces results in the experimental frame of interest. These results measure the extent to which the governing criteria are satisfied and are analyzed for guidance to direct the pruning towards a larger percentage of models that fully satisfy the criteria. At this point, Artificial Intelligence (AI) is useful to help analyze the results and predict which new prunings of the SES should performed at the next iteration. Built into the iteration loop is a second cycle of transition between *base* and *lumped* models where the *lumped* model can greatly accelerate the search for high-value models by enabling faster runs that provide useful information for the more detailed *base* model. Some fundamental distinctions between *base* and *lumped* models concern objectives, representation, entity attributes and variables, interaction processes, timing mechanisms, and computational complexity [16].

As illustrated in Figure 1, the architecture envisions a collaboration between human and AI agents. The human modeler develops the SES and the DEVS model base to span configuration space that encompasses the subset of interest. The AI agent, under control of the user, analyses the results and generates new prunings in order to increase the percentage of models of interest. Here, we lean on the agent to provide the grist for patterns that humans can discern and exploit with imagination, novel insights, and storytelling ability [24]. The modeler develops valid simplification morphisms for the DEVS *base* and *lumped* models and decides when and how to iterate between the levels of resolution in order to accelerate the overall process [29]. Here, a *base-lumped* pair of models refers to a pair of models—the first more "complex" than the second, which are equivalent in an experimental frame of interest in the system investigation [16]. Such equivalence allows the *lumped* model to stand in for the *base* model for the objectives underlying the frame. Thus, while a *lumped* model is not necessarily a uni-directional morphic projection of a *base* model, formalization and tool support of model-to-model transformations is a promising technology to support the workflow illustrated in Figure 1 [30].

To plumb these concepts in more depth, we turn towards multiresolution modeling methodology in the broader context of multifaceted modeling methodology [27].

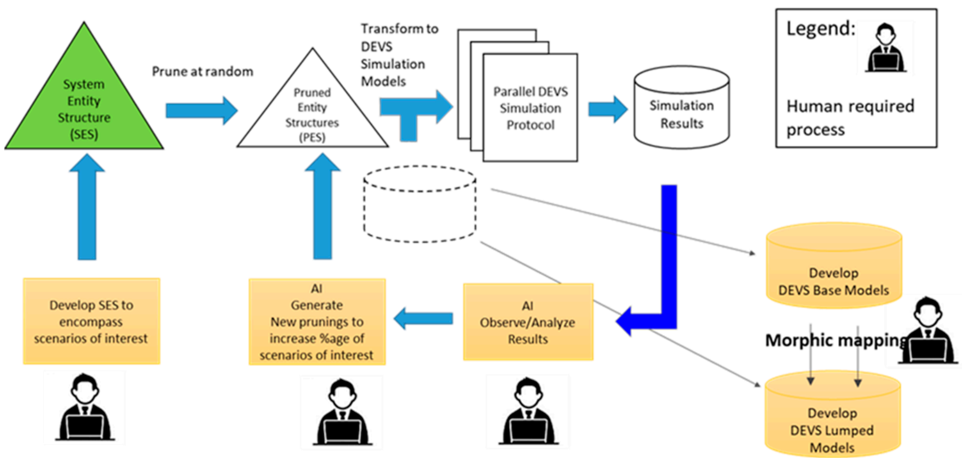

Figure 1. Architecture and Workflow for M&S working within MBSE.

3. Multifacetted Modeling and Simulation: Multiobjective, Multiperspective, Multiresolution Families of Models

Modeling and simulation are activities undertaken to support system engineering decision making—the ability to assess the effects of constructions and interventions before they are actually carried out and to pre-select promising ones, in view of the driving objectives. These objectives in turn serve to orient modeling efforts. We recognize that a CASOS may be subject to a multiplicity of system engineering objectives.

3.1. Multi-Perspective Families of Models

Multiple objectives require different levels of explanation be provided for the same system under study. Each level of explanation can be expressed within a dedicated EF, the one that provides answers to questions of interest from the perspective of its corresponding objective. Therefore, a family of perspective-specific models needs to be built.

While, in practice, each of these partial models is executed in isolation to provide a given level of explanation for the system of interest, all of them are related in reality, since they depict various abstractions of the same system, but from different perspectives (or facets). Therefore, while each perspective-specific model produces results within its corresponding experimental frame, partial models can be holistically integrated within a holistic experimental frame, and questions that are transversal to different perspectives can be accurately addressed, which is not possible in any of the perspective taken alone.

How do we identify the facets/perspectives in a general context and in a systematic way? We review a structured approach to building an ontology for the M&S of a domain of interest. The domain analysis ontology must provide a formal way to capture all the knowledge that might be in the range of M&S of the domain for which it is likely to be used. Therefore, it must capitalize on the abstractions used for the simulation of the entire targeted domain, beyond aspect specific modeling. Thus, the generic approach to domain analysis is a 4-layered ontology which highlights at each layer a generic key characteristic. As depicted by the SES presented in Figure 2, the following layers are defined:

- System level, where meaningful specializations of the class of systems that characterizes the domain of interest are highlighted,
- Facet level, where all cumulative aspects of a domain system are clearly separated,
- Scale level, where major spatial and temporal scales are emphasized, Model level, where conventional models often originating from decades of theoretical findings are identified as reusable artefacts to be selected and integrated in new studies.

The ***System level*** recognizes the whole complex system as a juxtaposition of multiple facets, while various specializations can be identified as possible instances of the same integrated set of facets in various specific contexts. For example, healthcare systems can be specialized into primary, secondary, ternary, and home care [31], while transportation systems can be specialized into air, ground, rail, and aquatic transport, and military systems can be specialized into air, ground, and marine forces.

The ***Facet level*** establishes three generic facets, i.e., “production facet”, “consumption facet” and “coordination facet”. Although the identification of a system’s facets may depend on the domain as well as the experts involved and the objectives in mind, we suggest the systematic adoption of these generic patterns. In other words, a complex system is made up of one or various facets, each of which being a production system (hence, leading to a ProF model), a consumption system (that gives a ConF model), or a coordinating system between production and consumption (giving a CooF model). These patterns encompass the traditional supply-demand duality that often characterizes complex systems [31]. The notion of “Production” encompasses the notion of “Supply” in that it involves not only the intentional supply of services needed, but all phenomena that produce positive and negative impacts on the system’s stakeholders. Examples of production in healthcare include vaccination and information diffusion (as production of ease), but also contamination and epidemics (as production of disease). Examples in transportation include the production of public or private transportation services, but also the production of air pollution, land use, and accidents. Examples in military systems include the production of security and protection, but also the production of life and infrastructure destruction. Similarly, the notion of "Consumption" encompasses the notion of "Demand", as consumers may not be only users seeking intentionally for services but all stakeholders that consume what is produced by the system. Examples of consumers are population, patients, travelers, pedestrians, territories, enemies, etc. An important element of this multi-perspective approach is that, while perspectives have mutual influence on each other, each perspective captures its received influences by means of parameters, which values explicitly reflect implicit assumptions and simplifications made about other perspectives influences. For example, when focusing on the system as a production system, the ProF model will make use of parameters (such as the arrival rate of patients in a hospital, or the arrival rate of travelers in a shuttle, or the arrival rate of enemies in a combat theatre) to aggregate all processes going on in the same system when taken as a Consumption system. In its turn, the ConF model will make use of parameters (such as the death rate of a population) to aggregate all processes going on in the same system when considered as a Production system. Coordination can be seen as cross-organization mechanism managing the entities and resources of existing ones, such that individual goals as well as system-wide goals are satisfied. It is needed to the extent that specific interaction and cooperation are required to ensure safe entangled and context-dependent behaviors.

The ***Scale level*** emphasizes on that a characteristic feature of complex systems is the occurrence of interactions between heterogeneous components at different spatial and temporal scales with various interpretations of the notion of scale, and a major concern about scale transfer processes where inter-scale interactions must be properly described, as emphasized in [31]. A scale refers to a set of relationships, which implicitly (or explicitly) point to spatial properties (such as location, shape, size, etc.), as well as temporal properties (such as exact or approximate timing, simultaneity or sequentiality, continuity, etc.). Thresholds between scales are critical points along the scale continuum where a shift in the importance of variables influencing a process occurs. As a result, the generic ontology proposed exhibits macro, meso and micro levels of abstraction both within the consumption and the production facets, leading respectively to the generic MaConF, MeConF, MiConF, MaProF, MeProF and MiProF models. The macro-meso-micro architecture is recognized as describing the three possible levels of inquiry on which social scientific investigations might be based [32].

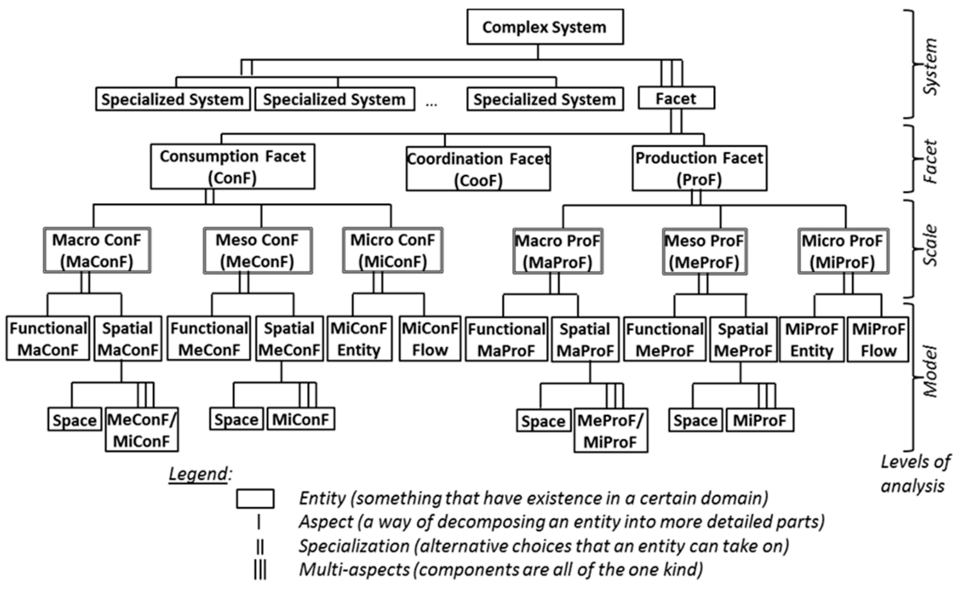

Figure 2. Generic SES for Multi-perspective Modeling.

The ***Model level*** identifies conventional models often originating from decades of theoretical findings as reusable artefacts to be selected and integrated in new M&S studies of complex systems. It defines the abstractions that can be directly simulated, by distinguishing four generic types of model, i.e., entity models, flow models, functional models and spatial models. While Entity models describe autonomous individuals with specific attributes and with or without goal-driven behavior, functional models are formulated as mathematical equations, spatial models are composed of individuals geographically located in a space model, and flow models capture scenarios an individual can undergo. These models explicitly describe the temporal and spatial properties pointed out (implicitly or explicitly) at the scale level. Consequently, the generic ontology has in each facet, entity and flow models at the micro level of abstraction, and functional and spatial models at the macro and meso levels of abstraction. It is worth noting that the fact that a spatial model at any macro level involves a space model that contains abstractions detailed at lower levels (i.e., meso and micro), and that, similarly, a spatial model at any meso level involves a space model that contains abstractions detailed at the micro level.

The generic ontology is meant to be instantiated in the analysis of any new domain of interest in view of its M&S. Such an instantiation provides the domain-specific ontology that will drive the multi-perspective modeling and holistic simulation (MPM&HS) process of the targeted domain.

3.2. Multi-Resolution Families of Models

Multi-resolution modeling (MRM) is essential for exploratory analysis of CASOS design spaces because it is neither cognitively nor computationally possible to keep track of all relevant variables and causal relationships [33]. A typical multiresolution scenario applicable to defense investigates the operational differences between low-level military entities such as individual tanks and the aggregated high-level units, e.g., battalions or *platoons* when moving in a battlefield. Attributes of an aggregated entity like a tank battalion are often determined by applying an aggregation mapping to the attributes of its individual entities. The mapping can group a set of tanks to a single tank battalion together with a function to derive holistic attribute values, e.g., an average speed of a tank battalion, from the constituent individual tank speeds (disaggregation is the inverse mapping). Here, the *base* model is typically "more capable" and requires more resources for interpretation and simulation than the *lumped* model. By the term "more capable", we mean that the *base* model is valid within a larger set of experimental frames (with respect to a real system) than the *lumped* model. Here, we note that the terms "*base*" and "*lumped*" are terms employed with the framework to denote the full range of possible pairs of models in which the first is more capable (e.g., more detailed, disaggregated, high resolution, fine grained) than the second (less detailed, aggregated, low resolution, coarse grained). We note that MRM sometimes refers to simulation environments in which entities are aggregated or disaggregated at runtime based on certain triggering conditions, in which case, resolution changes dynamically at runtime. Such dynamic structure models are included within the larger category of families of models structured at different resolutions meeting different modeling objectives.

The scope/resolution/interaction product is limited by the computational resources at hand. Thus, typically a large scope places constraint on the resolution and interaction that can be represented, whereas a smaller scope will permit higher resolution and interaction. Nevertheless, the trend towards high fidelity M&S implies increasing all factors in the scope/resolution/product (as in the *base* model) while lumping can reduce both scope and/or resolution independently.

Some typical distinctions often drawn between *base* and *lumped* models with respect to agent modeling are presented in Table 1 [33,34].

Table 1. Some fundamental distinctions between *base* and *lumped* models (in a military simulation example).

	Base Model	Lumped Model
Objectives	Results traceable to specific performance data and assumptions. Evaluate subtle differences in weapons, sensors, or tactics, Understand how different inputs affect combat performance	Predict overall results Include small numbers of parameters Parameter values amenable to identification from feasibly obtainable data
Representation	Individual agents as separate entities	Aggregate entities into groups typically respecting command hierarchy.
Entity Attributes and Variables	Location in space and time, position in social or other hierarchies, perception of the situation: threats and opportunities, capabilities, etc. updated at event occurrences or time steps	Averaged entity values attributed to groups, Discrete events compounded into rates for groups, Global state sets, cross-products of individual state sets
Interaction processes	Decomposed into sequences of events and activities, Tracking of individual behaviors	Processes aggregated into group level formulae abstracting individual behavior
Timing mechanisms	Coordinate the event sequences for the numerous participants so that subtle interaction patterns can be modeled	Micro stochastic sequences can be aggregated into macro behaviors using law of large numbers expressed more simply in stochastic or deterministic form
Computational Complexity (Scope/resolution/ interaction product)	Lean towards large scope, high resolution and unconstrained interaction	Lean towards smaller scope, low resolution and constrained interaction

However, the important point is that within a particular experimental frame of interest the *lumped* model might be just as valid as the *base* model. Furthermore, the trade-off between performance and accuracy [35] is a fundamental consideration where performance refers to the computational resources used in a simulation run and accuracy refers to the validity of a model with respect to a referent system within an experimental frame [16]. Use of computational resources tied to a simulator's time and space demands in generating the model's behavior are correlated with its scope/resolution product where scope refers to how much of the real world is represented, resolution refers to the number of variables in the model and their precision/granularity.

Models should be built in an incremental manner with continuous engagement and validation from the subject matter experts (SMEs) and their mapping to the experimental frames. This pairing allows the selection of the correct resolution of the model. This allows the development of early insight into the objective of the modeling, provides a holistic view of the system under study, provides a testing framework for the target, explains the target's behavior early in the M&S development cycle and serves as the foundation to add more complexity to the subsequent models. Parallel development of the experimental frames provides the experimentation and data collection requirements for the model and the computational requirements of the underlying simulation platform.

The incremental manner from lower complexity to higher complexity works when one tries to build models from top-down. However, there may be situations that in high fidelity models are already present and one needs to bring them together to develop a larger SoS model. This includes the incorporation of legacy models and simulators. In that situation, lumping needs to be carefully managed as information is lost in aggregation if not justified by valid abstractions such as from the underlying fundamental science (e.g., physics, chemistry, etc.). This is then further supported by systems morphisms and homomorphisms to ensure that there is continued correspondence between the model family (containing *base*, *lumped* and higher fidelity) and the model hierarchy. Figure 3 provides an illustration of the construction of such a model family as built up from *base*/*lumped* model pairs. For example, a *base* model may be composed of multiple components each of which can be lumped into simplified lumps and coupled together constitute a new *lumped* model. The *base* model might itself serve as a *lumped* component of a larger model leading to a hierarchical construction. Moreover, a *lumped* model may itself serve as a *base* model for an abstraction that supports reduced resolution. Thus, working together, resolution and composition operations can create a multi-resolution family of simulation models.

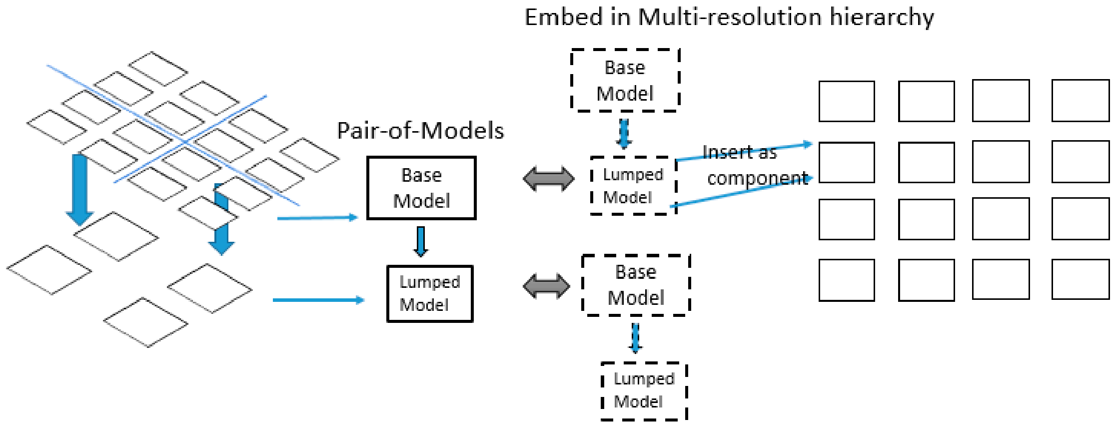

Figure 3. Relationship between the *base* and the *lumped* models within the model family.

A methodology for constructing a multiresolution family of models is illustrated in Figure 4, as follows: given requirements and constraints of the problem, consider a model that satisfies all such requirements and constraints as the *base* model to be aimed for. Create *lumped* models by making assumptions about the *base* model, including relaxing of constraints and dropping of requirements. Create higher resolution models by removing assumptions that were previously added while including more refined representations to address the affected constraints and requirements, meanwhile checking for consistency of predictions between related *base* and *lumped* models. The targeted *base* model is the one achieved when all assumptions that have been made are removed in this iterative process.

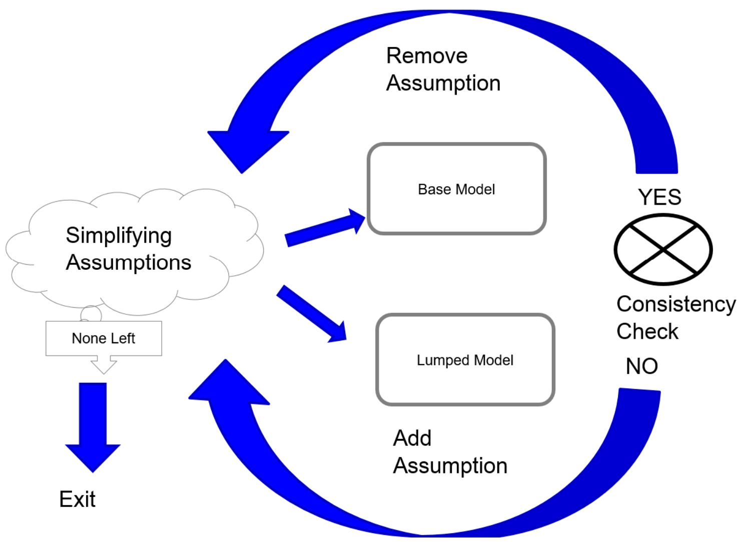

Figure 4. Development of *lumped* models via assumption addition/removal.

Appendix A discusses this methodology in an example of a system engineering application, an autonomous unmanned vehicle fleet providing emergency ambulance service.

4. Simulation Capabilities and Tools

Many of the enterprise architecture (EA) modeling tools (for example, Sparx Enterprise Architect, NoMagic MagicDraw/Cameo, IBM Rational/Raphsody) adhere to the Unified Modeling Language

(UML), Systems Modeling Language (SysML), Business Process Modeling Notation (BPMN), DoDAF and UAF specifications [12]. Much work has gone into developing the modeling infrastructure so that the multi-domain knowledge can be centrally managed and shared across all the stakeholders. However, the simulation capability required to experience and experiment with the dynamic behavior of such a model is quite rudimentary as the modeling capability still maps to traditional systems engineering practices. This further raises questions on the adequacy of widely understood modeling notations such as UML, SySML and BPMN for developing a complex adaptive systems model and the simulation infrastructure required for experimentation with such a model [5]. Consequently, much of the work done in computational analysis of system of systems M&S is done in the academic and research community which leads the development of the accompanying infrastructure employing High Performance Computing (HPC) and cloud environments. The SES/DEVS methodology has been employed in a wide range of modeling and simulation applications for a range of complex systems (known by various other labels such as adaptive systems, system of systems, complex adaptive systems, networked system of systems, or a combination of these) through simulation and experimentation.

5. Discussion and Way Forward

Two arguments need to be made: MBSE without simulation and MBSE with simulation. Before making the arguments, we must state unequivocally that modeling and simulation are distinct activities. Modeling facilitates understanding of phenomena (both natural and artificial) and helps develop an understanding (both personal and shared). This understanding when coupled with traditional systems engineering practices gave way to the development of MBSE in its current state. Simulation subsumes modeling, i.e., simulation is operational only when there exists a model to execute on a platform (e.g., mental, collaborative, computational). This execution affords experimentation with the model and provides opportunities to experience the "model" in various settings (for example, Live, Virtual and Constructive environments [14].

MBSE without simulation, henceforth, involves effort spent in the development of the model. In SoS, CAS, CASOS, etc. settings, due to a large number of stakeholders, this activity takes on a whole new meaning where developing a shared understanding is an achievement in itself. IT-enabled modeling environments commercially available provide the needed centralized repository and model editing environments to facilitate model development. The prime objective of this activity is to bring the stakeholders on the same page. In this regard, MBIS can exploit the immersive powers of storytelling to convey an evolving system design and concept of operations to technically unsophisticated stakeholders [24].

Between the MBSE without simulation and MBSE with simulation is the realm of executable models. Formal methods are applied in this model, which lead to software implementation. This enables testing and verification of systems under investigation during the model runtime. While they are not supported by experimentation infrastructure, indeed they do allow experience with the system under study.

MBSE with simulation affords experimentation and experience with the model. Simulation engineering requires an advanced computer science theory, methods and techniques to provide a computational substrate for the model to execute. When simulation engineering is coupled with systems theory to develop the computational platform, we get a composable simulation platform. DEVS is such an example. In SoS, CAS, CASOS, etc. settings, the computational platform becomes an explicit engineering exercise as new domains are brought in the simulation environment [15]. The prime objective of this activity is to experiment with the model and gain experience in understanding model's behavior. Combining M&S with MBIS can enhance the ability of models in virtual worlds to foster discovery of previously unknown interactions and dependencies among system elements and between the system and the environment [24].

Moving forward, as long as one adheres to the primary objectives, both MBSE without simulation and MBSE with simulation are worthy efforts that employ the art of modeling and the simulation technology to develop an abstract understanding of the system under consideration and experiment with it. Current MBSE practices must extend to incorporate both multiresolution and multi-perspective modeling within a holistic approach to contribute to complex systems engineering. MBSE, even with simulation, is inadequate to support complex systems engineering. Complexity Science principles incorporating concepts like nonlinearity, emergent behavior, network connectivity, etc. is being brought in to augment MBSE practices with DEVS [17,36] and efforts are underway to develop a comprehensive methodology for their application to next generation complex systems such as Internet of Things (IoT) and Cyber Physical Systems (CPS). SES and DEVS provide foundational theory and technology to engineer M&S-based complex systems in all of their different flavors (SoS, CAS, CASOS, etc.).

We examine these challenges as presentation of a way forward for DEVS M&S in the context of the roadmap [37] formulated by the International Council on System Engineering (INCOSE).

Along these lines, Zeigler, Mittal and Traoré [31] have identified strong requirements that must be satisfied to enable DEVS-based M&S to be practiced at its most productive level. Table 2 considers these developments in support of maturity levels 4 and 5 in Table 3 in the light of the multi-resolution methodology discussed above.

Table 2. DEVS-based M&S Developments to support maturity levels 4 and 5 of Table 3.

Development	**Description**
Deal with SoS nature	The framework provides a multiperspective methodology for developing coupled models of components from various formalisms capable of expressing the different perspectives needed for SoS together with holistic abstractions that support integration and coordination [17,38–41].
Develop an effective organizational ontology	DEVS-based M&S includes the macro level facets that properly organizes the CASOS domain and supports refinement into more detailed components at the meso and micro levels.
Enable the ontology to support combinatorial model compositions and exploration	The SES/MB (Model Base) supports hierarchical composition of the coupled model resulting from pruning that selects from the combinatorial family of possible compositions described by the SES. The DEVS formalism which can encompass models expressed in various formalisms typically found to be useful in simulation studies [38]. MBIS can enhance exploration of model behaviors in LVC settings [24].
Include the major facets major facets to ensure representation of all levels (macro, meso, and micro) of behavior,	The 4-layered ontology highlights at each layer a generic key characteristic. It capitalizes on the abstractions used for the simulation of the entire targeted domain.
Include a large spectrum of models for combinatorial composition	An example in healthcare [31] illustrates models spanning health diffusion, resource allocation, provider and provision modeling, population diffusion, spatial models including agent-based models at individual and higher level abstractions including coordination mechanisms.
Instrument the complex system to support on-going high quality data	The MSF includes experimental frames that can specify, collect and aggregate the information for higher levels in a multiperspective model. The simulation infrastructure guarantees correct execution of the composed model and the behaviors in a transparent manner [5,17].
Include pervasive incremental automated learning	A wide variety of mechanisms is available at different levels of abstraction and computational complexity with typical parallels drawn to biologically inspired learning and evolutionary processes including activity-based credit assignment, unsupervised techniques (e.g., clustering, rule mining) and reinforcement learning [42–45]. These can be based on the premise that new system states are being continuously captured in timely snapshots of data and added to an accumulated repository representing the system knowledge supporting iterative training employing updates in the system behavior.

Table 3. Way forward for DEVS M&S in relation to INCOSE Roadmap.

Maturity Level	INCOSE MBSE Roadmap	Correlation with DEVS M&S Development
1	Emerging MSBE Standards	DEVS model standard and DEVS simulation protocol standard
2	Matured MBSE methods and metrics: Integrated System (Hardware/Software) models	Experimental Frame representation of metrics, DEVS models for Integrated Systems
3	Architecture Models Integrated with Simulation, Analysis, and Visualization	DEVS framework for Architecture models and support for analysis and visualization
4	Defined MBSE theory, ontology and formalisms	See Table 2
5	Distributed and secure model repositories crossing multiple domains	See Table 2

6. Conclusions

As stated before, in the IoT and CPS era, the existing MBSE toolsets lack the needed simulation analysis and experimentation capabilities leading to epistemological emergent behaviors. Accordingly, there arises the goal of trying to manage the bad aspects of emergence while preserving its good qualities. This is reminiscent of Whitehead and Russell's [46] attempts to control self-reference in Formal Mathematics in the Principia Mathematica, which they eventually proved to be paradoxical in nature. Principia's solution of hierarchical set constructions may suggest a way forward in the SoS case and the DEVS formalism offers a ready-made solution for hierarchical model construction justified by closure under coupling [6,16].

MBSE in its current state is very much tied to traditional systems engineering and needs to be expanded to incorporate complex systems engineering practices. It seems clear that we need to get a better handle on the whole SoS life-cycle with a more deliberate combined MBSE/DEVS approach. This will help us focus on the problem and better understand the five attributes of SoS [47] that underlie and interact to induce emergence in a formally defined system with explicit coupling information [48].

Still, we should recognize enormous obstacles that must be overcome to achieve these visions. Progress may require new ways of thinking about systems that truly enable them to be developed with reusable components, eventually leading to composable M&S solutions. We must become able to identify the limitations in dealing with Big Data and limitations in dealing with its multi-dimensional, hierarchical, and uncertain nature. Here, we have considered the problems raised by MBSE taken as a modeling activity without the support of full strength integrated simulation capability and the potential for, and possible forms of, closer integration between M&S and MBSE as expressed in augmentation of the INCOSE roadmap for MBSE maturation with requirements for DEVS-based M&S evolution. Working to put the infrastructure in place to meet these requirements will move both systems and M&S communities along realistic paths towards realizing the INCOSE roadmap.

Author Contributions: All authors contributed equally to Conceptualization, Methodology, Writing—Original Draft Preparation, Writing—Review and Editing.

Conflicts of Interest: The authors declare no conflict of interest.

Disclaimer: The author's affiliation with the MITRE Corporation (McLean, VA, USA) is provided for identification purposes only and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions, or viewpoints expressed by the author. Approved for Public Release, Distribution Unlimited (Case: PR_18-2996-1).

Appendix A. Multifaceted Model Family Example

As suggested in the main text, the focus of our attention is on CASOS, complex adaptive systems of systems. Here, we briefly present an example to illustrate the emphasized attributes. Following Dahmann [3], consider the problem of first responders to catastrophic events providing emergency rescue and relief. Such a service requires coordination of multiple systems (so is an SoS) with complex interconnected networking and requires adaptation to the different challenges posed by different calls to service. Now, imagine that such a service is to be supported by a fleet of autonomous unmanned vehicles (UAV) responsible for all needed transportation tasks. We restrict the focus further to the system engineering design of such an SoS providing emergency ambulance service, as an application of the kind of multifaceted M&S methodology required to effectively deal with such a complex adaptive SoS (CASOS).

First, consider a short list of categories of objectives relevant to the design and implementation of a new ambulance system employing unmanned autonomous vehicles (UAVs) as outlined in Table A1. An all-inclusive model would be able to provide the basis for decision making in each of these categories. However, the impediments to constructing such a comprehensive model make it a near impossibility. Instead, we can envisage a collection of partial models, each oriented to one or more objectives [49].

The following is a subset of models that were developed to address the objectives 1–3, and 8–10 just listed:

- *UAVMotion* is a discrete event model representing the motion of the UAVs as agents in space employing only the kinetic parameters of the vehicles and the random space-time distribution of requests to get a first order prediction of the number required to meet the demand.
- *MarkovDutyCycleCTM* represents the duty cycle of a typical UAV as a Markov stochastic process with a small number of states representing its location as at the depot or in the service area and able or not to provide service.
- *Multiwork* represents the UAVs as individual servers in a discrete event model with a simple bidding protocol to coordinate response to incoming requests with the servers progressing through a duty cycle consistent with the Markov model.
- *Hierarchical Composition* elaborates on the Multiwork model by incorporating states of the vehicles (e.g., carrying patient) that bear upon speed of travel and available fuel. Hierarchical structure results from representation of UAV as itself a composite with components representing coordination protocols, kinetics, and fuel consumption.
- *Design for Adaptive Sustainment* is discussed later.

Table A1. Objectives relevant to system engineering of a UAV-based Emergency Ambulance Service.

#	Objectives	Models Needed
1	Determine travel and payload requirements for UAVs and personnel	Kinetic models of UAVs, capacities for carrying medical appliances Paramedic capabilities
2	Select locations of depots and deployment sites	Real estate cost, distances involved, traffic characterization
3	Response policy optimization	Centralized vs. distributed decision making of which UAV to handle emergency call
4	Marketing	Environment: consumer tastes, competition
5	Safety assurance	Design of alarms, escape routes, fail-safe plans
6	Interfacing with existing systems	Coordination mechanisms, communication protocols, interoperability
7	Patient satisfaction	Waiting time, comfort, etc.
8	Determine Emergency response required	Types and frequencies of medical emergencies to be treated Response timelines requirements and how to meet them
9	Autonomous adaptive behavior	Ability to adapt to changing environments associated with different catastrophic events
10	System sustainment, maintenance and evolution	Pricing of services, costs of equipment, investment capital, long term trends, unused capacity, growth potential

These models were developed in the order of presentation above following the methodology in Figure 4. The first model assumes vehicles are essentially point elements moving in space with abilities to respond to requests without coordination in a neighborhood. The second is a highly *lumped* model that represents the fleet in an ensemble sense similar to the ideal gas laws of physics. The third introduces treating vehicles as individuals requiring coordination to provide service, while the fourth elaborates on this representation to introduce more of the required constraints.

The fifth model directly relates to modeling of CASOS for system engineering design. Such a design might search for architectures in a trade-off space involving size and cost. Here, for example, half the number of UAVs at, say, half the cost, might be enough to ensure a response time that is only 10% higher than nominally specified. However, rather than the systems engineer having to determine the fleet size prior to fielding, s/he might design upper/lower brackets within which to constrain an adaptive plan. This might be similar to the way in which the number of Uber drivers in a town adjusts to its passenger demand. Such “design for adaptive sustainment” objectives call for inclusion of models of the environment in which adaptation is occurring as well as of the mechanism mediating the process. A wide variety of such representations is available at different levels of abstraction and computational complexity with typical parallels drawn to biologically inspired learning and evolutionary processes [43–45]. One possibility that seems especially apt here (and is rarely considered) rests on the analogy to the carrying capacity of an ecosystem for a member species. Here, a species dynamically adjusts to a population size in which it is in equilibrium with the resources necessary to sustain it [42]. Analogously to build in such self-adjustment into a fleet of UAVs requires an economic mechanism in which UAVs must “earn their keep” and means for infusion/withdrawal of UAVs to/from the operating fleet (a direct illustration of supply/demand in the ontology of Figure 2). Such adaptive models can be tested in simulation and fielded with little change in code using model continuity methods.

The assumptions, limitations, and contribution of each model are listed in Table A2. The developmental progression illustrates that addition of assumptions and removal of constraints does not necessarily proceed in a monotonic manner but in a manner that is guided by the incremental needs to establish bounds on predictions to come later and to develop easier structural scaffolds for later construction.

Table A2. A subset of models developed for objectives 1–3 and 8–10 of Table A1.

Model	Assumption	Limitation	Contribution
UAVMotion	Kinetic spatial distribution of requests, policy employs visibility and spatial properties	No duty cycle representation No startup No product properties	Predicts number of UAVs required under fundamental spatial properties
MarkovDutyCycleCTM	Combined stochastic and deterministic representation of duty cycle	Above limitations + Lumps UAVS into single number that determines service rate	Predicts numbers required under simple duty cycle approximation
Multi-workflow	UAVs individually represented as servers in duty cycle, UAV distance to request employed, policy employs visibility & spatial properties	No product properties—capacity and abstracted work cycle	More refined prediction including performance/cost from Pareto frontier
Hierarchical composition	All requirements 1–3 and 8 and constraints accounted for to first approximation	No spatial representation, no account of energy utilization	Good solution approximation under service requirements/constraints—agrees with earlier predictions
Design for Adaptive Sustainment	Extends UAVMotion model with economic UAV “earn their keep” mechanism and modification of fleet size	Same as UAVMotion plus simplification of economics and fleet size modification	Can predict dynamics of “carrying capacity” including fleet size equilibrium in fixed demand environment

References

1. Wymore, A.W. *Model-Based Systems Engineering*; CRC Press: Boca Raton, FL, USA, 1993.
2. Wymore, A.W. *A Mathematical Theory of Systems Engineering: The Elements*; Krieger: Huntington, NY, USA, 1967.
3. Dahmann, J. *Systems Engineering Guide*; MITRE Publication: McLean, VA, USA, 2018.
4. Mittal, S. Model Engineering for Cyber Complex Adaptive Systems. In Proceedings of the European Modeling and Simulation Symposium, Bordeaux, France, 10–12 September 2014.
5. Mittal, S.; Martin, J.L.R. Simulation-based Complex Adaptive Systems. In *Guide to Simulation-Based Disciplines: Advancing Our Computational Future*; Mittal, S., Durak, U., Oren, T., Eds.; Springer: New York, NY, USA, 2017.
6. Mittal, S. Emergence in Stigmergic and Complex Adaptive Systems: A formal Discrete Event Systems perspective. *Cogn. Syst. Res.* **2013**, *21*, 22–39. [CrossRef]
7. Mittal, S.; Rainey, L.B. Harnessing Emergence: The Design and Control of Emergent Behavior in System of Systems Engineering. In Proceedings of the Summer Computer Simulation Conference, Chicago, IL, USA, 24–27 July 2016.
8. Orellana, D.; Madni, A. Extending model based systems engineering for human machine interaction analysis and fault tolerant design. In Proceedings of the Infotech@Aerospace, Garden Grove, CA, USA, 19–21 June 2012.
9. Tolk, A.; Diallo, S.; Mittal, S. The Challenge of Emergence in Complex Systems Engineering. In *Emergent Behavior in Complex Systems Engineering: A Modeling and Simulation Approach*; Mittal, S., Diallo, S., Tolk, A., Eds.; John Wiley & Sons Ltd: Hoboken, NJ, USA, 2018.
10. Mittal, S.; Diallo, S.; Tolk, A. *Emergent Behavior in Complex Systems Engineering: A Modeling and Simulation Approach*; John Wiley & Sons Ltd: Hoboken, NJ, USA, 2018.
11. Mittal, S.; Durak, U.; Oren, T. *Guide to Simulation-Based Disciplines: Advancing Our Computational Future*; Springer: New York, NY, USA, 2017.
12. Svyatoslav, K. *The Practice of Enterprise Architecture: A Modern Approach to Business and IT Alignment*; SK Publishing: Melbourne, Australia, 2018.
13. Zeigler, B.P.; Nutaro, J.J. Towards a Framework for More Robust Validation and Verification of Simulation Models for Systems of Systems. *JDMS Appl. Methodol. Technol.* **2016**, *13*, 3–16. [CrossRef]
14. Diallo, S.; Mittal, S.; Tolk, A. Research Agenda for Next Generation Complex Systems Engineering. In *Emergent Behavior in Complex Systems Engineering: A Modeling and Simulation Approach*; John Wiley & Sons Ltd: Hoboken, NJ, USA, 2018.
15. Mittal, S.; Zeigler, B.P. Theory and Practice of M&S in Cyber Environments. In *The Profession of Modeling and Simulation: Discipline, Ethics, Education, Vocation, Societies and Economics*; Tolk, A., Oren, T., Eds.; John Wiley & Sons Ltd: Hoboken, NJ, USA, 2017.
16. Zeigler, B.P.; Muzy, A.; Kofman, E. *Theory of Modeling and Simulation*, 3rd ed.; Academic Press: New York, NY, USA, 2018.
17. Mittal, S.; Martin, J.L.R. *Netcentric System of Systems Engineering with DEVS Unified Process*; CRC Press: Boca Raton, FL, USA, 22 February 2013.
18. Camus, B.; Paris, T.; Vaubourg, J.; Presse, Y.; Bourjot, C.; Ciarletta, L.; Chevrier, V. Co-simulation of cyber-physical systems using a DEVS wrapping strategy in the MECSYCO middleware. *SAGE J. Simul.* **2018**. [CrossRef]
19. Zeigler, B.P.; Marvin, J.W.; Cadigan, J.J. Systems Engineering and Simulation: Converging Toward Noble Causes. In Proceedings of the 2018 Winter Simulation Conference, Gothenburg, Sweden, 9–12 December 2018.
20. Bocciarelli, P.; D'Ambrogio, A.; Giglio, A.; Paglia, E. Model Transformation Services for MSaaS Platforms. In Proceedings of the Model-driven Approaches for Simulation Engineering Symposium, Baltimore, MD, USA, 15–18 April 2018.
21. Amissah, M.; Toba, A.L.; Handley, H.A.H.; Seck, M. Towards a Framework for Executable Systems Modeling: An Executable Systems Modeling Language. In Proceedings of the Model-Driven Approaches for Simulation Engineering Symposium, Baltimore, MD, USA, 15–18 April 2018.
22. Aliyu, H.O.; Maïga, O.; Traoré, M.K. The high level language for system specification: A model-driven approach to systems engineering. *Int. J. Model. Simul. Sci. Comput.* **2016**, 7. [CrossRef]

23. Abdurrahman, A.; Sarjoughian, H. Model-Driven Time-Accurate DEVS-Based Approaches for Design. In Proceedings of the Model-Driven Approaches for Simulation Engineering Symposium, Baltimore, MD, USA, 15–18 April 2018.
24. Madni, A.M. *Transdisciplinary Systems Engineering: Exploiting Convergence in a Hyper-Connected World*; Springer: Heidelberg, Germany, 2018.
25. Kavak, H.; Padilla, J.; Lynch, C.; Dialio, S. Big Data, Agents, and Machine Learning: Towards A Data-Driven Agent-Based Modeling Approach. In Proceedings of the Annual Simulation Symposium, Baltimore, MD, USA, 15–18 April 2018.
26. Wang, F.; Zhang, J.J.; Zheng, X.; Wang, X.; Yuan, Y.; Dai, X.; Zhang, J.; Yang, L. Where Does AlphaGo Go: From Church-Turing Thesis to AlphaGo Thesis and Beyond. *IEEE/CAA J. Autom. Sin.* **2018**, *3*, 113–120.
27. Zeigler, B.P. *Multifaceted Modeling and Discrete Event Simulation*; Academic Press: Cambridge, MA, USA, 1984.
28. Pawletta, T.; Schmidt, A.; Zeigler, B.P.; Durak, U. Extended Variability Modeling Using System Entity Structure Ontology Within MATLAB/Simulink. In Proceedings of the 49th Annual Simulation Symposium, Pasadena, CA, USA, 3–6 April 2016.
29. Choi, S.H. Ranking and Selection Framework for Efficient Reverse Simulation. Ph. D. Dissertation, Korea Advanced Institute of Science and Technology, Daejeon, Korea, 13 June 2018.
30. Zeigler, B.P. Simulation-based Evaluation of Morphisms for Model Library Organization. In *Model Engineering for Simulation*; Zhang, L., Lie, L., Zeigler, B.P., Eds.; Elsevier: New York, NY, USA, 2019.
31. Zeigler, B.P.; Mittal, S.; Traoré, M.K. Fundamental Requirements and DEVS Approach for Modeling and Simulation of Complex Adaptive System of Systems: Healthcare Reform. In Proceedings of the Symposium on Modeling and Simulation of Complexity in Intelligent, Adaptive and Autonomous Systems, Baltimore, MD, USA, 15–18 April 2018.
32. Jepperson, R.; Meyer, J.W. Multiple Levels of Analysis and the Limitations of Methodological Individualisms. *Sociol. Theory* **2011**, *29*, 54–73. [CrossRef]
33. Davis, P.K.; Bigelow, J.H. *Experiments in Multi Resolution Modeling (MRM)*; Rand: Santa Monica, CA, USA, 1998.
34. NPS Faculty. Aggregated Combat Models. Available online: http://faculty.nps.edu/awashburn/Washburnpu/aggregated.pdf (accessed on 3 April 2017).
35. Davis, P.K.; Hillestad, R. Families of Models that Cross Levels of Resolution: Issues for Design, Calibration and Management. In Proceedings of the Winter Simulation Conference, Los Angeles, CA, USA, 12–15 December 1993.
36. Martin, J.L.R.; Mittal, S. Model Management and Execution in DEVS Unified Process. In *Model Engineering for Simulation*; Zhang, L., Zeigler, B.P., Laili, Y., Eds.; Academic Press: Cambridge, MA, USA, 1 March 2019.
37. MBSE Wiki. Available online: http://omgwiki.org/MBSE/doku.php (accessed on 11 November 2018).
38. Zeigler, B.P.; Sarjoughian, H. *Guide to Modeling and Simulation of System of Systems*; Springer: New York, NY, USA, 2017.
39. Zeigler, B.P.; Nutaro, J.J.; Seo, C. Combining DEVS and model-checking: concepts and tools for integrating simulation and analysis. *Int. J. Simul. Process. Model.* **2017**, *12*. [CrossRef]
40. Gholami, S. Combined DEVS multiresolution simulation and model checking. In Proceedings of the Winter Simulation Conference, Las Vegas, NV, USA, 3–6 December 2017.
41. Neto, V.V.G.; Manzano, W.; Kassab, M.; Nakagawa, E.Y. Model-Based Engineering & Simulation of Software-Intensive Systems-of-Systems. In Proceedings of the 12th European Conference on Software Architecture: Companion Proceedings, Madrid, Spain, 24–28 September 2018.
42. Arrow, K.; Bolin, B.; Costanza, R.; Dasgupta, P.; Folke, C.; Holling, C.S.; Jansson, B.O.; Levin, S.; Mäler, K.G.; Perrings, C.; Pimentel, D. Economic Growth, Carrying Capacity, and the Environment Science. *Ecol. Econ.* **1995**, *15*, 91–95. [CrossRef]
43. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction, Bradford Book*; MIT Press: Cambridge, MA, USA, 1998.
44. Muzy, A.; Zeigler, B.P. Activity-Based Credit Assignment Heuristic for Simulation-Based Stochastic Search in a Hierarchical Model Base of Systems. *IEEE Syst. J.* **2017**, *11*, 1916–1927. [CrossRef]
45. Vasbinder, J.W.; Gao, H. *Selected Papers of John H. Holland: A Pioneer In Complexity Science*; World Scientific Press: Singapore, 2018. [CrossRef]
46. Whitehead, A.N.; Russell, B. *Principia Mathematica*; Cambridge University Press: Cambridge, MA, USA, 1913.

47. Maier, M.W. Architecting Principles for Systems-of-Systems. *J. Int. Counc. Syst. Eng.* **1998**, *1*, 267–284. [CrossRef]
48. Ören, T.; Mittal, S.; Durak, U. Induced Emergence in Computational Social Systems Engineering: Multimodels and Dynamic Couplings as Methodological Basis. In *Emergent Behavior in Complex Systems Engineering: A Modeling and Simulation Approach*; Mittal, S., Diallo, S., Tolk, A., Eds.; John Wiley & Sons Ltd: Hoboken, NJ, USA, 2018.
49. Zeigler, B.P. Structuring the Organization of Partial Models. *Int. J. Gen. Syst.* **1978**, *4*, 81–88. [CrossRef]