1

The Complexity in Application of Modeling and Simulation for Cyber Physical Systems Engineering

Saurabh Mittal[1] *and Andreas Tolk*[2]

[1] *The MITRE Corporation, Fairborn, OH, USA*
[2] *The MITRE Corporation, Hampton, VA, USA*

1.1 Introduction

Cyber Physical Systems (CPS), according to a definition provided by National Science Foundation (NSF) are hybrid networked cyber and engineered physical elements co-designed to create adaptive and predictive systems for enhanced performance. These systems are built from, and depend upon, the seamless integration of computation and physical components. Advances in CPS are expected to enable capability, adaptability, scalability, resiliency, safety, security, and usability that will expand the horizons of these critical systems.

CPS engineering is an activity that brings these elements together in an operational scenario. Sometimes, an operational scenario may span multiple domains, for example, Smart Grid incorporating Power critical infrastructure and Water infrastructure. Intelligent home devices such as a smart washing machine utilize both the infrastructures. Another example would be Smart Transportation system wherein intelligent transportation devices interact with numerous smart vehicles to coordinate large scale traffic behaviors. Numerous such examples exist within the Internet of Things (IoT) perspective. These complex systems involve components at varying level of specifications. The constituent elements are supplied by multiple vendors and composing a solution without a formal test and evaluation infrastructure is a real challenge. Integration of functionality is not happening before the deployment, but after CPS are already deployed. CPS engineering requires a consistent model of operations that need to be supported by the compositions of various CPS contributors. CPS engineering lacks tools to design and

Complexity Challenges in Cyber Physical Systems: Using Modeling and Simulation (M&S) to Support Intelligence, Adaptation and Autonomy, First Edition. Edited by Saurabh Mittal and Andreas Tolk.

experiment within a lab setting. How does one develop a repeatable engineering methodology to evaluate ensemble behaviors and emergent behaviors when larger systems involving critical infrastructure cannot be brought in a lab setting?

This increase in overlapping CPS capability in multitude of domains also introduces a level of complexity unprecedented in other engineered systems. The cross-sector deployment and usage introduces risk that may have cascaded impacts in a highly networked environment. One possibility to reduce the technical risk is to remotely control the systems in the cyber environments, but the sheer number of variables and possible situations introduce complexity at multiple scales. This complexity results in test plans with limited coverage. Additional cyber physical system related issues of intelligence, adaptation, autonomy, and security make the problem even worse. The proposed solution is the enhanced use of Modeling and simulation (M&S). The M&S discipline has supported the development of complex systems since its inception. During the Spring Simulation Multi-Conference 2017, a group of invited experts discussed general challenges in M&S of CPS. In 2018, as follow-on panel was launched dealing with how the combination of various simulation paradigms, methods – so-called hybrid simulation – can be utilized regarding complexity, intelligence, and adaptability of CPS.

While the focus of CPS is both on computation and physical devices, it belongs to the class of super complex systems in a man-made world, where labels such as System of Systems (SoS), Complex Adaptive Systems (CAS), and Cyber CAS (CyCAS) are used interchangeably (Mittal 2014; Mittal and Risco-Martín 2017a). All of them are multi-agent systems. 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.

Complex systems engineering identified a set of methods needed by systems engineers to govern such complex systems and cope with new challenges, like emergent properties or behavior not known in traditional systems. Many of these methods are rooted in the M&S discipline (Mittal et al. 2018). This chapter will provide an overview on the M&S methods and technologies that aid CPS engineering in the development and testing phase, and CPS governance when they are deployed in complex cyber environments. How to apply such means to enable the full potential of CPS is one of the grand challenges of our days. With this volume, we contribute to the discussion of developing a computational infrastructure for modeling, simulation, experimentation, and analytics in a transdisciplinary CPS context.

The chapter is organized as follows. Section 1.2 provides an overview on multiple modalities of CPS. Section 1.3 describes the fundamental issues with CPS engineering. Section 1.4 describes the current M&S technology, especially the co-simulation methodology, available for CPS engineering for developing a virtual CPS environment. Section 1.5 describes the intelligence, adaptation, and autonomy aspect of CPS and how the computational element in CPS provides opportunities for advanced control and access mechanisms. Section 1.6 concludes the chapter.

1.2 Multimodal Nature of CPS

CPS are also considered as systems with integrated physical and computational capabilities that can interact with humans through variety of modalities (Baheti and Gill 2011). This ability to interact with the physical world through computational means, and by doing so expanding the capabilities of the user of the CPS, allows the CPS to interact within a team, such as enabling human-machine-collaborations, as well as with the environment, such as providing alternative means of locomotion – moving of the CPS –, actuation – positioning of sub-components, such as sensors –, or manipulation – interacting with the environment.

This multimodality, the ability to interact with humans, others CPS, and the environment via a multitude of computational and physical means, is one of main sources for the complexity challenges we are coping with. It allows CPS to work in different domains and sectors, and provide their services to many different users. The same functionality can be accessed via several different interfaces to be applied in a multitude of contexts in various domains, making the validation of the CPS challenging, if not impossible. As observed in (Rajkumar et al. 2010), "... the gap between formal methods and testing needs to be bridged. Compositional verification and testing methods that explore the heterogeneous nature of CPS models are essential. V&V must also be incorporated into certification regimes" (page 735).

But validation is not the only concern. The multimodality leads to a multitude of interconnections between potentially many CPS, users, and components of the environment, creating a system of interlinked and interdependent objects. Combined with capabilities that now can be applied by CPS in the same domain, the overall complexity increases significantly.

The other side is, however, that the amount of options for an appropriate reaction in an unforeseen turn of events increases also. If many CPSs can provide a wide variety of services to the same domain, the likelihood that even under catastrophic circumstances we still have options for the appropriate reactions available increases as well. Multimodality is therefore not only the source for more complexity, it also provides the means to cope with it, as it enables higher agility and flexibility. If one modality fails or is not available, it can be quickly replaced by an alternative invocation structure. If one service usually applied within a domain does not succeed, an alternative service may lead to the desired result as well. The multimodel nature of CPS creates the challenge, but it also provides the means to cope with it.

1.3 Why CPS Engineering Is Complex?

In todays world, engineering has taken on a new meaning when different branches of Science are brought together, for example, biology and physics in Biomedical engineering, cognitive psychology and systems in Intelligent Systems engineering, urban policy and management, and automobile in Transportation engineering, and the most complex of all, Smart City engineering that includes transportation, smart automobiles, intelligent infrastructure, human factors, cybersecurity, and many more. Likewise, the multi-domain warfare involves air, sea, and land falls into this category as well, so do IoT, the Industry 4.0 initiatives, and CPS. This is the world of complex systems engineering. One characteristic property of such complex systems is that there are always some unknowns. There is high variability. One can never have a complete set of variables to apply brute force engineering based on a single branch of science. Consequently, per the Law of Requisite Variety, it is impossible to develop a controller for such a complex system (Mittal and Rainey 2015). This incomplete information and uncertainty in developing control mechanisms lead to emergent behaviors which is the hallmark of any complex system (Mittal 2013; Mittal et al. 2018). Consequently, methodologies are needed that embrace emergent behaviors as features of such a complex system.

Complex systems today are multi-disciplinary systems that require multiple branches of Science to interact with competing, contrasting, or orthogonal theories. A model may be valid in one scientific theory and simultaneously, may be completely invalid in another branch of Science, for example, consider Quantum mechanics and Newtonian mechanics operating in the same computational model. While they work in reality, it is a difficult problem altogether in a computational realm. Yet, M&S is the best we have to test any upcoming scientific theory or engineer a system-at-scale in a lab setting in any multi-disciplinary endeavor (Mittal et al. 2018).

Due to the proliferation of various modalities, both hard sciences (bound by physics and mathematics) and soft sciences (e.g. cognitive science, sociology) need to be brought in a computational environment for verification and validation (V&V), test and evaluation (T&E), and experimentation purposes for such CPSs. Bringing these fundamental sciences together require the incorporation of emergent behavior that may arise as different scientific theories are brought to interact within a CPS use-case. Development of CPS through the fundamentals of Systems Theory supported by an equally robust M&S theory is the preferred way forward (Mittal and Zeigler 2017).

A typical CPS comprises of the following components:

- Sensors
- Actuators
- Hardware platforms (that hosts sensors and actuators)
- Software interfaces (that accesses hardware directly or remotely through a cyber environment)
- Computational software environments (that may act both as a controller or service provider)
- Networked environments (that allows communication across geographical distances)
- End user autonomy (that allows CPS to be used as a passive system or an active interactive system)
- Critical infrastructures (water, power, etc., that provides the domain of operation and operational use-case)
- Ensemble behaviors
- Emergent behaviors

Figure 1.1 shows various aspects of CPS divided into Left-Hand Side (LHS) and Right-Hand Side (RHS). LHS consists of collection of users, systems (both

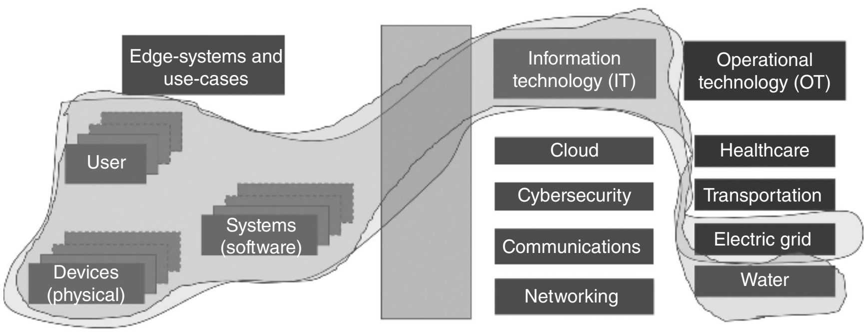

Figure 1.1 CPS landscape.

hardware and software), and devices (physical platforms). Traditional systems engineering practices and end-user use-cases can be developed in LHS. The RHS shows aspects related to infrastructures. Fundamentally, they can be characterized into Information Technology (IT) and Operational Technology (OT). Between the LHS and RHS is the network/cyber environment that allows information exchange between the two. With the network spanning large geographical distances, the presence of large number of entities/agents and their concurrent interactions in the CPS result in ensemble and emergent behaviors. The infrastructure-in-a-box is largely unavailable but can be brought to bear with various existing domain simulators in an integrated simulation environment.

In M&S, we are not only limited to the computational implementations of models. We distinguish between live simulations in which the model involves humans interacting with one another (role playing, play acting, etc.), virtual simulations where the model is simulated by a fusion of humans and computer-generated experiences, and constructive simulations where the model is entirely implemented in a digital computer and may have increased levels of abstraction. Increasingly, we are mixing the three forms of simulation in what is commonly known as live-virtual-constructive (LVC) simulation (Hodson and Hill 2014). LVC simulations are used mainly for training but they can be adapted for the type of experimentation/exploration needed to investigate emergent behavior (Mittal et al. 2015), as shown by the cyclical process in Figure 1.2, elaborated in (Mittal et al. 2018).

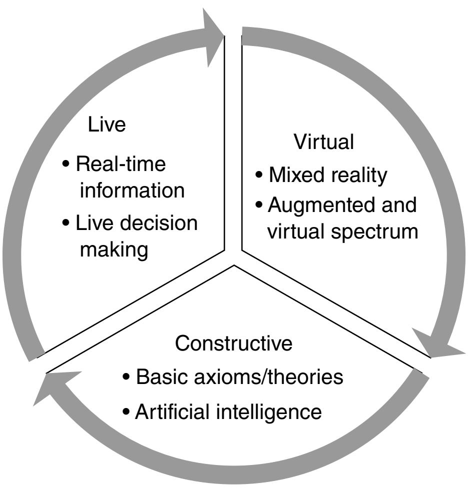

Figure 1.2 Experimental LVC approach for generating emergence.

Table 1.1 CPS contributor and the associated M&S paradigm.

CPS contributor	M&S paradigm	LVC element
Sensors	Continuous, physics-based	L, V, C
Actuators	Continuous, physics-based	L, V, C
Hardware platform	Both continuous and discrete	L, V
Software platform	Discrete	V, C
Network	Discrete	L, V
End user	Discrete, agent-based	L, V, C
Critical infrastructure	Both continuous and discrete (hybrid)	V, C
Ensemble, emergent behaviors	Discrete agent-based	V, C

Table 1.1 associates each of the CPS constituent element with the corresponding M&S paradigm and how it can be incorporated in the LVC environment.

From a systems theoretic perspective, a CPS model is a hybrid system made up of both continuous and discrete systems. A continuous system (CS) is one that operates in continuous time and in which input, state, and output variables are all real values. A discrete (dynamic) system (DDS) is one that changes its state in piece-wise constant event-based manner (which also included discrete-time systems as they are a special case of discrete event systems) (Lee et al. 2015). A typical example of a hybrid system is a CPS in which the computation subsystem is discrete and a physical system is CS. A CyCAS (Mittal 2014) in LVC environment also qualifies as a CPS, with live systems as CS, constructive systems as discrete and virtual systems as hybrid, containing both continuous and discrete. At the fundamental level, there are various ways to model both timed and untimed discrete event systems, all of which can be transformed to, and studied within, the formal Discrete Event Systems (DEVS) theory (Mittal 2013; Mittal and Risco-Martin 2013; Mittal and Martin 2016; Traor et al. 2018; Vangheluwe 2000; Zeigler et al. 2000).

1.4 M&S Technology Available for CPS Engineering

M&S is being considered as a vehicle by which complex systems engineering, including CPS engineering could be done. However, using M&S for CPS engineering is not straight forward due to the inherent complexities residing in both the modeling and the simulation activities. Simulation subsumes modeling. Performing a CPS simulation requires that a CPS model be first built. While CPS modeling is not the focus of this chapter, a recent panel explored the state of the art of CPS modeling and the complexity associated in engineering intelligence, adaptation, and autonomy through M&S. The literature survey conducted in Tolk et al., (2018) enumerate the following active research areas and the associated technologies for CPS modeling and concluded that the need for a common formalism that can be applied by practitioners in the field is not yet fulfilled:

- DEVS formalism: Strong mathematical foundation that support multi-paradigm modeling, multi-perspective modeling, and complex adaptive systems modeling to handle emergent behaviors.
- Process algebra: Provides hybrid processes using multi-paradigm modeling. Models combine behavior on a continuous time scale with discrete state transition behavior at given points in time.
- Hybrid automata: Combines finite state machines with Ordinary Differential Equations (ODE) to account for non-deterministic finite states. Bond graphs are used to govern changes.
- Simulation languages: Combines discrete event and continuous system simulation languages. Involves modular design of hybrid languages, multiple abstraction levels combining different formalisms.
- Business processes: Use of standardized notation languages like Business Process Modeling Notation (BPMN) provides value in securing buy-in from the stakeholders in an efficient manner.
- Interface design for co-modeling: Functional Mock-up Interface (FMI) as a means of integration of various CPS components. DEVS can also be used as a common denominator in a vendor neutral manner.
- Model-driven approaches: Model transformation chains to arrive at a single formal model. Governance is required to develop such automation.
- Agent-based modeling: Paradigm to employ component models at scale with individual behaviors, to study ensemble effects.

The above-mentioned approaches and technologies allow the development of CPS models, albeit in a piece-wise manner. These model pieces and their definitions and specifications are dictated by the cross-domain CPS operational use-case. Assuming we now have a validated model (i.e. a model that has been deemed valid by the stakeholders), next comes the task of executing it on a computational platform, i.e. simulation. The piece-wise model composition sometime does not directly translate into a monolithic simulation environment due to the confluence of both the continuous and discrete system in the hybrid system. In the literature survey (Tolk et al. 2018) as well as in many discussions with the experts, the use of co-simulation was identified as the preferred course of action in support of CPS for development, testing, and eventually training.

Co-simulation is the co-existence of independent simulators to support a common model (Mittal and Zeigler 2017). To understand co-simulation, consider a complex system model comprising of Electric Grid, thousands of smart homes, and data-communication network. This would require modeling to be done for:

1) Continuous system for the Power system (using GridLab-D power flow simulator)
2) Continuous system for Building simulation (using LabView simulator)
3) Discrete system for the smart home behavior (using model-predictive software controllers in Generic Algebraic Modeling Systems (GAMS) language)
4) Discrete system model for data communication network (using OmNet++ simulator)

The simulators used to implements both the discrete and continuous system models in the common Electric Grid hybrid system model are shown in parenthesis. Such hybrid systems demonstrate how a large-scale hybrid modeling could be attempted and can eventually lead to a robust simulation environment, bringing together user(s) in smart homes as agents with big infrastructure such as Electric Grid through the power flow simulator. The first example at Oak Ridge National Lab (ORNL) developed a complex system with item #1 and #4, described in Nutaro et al. (2008). The second example at National Renewable Energy Lab (NREL) developed a system comprising of item #1, #2, and #3, described in Pratt et al. (2015), and with a stronger flavor on co-simulation application in Mittal et al. (2015). Both examples integrated different modeling paradigms and ran simulations on High Performance Computing (HPC) environment in virtual (as-fast-as-possible) and real (wall-clock) time. The NREL effort also integrated air-conditioner hardware with the simulation exercise for a real-time 7-day scenario, described in detail in Pratt et al. (2017). Both the efforts at ORNL and NREL employed the DEVS formalism for integrating the constituent simulators. The survey published in Thule et al. (2008) provides a state-of-the-art overview of co-simulation practices and applications in multiple domains, including CPS.

A scalable M&S architecture has distinct modeling and simulation layers. In order to deploy in cloud environment, sufficient automation is needed at both the simulation layer and the modeling layer. This can now be achieved by current practices in DevOps implemented using Docker technology. DevOps, a recent buzzword, provides methodologies to automate developer operations, such as compiling, building, releasing, testing through executable scripts. Mittal and Risco-Martin (2017b) integrated Docker with the granular service oriented architecture (SOA) Microservices paradigm and advanced the state-of-the-art in model and simulation interoperability. This automated deployment of various “DEVS nodes” under a single administrative control is defined as a DEVS Farm. They described the architecture incorporating DevOps methodologies using containerization technologies to develop cloud-based distributed simulation farm for DDS systems specified using DEVS formalism. The research will extend towards containerization of various other simulators either through DEVS wrappers or Functional Mockup Interface (FMI) standards as functional mockup units (FMUs).

Knowledge engineering is an activity that has not been adequately dealt within cognitive architectures when it comes to bringing high-level knowledge structures into existing architectures. They have largely focused on symbolic representation, memory structure, and symbol manipulation. As the corpus is small, likewise, the environments these cognitive architectures can be put to use, is also limited to simple to moderate operational environments. Further, issues like abductive reasoning, dynamic memory that acquire new conceptual structures, creative aspects of problem solving, emotional processing, along with plausibly related concepts of metacognition and goal reasoning have received little attention (Langley 2017) and may require revision to established cognitive theories. A middle ground using a cognitive framework like Belief-Desire-Intention (BDI) in conjunction with (i) algorithms and heuristics, and (ii) high level knowledge representation, will provide adequate strength for a developing a rational agent, capable of handling complex situations. Prior work at Air Force Research Lab (AFRL) demonstrates the development of integrated cognitive systems for building artificial systems (Douglass and Mittal 2013). Further, work (Mittal and Zeigler 2014a, 2014b) on attention-switching for resource-constrained complex intelligent dynamical systems (RCIDS) provide further evidence of bringing together cybernetics, Systems Theory, Cognitive Science, and Software engineering to develop attention-focusing activity-based systems.

These recent developments bring together cloud technologies, co-simulation methodologies, and hybrid modeling approaches to deliver an M&S substrate that is applicable across the entire CPS landscape (Figure 1.3). The LHS in Figure 1.1 employs traditional Systems Engineering practices, and provides the context

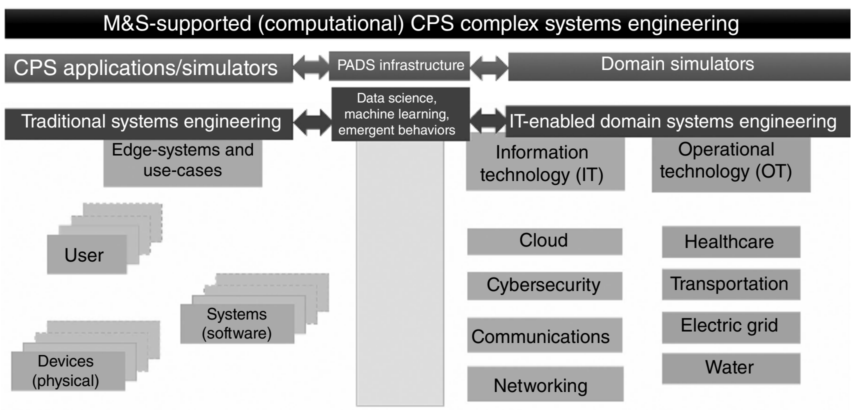

Figure 1.3 M&S supported (computational) CPS engineering test-bed perspective.

use-case for CPS applications. The RHS provides various domain simulators and employ IT and OT to provide "infrastructure-in-a-box" through LVC architectures. To bridge, LHS and RHS, emerging disciplines like Machine Learning and Data Science will need to be employed to get a handle on data-driven approaches that tackle emergent behaviors when LHS and RHS interact in a parallel distributed discrete event co-simulation environment.

1.5 Intelligence, Adaptation, and Autonomy Aspects

Intelligent, adaptive, and autonomous are characteristics often used to describe CPS. In this section, we will provide some of the references to important work supporting to gain a better understanding of these tightly connected topics.

1.5.1 Intelligence

Having to define intelligence, adaption, and autonomy is not an easy task, in particular not for the first term. Scientists and philosophers try to understand for centuries what intelligence is. In the context of computational intelligence, the famous "Imitation Game," as described in Turing (1950), avoids providing definitions of intelligence itself but instead focuses on the question: "Can computers pass a behavioral intelligence test?" For many applications, such a test is suitable, but not all forms of computational intelligence can be tested in form of such Turing tests, in particular not when the modalities give the nature of the system under test away. A smart device that behaves intelligently will nonetheless be immediately be recognized as a device, which likely will bias the evaluating person, who now knows that he is testing a machine, not a human.

However, systems engineering and computer science are collaborating for several years now to develop smart systems. In Tolk et al. (2011), we summarized several abilities an intelligent system needs to expose. This list reflects the collective view of various artificial intelligence viewpoints. It is neither complete nor exclusive, but it is still valid and builds a good foundation for smart CPS understanding as well. This list contains adaptability and autonomy as characteristics for intelligence based systems as well, again showing the close proximity of these concepts pivotal to CPS.

- Intelligent system can *explain* their decision. It is important that smart decision come to an applicable decision, but they should also be able to explain the reasoning behind such decisions. This is particularly important in human-machine-teams, where the human must be able to understand the decision process and interact with it, or eventually even overrule the decision.

Furthermore, the system behavior of smart devices may change due to learning and adaption, so explaining new reasoning is important.

An example is the explanation component of expert systems used for diagnosis tracing the line of reasoning used by the underlying inference engine to answer the questions: Why is the answer to the question the one you recommend? For systems that are able to modify themselves being able to explain their reason is mandatory to ensure credibility.

Furthermore, the decision space of computational systems is always closed by definition, as it is the union of the range of all included computable functions. If a situation comprises a characteristic attribute outside of the dimension of the space, the attribute does not contribute to the decision which hence may be sub-optimal. This is very important if the intelligence is based on data-driven machine learning using neural networks, as such system will always generate a solution based on the selected training data set and its domain.

- Intelligent systems must be *robust*. This characteristic property of a system means that the system behaves well and adequate not only under ordinary conditions, but also under unusual conditions that stress the original requirements and derived assumptions. In other words, robust systems do not break easily, but are able to continue to behave well even under variant circumstances that could lead to failure of system.

  In the more recent systems engineering literature, the term *antifragile* has been introduced to describe systems that are not only robust, but that actually are getting better under stress (Nicholas Taleb 2012). Again, the often circular definitions are showing up here as well, as many of the characteristics captured in this list are used for explaining antifragile systems as well (Jones 2014).
- Another related characteristic is *fault tolerance*. The intelligent systems will continue to behave well and continue to adequately perform even if one or more of its internal system components fail or break. This tolerance is applicable to external causes, such as malevolent behavior of other systems, as well as to internal causes, such as simple wear and attrition.

  Conducting maintenance procedures to avoid wear and having the ability to repair externally caused damage immediately can help, but these are just custodian abilities to provide fault tolerance. The ability to adapt to a new capability set and plan accordingly is also contributing to fault tolerance, and they will be dealt with in more detail in bullets of their own.
- Another characteristic often used to describe intelligent systems is the ability to *self organize*. They can organize their internal components and capabilities in new structures without a central or an external authority in place. Often, this characteristic also applies to the population of several such systems when conducting a mission to reach a common objective. These new structures can be temporal and spatial. In some cases, instead of self-organizing the term

self-optimizing is used synonymously. When these self-organizing systems interact within a persistent environment, such as in stigmergic systems, new macro behavioral patterns emerge because the environment becomes an interacting agent itself due to a persistent and evolving structure (Mittal 2013).

In particular when complex tasks have to be conducted by many simple systems, the use of cooperative swarm systems applying specialization and self-organization as a special form of distributed artificial intelligence that may lead to emergent structures on the swarm level. Even in defense operations, the use of self-organizing teams has been identified as a good practice in complex environments (Alberts et al. 2010).

- This aspect of *cooperation* generally exposes social capabilities, which is also a characteristic of intelligent behavior.

  Cooperative systems interact with other systems and potentially humans as well via some kind of communication language or any other modalities described earlier in this chapter. This interaction is not limited to pure observation, but these systems can exchange plans, distribute tasks, etc. Whiteboard technologies are as often used as direct communication.
- Intelligent systems are also able to *learn* from observing the achieved results and compare them with the desired outcome. Using methods such as reinforcement learning, decisions that led to positive results are enforced while those with negative results are avoided.

  Learning can also occur by observing other systems and the results of their activities. It is also possible to observe human partners and mimic their behavior.

  Learning can imply deductive as well as inductive methods. Systems can learn general principles from the observations of detailed example, and they can apply general behavioral schemes to guide their decision in new environments. When new knowledge is captured in from of applicable models, abductive learning is possible as well (Håkansson and Hartung 2014).
- The final characteristic to be addressed in this list is *agility*. In general, agile systems are able to manage and apply knowledge and their capability effectively so that they behave well and adequate in continuously and unpredictably changing environments.

  In particular in complex environments, agility is pivotal to react quickly and appropriately in unforeseen situations. It is often connected with intellectual acuity for the situation and the necessary intellectual capability to cope with newly arising challenges.

In the recent years, computational advantages allowed for the rebirth of machine learning, in particular based on data-driven approaches (Witten et al. 2016). The usefulness of such approaches and the success of data science related approaches are clearly documented in many publications as well as solutions, as IBM's famous Watson (High 2012). However, when it comes to CPS, these approaches may not necessarily be the best choice. Many alternative methods supporting computational intelligence, such as collectively described in Steinbrecher (2016), are benefiting from the same computational advantages and provide traceable and explainable solutions. Furthermore, many of the heuristics developed in the first wave of artificial intelligence have been refined over the years and provide today impressive solutions. It is worth mentioning that explainability itself is not well defined with the community. With a tighter focus on CPS, the terms assurance and traceability are often used as the two bounding examples to express the intention behind requiring this characteristic. Assurance is result oriented, asking for system to be able to explain who they ensure that the results of their rules are going to fall within given constraints, such as rules of engagements, safety concerns, etc. The recent Air Force study summarizes several concepts used by the community (Clark et al. 2013). Traceability is interested in how a result can be traced back to all the various decisions made and rules applied. It is very close to the requirements traceability asked for in systems engineering. A broader spectrum of related concepts has been recently compiled in Karlo Došilović et al. (2018).

CPS engineers must be aware of such alternative solutions that may fulfill their requirements better than those currently in the main stream.

1.5.2 Autonomy

Many of these characteristics also apply to enable adaptation and autonomy. As with intelligence, the definition of these terms is challenging.

An autonomous system performs the desired tasks and behaves well and adequate even in complex environments without continuous human guidance. Williams compiled an overview of definitions with focus on the various autonomy scales (2015). Like intelligence, autonomy is multi-faceted and can be observed via many modalities in various forms. The use of different levels is therefore common practice, in particular using multi-dimensional scales consisting of levels with descriptive indicators for each set of dimensions defining a particular facet of autonomy. Wiley concludes his synthesis with the definition of the following autonomy dimensions (Williams 2015).

- *Goals*: an autonomous agent has goals that drive its behavior.
- *Sensing*: an autonomous agent senses both its internal state and the external world by taking in information (e.g. electromagnetic waves, sound waves).
- *Interpreting*: an autonomous agent interprets information by translating raw inputs into a form usable for decision making.

- *Rationalizing*: an autonomous agent rationalizes information against its current internal state, external environment, and goals using a defined logic (e.g. optimization, random search, heuristic search), and generates courses of action to meet goals.
- *Decision making*: an autonomous agent selects courses of action to meet its goals.
- *Evaluating*: an autonomous agent evaluates the consequences of its actions in reference to goals and external constraints.
- *Adapting*: an autonomous agent adapts its internal state and functions of sensing, interpreting, rationalizing, decision making, and evaluating to improve its goal attainment.

When comparing the intelligence characteristics with the autonomy dimensions, the close proximity of both concepts immediately becomes eminent. Nearly all papers on the different degrees of autonomy reference the seminary work of Sheridan and his research into levels of autonomy (1992), who introduced the following 10 levels of increasing autonomy.

1) The computer offers no assistance, the human must do it all.
2) The computer offers a complete set of action alternatives.
3) The computer narrows the selection down to a few.
4) The computer suggests the best selection.
5) The computer executes the option upon human approval.
6) The computer allows the human a restricted time to veto before execution.
7) The computer automatically executes and informs the human.
8) The computer informs the human after execution.
9) The computer decides if to inform the human.
10) The computer acts completely on its own.

The Armed Forces used these categories to identify the following definition for classes of autonomous systems (Williams 2008).

- *Human operated systems* are fully controlled by humans. All activities result from human initialization, eventually based on provided sensor information.
- *Human assisted systems* perform activities in parallel with human inputs, augmenting the human's ability.
- *Human delegated systems* perform limited control activities. The human can overrule the system at any time.
- *Human supervised systems* conduct all activities needed to perform a given mission, but they inform the human consistently, including providing explanations for decisions.
- *Mixed initiative systems* are capable of human-machine teams and can take over given tasks independently.
- *Fully autonomous systems* require no human intervention or presence. They conduct all activities across all ranges of conditions.

It should be pointed out that in particular for CPS such systems are often integrated into a larger system of systems. An anti-break system conducts a well-defined function autonomously, assisting the human in the operation of a car. The auto-pilot of an airplane autonomously flies it under fully and well-define constraints, etc. The borders between the levels are therefore often fluent.

1.5.3 Adaption

The requirement to be adaptive is known from both concepts discussed so far. It describes the ability of a system to change to better fit into a changing environment. These changes can be behavioral as well as structural (Antonio Martn et al. 2009).

Structural changes modify the physical components of the CPS. Many CPS have so called actuators that are used to move components of the CPS into new positions, such as robotic arms, sensors, antennas, etc. In addition, many CPS have modular components that can be switched in case of need to support different environments, such as wheeled or tracked locomotion devices. An interesting featured not yet sufficiently researched is the applicability of new capabilities, such as 3D printing in the field. While we already utilize 3D printers for on-demand spare part production, in the future completely new components will be possible that will help the system to adapt to new challenges.

From the computational perspective, the behavioral adaption is interesting, which addresses mainly the cyber component of the CPS. These computational components provide functions that can be modified, optimized, or completely replaced by an alternative set. The aspect of learning and the methods discussed in Steinbrecher (2016) will be applied here. The aspects of building computational adaptive systems were introduced by Holland in (1992) and mainly embraced for the computational support of social studies. Some of the currently utilized methods are described in detail in Antonio Martn et al. (2009).

Like already observed for intelligence and autonomy, there is no generally accepted definition for adaption, the list of characteristics of adaptive systems is open as well. Nonetheless, the literature agrees on some aspects, as they are covered in the following enumeration. One of the reasons is the high degree of interdisciplinary of the field, as many different user domains take advantage of the computational progress and the ease of available tools.

- Adaptive systems are composed of individual agents that interact with each other, may compete for common resources, and that all follow usually well-defined decision and action processes.
- The macro-behavior of the system results from the interaction of these agents. In complex adaptive systems, the resulting behavior of the system can usually

not directly be derived from the well-understood behavior of the single agents within the systems.

- To enable adaption, feedback loops are a necessary element of the highly dynamic interaction processes. The agents have to be able to learn from this feedback information.
- Cooperation and specialization of agents are also characteristics often mentioned in the literature. The persuasion of a common goals under the constraint of limited resources are driving forces without having to program the behavior into the processes of the agents explicitly.

Currently, the CPS community does not yet take enough advantage of the rich body of knowledge regarding such methods and solutions from the discipline of M&S. As discussed in Tolk et al. (2018), the reason for this may be the lack of awareness of the methods available, but also of the complexity of the underlying problems.

However, as discussed in Mittal et al. (2018), the use of simulation methods is good – if not best – practice when coping with complex environment that often does not provide immediate feedback for actions, but exposes effect of effects and temporarily shifted feedback, often only after several decision cycles after the originating action. The highly non-linear nature of relations in complex system's constituent sub-systems is another challenge. Adaptability in complex environments will require capabilities as provided by M&S. As pointed out in Tolk (2015), there is a close connection between CPS and the agent metaphor. Within the virtual environment, intelligent software agents exhibit the same characteristics as CPS in their environment. As such, software agents are not only good candidates for co-simulation approaches to evaluate the scalability of control approaches, they can also serve to get insights into possible emergence and can serve as a test bed for new rule sets and supporting solutions, or they can be used for the optimization of existing ones. Obviously, the agent-based simulation approaches can and should be augmented by other paradigms and approaches, leading to hybrid approaches as initially discussed in Tolk et al. (2018).

What is furthermore of interest for the application of M&S methods to cope with the complexity of CPS is that they are naturally embedded into to life environment. The continuous feedback from their situated environment to allow them to adapt or optimize their decisions, as discussed above, has been the topic of applied M&S research as well. Of particular interest is the research on Dynamic Data Driven Application Systems, which were made popular in earlier US Air Force research (Darema 2004) and only recently re-introduced as a research topic that can take advantage of the developments in computer technology in Biswas et al. (2018). In the hybrid simulation world, the same principle is referred to as symbiotic simulation (Onggo et al. 2018). All these approaches have in common that simulation solutions, such as CPS cyber components, are regarded as embedded solutions within the situated environment, using feedback loops for control, and utilizing data science methods to create actionable observation. Again, many techniques used for years in the training community, such as captured under the LVC paradigm (Hodson and Hill 2014), should be closely evaluated regarding their ability to contribute to the management of complexity of CPS over the whole lifecycle.

1.6 Conclusion

CPS are complex hybrid systems that span multiple sectors of the society at deployment level. They employ both discrete and continuous systems as they interact with the continuous physical world and the discrete digital world that involve humans as well. The constituent systems may exhibit features of an intelligent, adaptive, or autonmous agent in varying degrees. With IoT in infancy and projected to increase manifold, the CPS when deployed on the World Wide Web (Internet) introduce a lot of risk in the deployed critical infrastructure (such as Energy, Water, Transportation, etc.) which they indirectly use. Consequently, the IoT and CPS of tomorrow is fraught with many challenges and their growth may get stifled as the stakeholders that manage and operate the underlying critical infrastructures become aware of the risks involved. To make matters even more complicated, CPS may have multiple modalities which put the existing CPS in a new operational environment. Without sufficient T&E and V&V, it cannot be predicted if the CPS would produce reliable behavior and would not cause new emergent stresses to the underlying infrastructures leading to cascaded failures. The IT and OT communities that operate various underlying infrastructures may not be cognizant of the larger IoT context as it may fall outside their business operations. Consequently, the responsibility rests solely on CPS engineers to develop a robust CPS with expected modalities and definition of the CPS operations. However, there does not exist an experimentation test-bed that allows CPS T&E in relation to the operational context (over the underlying critical infrastructure).

The use of agent-based development and testing platforms for CPS has been subject of several publications that make the clear connection between intelligent, adaptive, and autonomous CPS and their "digital twin" in form of intelligent software agents within a situated virtual environment that replicates the nature of the physical environment in which the CPS is acting in. One example is given in Sanislav and Miclea (2012), more will follow in the chapters of this book. However, while this approach is sufficient for developing and testing computational functionality in well-defined missions and environments, the complex environment of

CPS often require the human in the loop for testing. The many recent advantages in the domain of LVC methods, such as published in Hodson and Hill (2014) for the military domain, must be taken into consideration as well.

M&S technology provides tractable methods for analysis and experimentation. M&S allows the exploration of the emergent behavior when dealing with complex systems engineering provided adequate attention is paid to simulation engineering and no emergent behaviors arise as a result of error approximation due to selection of a wrong modeling method or error propagation through wrong simulation integration that does weaves the space-time-series incorrectly. M&S is integral part of complex systems engineering and must be utilized through the following mechanisms for maximum impact:

- Live, Virtual, and Constructive (LVC) environments
- Systems engineering testing, evaluation, validation, and verification
- Operations, distribution, and communications

The application of M&S technology to CPS engineering is a non-trivial effort. It requires development of a computational infrastructure that brings together a hybrid model employing various domain simulators in a co-simulation environment for CPS contextual use. The co-simulation environment must be integrated with the state-of-the-art simulation technology employing hybrid modeling, cloud-computing, DevOps, parallel and distributed execution, and abstract time implementation. This will allow simulation experiment studies to be conducted in fast mode with reduced time-to-results. Over the decades, M&S has been a strong partner in developing intelligent, adaptive, and autonomous systems. The communities in these areas acknowledge, use, and advance the M&S technologies themselves. This chapter has discussed the state-of-art in M&S and how the aspects of intelligence, adaptation, and autonomy guide the specifications of CPS in a broader context. The intelligent, adaptive, and autonomous systems communities have begun to co-simulation technologies in piece-wise manner wherein two domains may be brought together through simulators. CPS expands the co-simulation concept many fold as we require an M&S environment with a swappable “CPS context” to experiment with various modalities across multiple critical infrastructures.

The chapters provided in this book have been contributed by invited recognized experts in their contributing fields, which has not been limited to the M&S domain. We asked experts on co-simulation and modeling formalisms as well as CPS practitioners and experts interested in the social aspects of CPS integration to support us in writing this book. The resulting compendium should offer contributions as well as research ideas to bring our communities closer together to tackle the big challenges of complexity in the CPS domain as well as showing the many facets of research and viewpoints that should be considered in this process.

Acknowledgments

The work presented in this paper was partly supported by the MITRE Innovation Program. The views, opinions, and/or findings contained in this paper are those of The MITRE Corporation and should not be construed as an official government position, policy, or decision, unless designated by other documentation. Approved for Public Release; Distribution Unlimited. Public Release Case Number 19-0424.

References

Alberts, D. S., Huber, R. K., & Moffat, J. (2010). *NATO net-enabled capability command and control maturity model (N2C2M2)*. Technical report, Command and Control Research and Technology Program.

Antonio Martn, J., de Lope, J., & Maravall, D. (2009). Adaptation, anticipation and rationality in natural and artificial systems: Computational paradigms mimicking nature. *Natural Computing*, *8*(4), 757.

Baheti, R., & Gill, H. (2011). Cyber-physical systems. *The Impact of Control Technology*, *12*(1), 161–166.

Biswas, A., Hunter, M., & Fujimoto, R. (2018). Energy efficient middleware for dynamic data driven application systems. In M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, & B. Johansson (Eds.), *Winter Simulation Conference* (pp. 628–639). IEEE.

Clark, M., Koutsoukos, X., Porter, J., Kumar, R., Pappas, G., Sokolsky, O., ... Pike, L. (2013). *A study on run time assurance for complex cyber physical systems*. Technical report. Aerospace Systems Directorate, Air Force Research Lab, Wright-Patterson Air Force Base.

Darema, F. (2004). Dynamic data driven applications systems: A new paradigm for application simulations and measurements. In M. Bubak, G. D. van Albada, P. M. A. Sloot, & J. Dongarra (Eds.), *Computational Science – ICCS 2004. ICCS 2004*, Lecture Notes in Computer Science (Vol. *3038*, pp. 662–669). Berlin, Heidelberg: Springer.

Douglass, S., & Mittal, S. (2013). A Framework for Modeling and Simulation of the Artificial. In A. Tolk (Ed.), *Ontology, Epistemology, and Teleology for Modeling and Simulation*, Intelligent Systems Reference Library (Vol. *44*). Berlin, Heidelberg: Springer.

Håkansson, A., & Hartung, R. (2014). An infrastructure for individualised and intelligent decision-making and negotiation in cyber-physical systems. *Procedia Computer Science*, *35*, 822–831.

High, R. (2012). *The Era of Cognitive Systems: An Inside Look at IBM Watson and How It Works*. Armonk, NY: IBM Corporation, Redbooks.

Hodson, D. D., & Hill, R. R. (2014). The art and science of live, virtual, and constructive simulation for test and analysis. *Journal of Defense Modeling and Simulation*, *11*(2), 77–89.

Holland, J. H. (1992). Complex adaptive systems. *Daedalus*, *121*, 17–30.

Jones, K. H. (2014). Engineering antifragile systems: A change in design philosophy. *Proceedings of the 1st International Workshop: From Dependable to Resilient, from Resilient to Antifragile Ambients and Systems*.

Karlo Došilović, F., Brčić, M., & Hlupić, N. (2018). Explainable artificial intelligence: A survey. *Proceedings of 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 210–215). IEEE.

Kruse, R., Borgelt, C., Braune, C., Mostaghim, S., & Steinbrecher, M. (2016). *Computational Intelligence: A Methodological Introduction*. New York: Springer.

Langley, P. (2017). Progress and challenges in research on cognitive architectures. *Proceedings of Thirty-first AAAI Conference on Artificial Intelligence*.

Lee, K. H., Hong, J. H., & Kim, T. G. (2015). System of systems approach to formal modeling of CPS for simulation-based analysis. *ETRI Journal*, *37*, 175–185.

Mittal, S. (2013). Emergence in stigmergic and complex adaptive systems: A formal discrete event systems perspective. *Journal of Cognitive Systems Research*, *21*, 22–39.

Mittal, S. (2014). Model engineering for cyber complex adaptive systems. In *European modeling and simulation symposium*.

Mittal, S., Diallo, S., & Tolk, A. (2018). *Emergent Behavior in Complex Systems Engineering: A Modeling and Simulation Approach*, volume 4 of *Steven Institute Series on Complex Enterprise Systems* (pp. 4). Newark: Wiley.

Mittal, S., Doyle, M., & Portrey, A. (2015). Human-in-the-loop modeling in system of systems M&S: Applications to live, virtual and constructive (LVC) distributed mission operations (DMO) training. In L. B. Rainey & A. Tolk (Eds.), *Modeling and Simulation Support for System of Systems Engineering Applications*. Hoboken, NJ: Wiley.

Mittal, S., & Rainey, L. B. (2015). Harnessing emergent behavior: The design and control of emergent behavior in system of systems engineering. In *Proceedings of Summer Computer Simulation Conference*.

Mittal, S., & Risco-Martin, J. L. (2013). *Netcentric System of Systems Engineering with DEVS Unified Process*. Boca Raton, FL: CRC Press.

Mittal, S., & Risco-Martin, J. L. (2016). DEVSML Studio: A framework for integrating domain-specific languages for discrete and continuous hybrid systems into DEVS-Based M&S environment. In *Proceedings of Summer Computer Simulation Conference*.

Mittal, S., & Risco-Martín, J. L. (2017a). *Simulation-Based Complex Adaptive Systems* (pp. 127–150). Cham: Springer International.

Mittal, S., & Risco-Martin, J. L. (2017b). DEVSML 3.0 stack: Rapid deployment of DEVS farm in distributed cloud environments using microservices and containers. In *Proceedings of the 2017 Spring Simulation Multi-Conference (SpringSim'17)*.

Mittal, S., Ruth, M., Pratt, A., Lunacek, M., Krishnamurthy, D., & Jones, W. (2015). A system-of-systems approach for integrated energy systems modeling and simulation. In *Proceedings of Summer Computer Simulation Conference*. SCS.

Mittal, S., & Zeigler, B. P. (2014a). Modeling attention-switching in resource-constrained complex intelligent dynamical system (RCIDS). In *Proceedings of Symposium on Theory of Modeling and Simulation/DEVS, Spring Simulation Conference*.

Mittal, S., & Zeigler, B. P. (2014b). Context and attention in activity-based intelligent systems. In *Proceedings of Activity-based Modeling and Simulation (ACTIMS14), ITM Web of Conferences* (vol. 3).

Mittal, S., & Zeigler, B. P. (2017). The practice of modeling and simulation in cyber environments. In A. Tolk & T. Oren (Eds.), *The Profession of Modeling and Simulation* (pp. 223–264). Hoboken, NJ: Wiley.

Nicholas Taleb, N. (2012). *Antifragile: Things That Gain from Disorder* (Vol. *3*). New York: Random House Incorporated.

Nutaro, J., Kuruganti, P. T., Shankar, M., Miller, L., & Mulle, S. (2008). Integrated modeling of the electric grid, communications, and control. *International Journal of Energy Sector Management*, *2*, 420–438.

Onggo, B. S., Mustafee, N., Juan, A. A., Molloy, O., & Smart, A. (2018). Symbiotic simulation system: Hybrid systems model meets big data analytics. In M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, & B. Johansson (Eds.), *Winter Simulation Conference* (pp. 1358–1369). New York: IEEE.

Pratt, A., Ruth, M., Krishnamurthy, D., Sparn, B., Lunacek, M., Jones, W., ... Marks, J. (2017). Hardware-in-the-loop simulation of a distribution system with air conditioners under model predictive control. In *Power & Energy Society General Meeting* (pp. 1–5).

Pratt, M. R. A., Lunacek, M., Mittal, S., Wu, H., & Jones, W. (2015). Effects of home energy management systems on distribution utilities and feeders under various market structures. In *Proceedings of the 23rd International Conference and Exhibition on Electricity Distribution*.

Rajkumar, R., Lee, I., Sha, L., & Stankovic, J. (2010). Cyber-physical systems: The next computing revolution. In *Proceedings of the 47th Design Automation Conference (DAC), 2010*, Anaheim, CA, (pp. 731–736). New York: IEEE.

Sanislav, T., & Miclea, L. (2012). Cyber-physical systems-concept, challenges and research areas. *Journal of Control Engineering and Applied Informatics*, *14*(2), 28–33.

Sheridan, T. B. (1992). *Telerobotics, Automation, and Human Supervisory Control*. Cambridge, MA: MIT Press.

Thule, C., Broman, D., Larcen, P. G., Gomes, C., & Vangheluwe, H. (2008). Co-simulation: A survey. *ACM Computing Surveys*, *51*, 420–438.

Tolk, A. (2015). Merging two worlds: Agent-based simulation methods for autonomous systems. In A. P. Williams & P. D. Scharre (Eds.), *Autonomous Systems: Issues for Defence Policymakers* (pp. 291–317). Norfolk, VA: NATO Allied Command Transformation.

Tolk, A., Adams, K. M., & Keating, C. B. (2011). Towards intelligence-based systems engineering and system of systems engineering. In A. Tolk & L. C. Jain (Eds.), *Intelligence-Based Systems Engineering* (pp. 1–22). Berlin, Heidelberg: Springer–Verlag.

Tolk, A., Page, E., & Mittal, S. (2018). Hybrid simulation for cyber physical systems: State of the art and a literature review. In *Proceedings of Annual Simulation Symposium, Spring Simulation Multi-Conference* (pp. 122–133).

Traoré, M. K., Zacharewicz, G., Duboz, R., & Zeigler, B. (2018). Modeling and simulation framework for value-based healthcare systems. *Simulation*, *95*(6), 481–497.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, *59*(236), 433–460.

Vangheluwe, H. (2000). Devs as a common denominator for multi-formalism hybrid systems modelling. In *IEEE International Symposium on Computer-Aided Control System Design* (pp. 129–134).

Williams, A. (2015). Defining autonomy in systems: Challenges and solutions. In *Autonomous Systems: Issues for Defence Policymakers* (pp. 27–62). Norfolk, VA: NATO Allied Command Transformation.

Williams, R. (2008). *Autonomous systems overview*. Technical report, BAE Systems.

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). *Data Mining: Practical Machine Learning Tools and Techniques*. Cambridge, MA: Morgan Kaufmann.

Zeigler, B. P., Praehofer, H., & Kim, T. G. (2000). *Theory of Modeling and Simulation. Integrating Discrete Event and Continuous Complex Dynamic Systems* (2nd ed.). San Diego, San Francisco, New York, Boston, London, Sydney, Tokyo: Academic Press.