Chapter

Introductory Chapter: Recent Trends in Systems-of-Systems Design, Modeling, Simulation and Analysis for Complex Systems, Gaming and Decision Support Final

Tien M. Nguyen and Tung X. Bui

1. Introduction

This book is a collection of some of the latest development work that embrace three major interrelated scientific disciplines – systems theory, decision theory and game theory, and the information and communication science with the goal to deploy innovative and effective computer-support application Systems of Systems (SOS). In addition, it also suggests the adoption of the Decision Support as a process-driven approach to iteratively engineer SOS. The book is a collective effort that uses a diverse set of studies using a wide spectrum of SOS applications to shed insights in the use of MS&A models to assist decision makers. Thanks to the exponential growth of big data related to all aspects of human activities, the surge in decision-making complexity due to the current climate of uncertainty with unforeseen consequences, and the increasing pervasiveness of advanced information and communication technologies (ICT) such as the proliferation of mobile apps, Internet-of-Things (IoT) and bots, we have witnessed an acceleration of integration of "complex systems" across of a wide range of application domains that include, but are not limited to, telecommunications, medicine (healthcare), military, manufacturing, transportation, energy, social networking platforms, education, and arts and culture. From Reference [1] the perspectives of system engineering, complex systems can be classified as:

- SOS Type 1: A family of system-of-systems that provides similar core services, e.g., communication services. But each system provides different core service types, e.g., non-secure FDMA vs. secure TDMA[1] communication services;

- SOS Type 2: An integration of many families of SoS. When combined, this type of system provides unique SOS capabilities at the enterprise level (i.e., integrated level). An example of this complex system is a combination of a family of communications SoS with a family of Global Position Satellite (GPS) SoS; and

[1] FDMA = Frequency Division Multiple Access vs. TDMA = Time Division Multiple Access.

- SOS Type 3: An integration of many heterogenous, independent but interrelated types of systems with each system providing distinctive core services. For example, a production line consists of (i) electrical system, (ii) sensor system, and (iii) mechanical system with belt conveyor, etc.

Note that we use U.S. Department of Defense (DoD) SOS System engineering guide [1] to define SOS. While many existing papers, documents and System-of-Systems (SoS, which is not SOS) standards considered integration of (i) many systems of the same type of systems together which is identical to our Type 1, and (ii) many different types of systems as a system consisted of many systems and referred to as SoS, which is identical to our SOS Type 3. In this chapter, we focus our discussion on SOS Type 2, since existing SoS engineering standards can be directly applied to SOS Type 1 and Type 3 but not Type 2.

It would be safe to claim that the disciplines of current system thinking, decision science and computer and telecommunication engineering have played a critical role in the emergence of SoS[2] and SOS. Traditional system theory posits that the whole is greater than the sum of its isolated parts [2–4]. And to achieve this holistic added value, it seeks to identify, analyze and create processes through optimal arrangements or adaptations of individual and independent subsystems' components and systems under system and SoS perspectives, respectively. This part-to-whole and whole-to-part thinking has been extended to SOS perspective for complex systems belong to SOS Type 2 and is prevalent in the selected chapters of this book.

Decision theory deals with the reasoning -- be it rational or not -- that drives a person's choice. The three core concepts in decision theory are elicitation and interpretation of the decision maker's preferences, the search of available options, and the management of uncertainty, risks and regrets [5–7]. In organizational or collective settings, decision making is extended to multiple stakeholders. Von Neumann, Morgenstern and Nash are universally credited for their pioneering work on game theory [8–10]. They proposed mathematical models of strategic interaction among rational decision-makers. The latter can be either cooperative or non-cooperative. The discussion on modern SOS in this volume constantly the basic foundations of game theory that uses Nash equilibrium as a prime operational goal.

The third underlying discipline that unifies the chapter of this book is the discipline of computer, information, communication and computer system (ICS) engineering. As ICS engineers continue to stay at the forefront of technological development, specific issues related to process flow design and conflict management of federated systems have emerged to be most challenging. The majority of chapters in this book, throughout their specific domain applications, such as space systems, attempt to address the challenges related to merging computing and communications due to the limitations of existing core protocols and at SOS design level, a deep understanding of how to conceptualize, design and manage coordinating parallelism [11, 12]. These design issues appear virtually in all chapters of this book. And the authors have demonstrated several different approaches to address these challenges in their specific applications.

As mentioned earlier, the authors use the DoD Systems Engineering Guide published in 2008 for defining SOS [1]. U.S. DoD has defined SOS as "a set or arrangement of systems that results when independent and useful systems are integrated into a larger system that delivers unique capabilities." SOS development involves the creation of systems, which are collections of legacy, evolving and new systems that must have a high degree of flexibility and adaptability. Two of the main considerations for defining SOS are:

---

[2] Note that in this Chapter SoS can be considered as SOS Type 1 and Type 2 but not Type 3. Existing standard SoS engineering approach needs to be modified or extended to address SOS Type 2 enterprise challenges.

i. The lack of authority over the constituent systems because of their independent management, funding and objectives that may not align with those of the SOS as a whole.

ii. The emergent behavior adds a large degree of unpredictability, as overall system behavior cannot be predicted by having individual knowledge of each of the constituents.

Recently in 2019, the International Council on Systems Engineering (INCOSE) [13] has released the latest version of the "Guidelines for the Utilization of ISO/IEC/IEEE 15288 in the Context of System of Systems (SoS) Engineering", which is an extension of existing system engineering, to industry for review and comments. This document provides guidance for the utilization of ISO/IEC/IEEE 15288 in the context of SoS in many domains. This INCOSE guidebook perceives that SoS engineering demands a balance between linear procedural procedures for systematic activity and holistic nonlinear procedures due to additional complexity from SoS emergence. ISO/IEC/IEEE 15288 is considered by most system engineers as the foundation for SoS Engineering for many civilian and commercial applications, while U.S. DoD Systems Engineering Guide for SOS is considered as the foundation for SOS engineering design and implementation for defense applications.

The following sections address the objectives presented in the abstract section.

2. Current trends on SOS engineering

Using a requirement-based approach, existing System-of-Systems (SoS) engineering is a software engineering discipline that deals with the analysis, design, development, deployment and evaluation of heterogenous systems. A peculiar aspect of SoS engineering is to deal with a significant level of uncertainty in requirements engineering [13]. As such, while the goal of traditional system engineering is to build the system right, SoS engineering goal is to build the right system in the SoS context. Instead of optimizing individual systems, the primary objective of SoS engineering methodology is to maximize the overall performance of an integrated platform.

Many researchers, scientists and system engineers in the field have recognized the following challenges related to existing requirement based SoS approach when extended to complex SOS Type 2 defined above: (i) Requirements are usually derived from the assumptions of certain selected technology enablers that are usually a few years behind the current technologies, and (ii) When the requirements are changing at a fast pace, the current (As-Is) system architecture design using this SoS requirement-based approach becomes a "bottleneck" when interfacing with newer (To-Be) systems that are not within the same family. The current SOS engineering approach is to focus on the use of capability-based SOS engineering approach to address the "requirements" challenges for monolithic and complex systems (SOS Type 2 or complex SOS) with changing requirements due to technology changes and dynamic environments.

3. Current trends on SOS architecture design and MS&A

Known as complex SOS, these new deployments represent a step further in integrating task-driven dedicated systems that pool and share resources (i.e., data) and capabilities (i.e., modeling and intelligence) together to expand more functionalities and improved SOS performance (i.e., effectiveness and efficiency) to deliver unique capabilities than simply the sum of the individual constituent systems. Many of these SOS appear invisible to most users, but their presence is omnipresent in our daily lives: smart power grids, integrated traffic networks for air, land and sea connectivity, health information systems, global supply chain networks, and many others.

The two key parts of using existing standard requirement based SoS engineering for extension to SOS are to: (i) design a complex SOS Type 2 architecture, and (ii) develop MS&A models to characterize and predict the behavior of the complex SOS Type 2 architecture. Like SoS perspective, in SOS, most if not all individual systems and systems' components are already in place. Therefore, the development of a complex SOS Type 2 architecture requires a full consideration of the technical feasibility of the constituent systems within Family of Systems (FoS), systems within different types of FoS, and the compatibility between the SOS requirements (i.e., systems of different FoS' systems) and those of individual systems' components. Practically, the complex SOS architectural focus could be just to establish adequate communication protocols between systems and ensure proper interoperability. If the SOS requirements is distinctively unique, a more elaborate SOS architecture design is required to modify some existing constituent systems to achieve overall SOS effectiveness. As such, the task of SOS architecture design and analysis is to strike a balance between the enterprise goals of all involved stakeholders. Reference [14] employs this SoS requirement-based approach to address architecture design challenges from mission definition to architecting complex SOS. However, for complex SOS Type 2 architecture with different FoS types, with requirements that are not stable due to the technology changes and dynamic operational environment conditions for one type of FoS as compared to the other types of FoS, the requirement-based design approach may lead to a stove-pipe architecture solution due to the following reasons:

- As mentioned above, requirements are usually derived based on the assumptions of selected technology enablers that are usually a few years behind the current technologies (e.g., at least 3 to 5 years for space systems),

- When the requirements of one type of FoS are changing at a fast pace, the As-Is system architecture design using SOS requirement-based approach becomes a "bottleneck" when interfacing with newer systems that are not within the same family,

- The family of SoS that requires frequent upgrade/refresh due to technology changes can cause a loss of interoperability with family of SoS that have stable requirements,

- Practically, SOS management at the enterprise level can pose a real challenge, when a family of SoS systems with stable requirements are not able to synchronize with the other families of SoS systems to be deployed at a later date; Managing this misalignment of systems and requirements synchronization is unmanageable task.

As discussed earlier, the current trends are to extend SoS requirement-based engineering to SOS capability-based engineering approach for the design and development of MS&A of complex systems. These current trends are based on the following capability-based engineering concepts:

- SOS architecture design is based on the top-down approach by associating each system with its high-level capabilities not subsystem's components' requirements. This approach leaves the flow-down of capability-to-requirement to the selected contractors for the design and build of each FoS,

- Managing the capabilities at the SOS enterprise level, i.e., SOS capability alignment, synchronization and integration will be managed using capability-based engineering approach.

- Develop MS&A models to simulate, analyze and characterize SOS capabilities with well-defined SOS architecture performance metrics focusing on capabilities, and not detail requirements' metrics.

4. Existing SOS Modeling, simulation and analysis (MS&A)

The current trends for the development of SOS MS&A models requires to classify and decompose complex SOS according to their presumed capability's attributes and relationships. In the context of this chapter, the classification and decomposition will be performed using SOS capability-based engineering perspective, which is also referred to as SOS perspective that encapsulate the three SOS types as defined earlier. Using [15], the book chapters have presented SOS taxonomy for space and airborne systems.

Optimizing the synergy between independent and heterogeneous systems or FoS' systems is a particular challenge in SOS architecture analysis. In order to analyze the SOS architecture effectiveness and to specify SOS characteristics, behaviors and features, engineers develop SOS MS&A frameworks and models and associated SOS performance metrics. The book chapters successfully demonstrated the use of SOS perspective for the development of MS&A models to characterize the performance of a notional space systems-of-systems with different FoS types.

5. A SOS perspective and current trends on the MS&A of decision support systems (DSS)

Decision-making is being profoundly challenged by the digitization of the business world and the rise of environmental uncertainty and risks. Augmented by digital technologies, decision makers have in their hands massive amount of open source data, and often, they are forced to make swift decisions while trying to mitigate increasing level of risks. Given the diversification of massive information sources that go beyond the organization boundaries, decision makers are facing with a triple level of uncertainty – increased difficulty in identifying the possible courses of actions given a complex decision; increased difficulty in estimating the likelihood of decision outcomes of a chosen action, and unexpected emergence of new actors – be it allies or foes – along the decision-making process.

DSS are commonly known as computer-based systems that are designed for aiding decision-makers to transform an ill-defined and unstructured problem into a well-defined and structured problem. In that process, the problem can be iteratively analyzed, and possible decision outcomes can be visualized. The role and function of a DSS is to guide the user throughout the decision-making process that eventually lead to a final decision – either optimal or satisfactory. In this DSS perspective, MS&A is seen as a set of tools that help decision makers to (i) express their preferences and needs, (ii) explore all possible solutions, (iii) perform evaluation analyses, and (iv) select the best possible preferred outcomes. In complex decision-making situations, a DSS is a computer-based system that supports its user(s) to make effective decisions in ill-structured problems.

A DSS is typically composed of four components [4]:

- A Data Component: that allows the decision maker to retrieve relevant data from all possible relevant sources to generate new alternatives and new perspectives in decision making;
- A Model Component: that enables selection, creation, and manipulation of models to help decision makers best capture the dimensions of the decision problems;
- An Interface Component: that provides context-sensitive and personalized interaction with the users; and
- A Communication Component: that facilitates structured and free-format information exchanges between users.

In a SOS environment, the level of sophistication of these four components increases with the size of interconnected systems. The overarching SOS engineering challenge is to deal with the uncertainty related to the design of interoperability and coordination. It is critical to design context-dependent interfaces between individual systems using technologies such as the Unified Modeling Language (UML), with the need for integration and coordination:

- SOS Dynamic Program Solver: Given a specific complex problem at hand, the DSS facilitates the generating of problem-solving solutions. It can simply be the execution of an existing ready-to-be-applied decision model, or it can set up a sequence of trial-and-error algorithms in search for an optimal problem formulation.
- SOS Daemon: In a highly distributed system, the role of this kernel of the multi-tasking DSS is to manage, coordinate and control in the background, without the direct control of the user, the overall execution of the SoS. Upon request, the daemon activates the processing of data, model and analysis, interpretation and visualization.
- SOS Scheduler: In a large-scale, loosely coupled distributed architectures, the SoS scheduler task is to generate a feasible scheduling scheme to execute decision making problems. It is providing process-driven policies to help the SOS Deamon to sequence DSS execution – either in a concurrent or sequential manner – and to allocate computational resources to each of the DSS tasks.
- SOS Planner and Controller: In highly vulnerable operations, the Planner and Controller performs the function of a hardware/software watchdog to ensure that the SoS can be promptly reset if it is disrupted by malfunctions or failures. In normal operations, the DSS should allow for the coordination of modeling and simulation processes.

Over the last thirty years, with the emergence of Decision Support Technology, decision makers have benefited from dedicated DSS applications, from engineering to business and healthcare to engineering. Elevating the role and function of DSS to support complex SoS presents both opportunities and challenges to the discipline. As exemplified in the book chapters, the concept of DSS can offer group decision for risk management and collective generation of social urban policies. Furthermore, game theories can be used to engage antagonists in finding a solution that would be acceptable to all.

6. A SOS perspective and current trends on MS&A of game theoretical models (GTM)

The development of computer models to perform the MS&A of a particular game theoretical model, whether it is used for analyzing and predicting the behaviors of complex systems or future economic well-being or forecasting stock market, are very challenging for system analysts, economic analysts, financial analysts and mathematicians. As pointed out in the book chapters, practical GTM are usually very complex because these models require the analysts and mathematicians to have a deep understanding of a combination of:

(i) Game theory, i.e., what type of game that the analyst should be selected for the problem being investigated and how-to set-up the game to achieve desire outcomes,

(ii) Advanced mathematics[3] in several fields, including modeling and simulation, algebra, probability, statistics and calculus, and

(iii) The problems being investigated with profound subject matter expertise in a particular scientific area. As an example, the analyst may require having a thorough understanding of the behavior of the cellular system being investigated or understanding of several medical fields, including tumor cells, thoracic surgery, neocortical epilepsy and kidney donation.

Currently, researchers, scientists, engineers and analysts have applied game theory to many applications, including but not limited to health care, wireless communications systems, complex systems acquisition, finances, and economics. As mentioned in Section 4 above, for the development of SOS MS&A models using SOS perspective, a game theoretical modeling taxonomy is required. **Figure 1** provides a description of a game theoretical taxonomy that the SOS modelers can use in selecting proper game "engine" (or system in SOS context) to perform required SOS MS&A modeling tasks. As shown in **Figure 1**, there are two types of games, namely, static games and dynamic games. For each game type, depending on the information available to the players, the MS&A modelers can choose to play complete or incomplete information with either corporative or non-corporative games. For static games, the modelers develop game engines using normal form. While, dynamic games use extensive form with the complete information games being divided into perfect information and imperfect information games. The definitions for these games are given below:

- Complete information games: Players know all information about the other players, e.g., their "types", strategies, payoffs and preferences.
- Incomplete information games: Players may or may not know some information about the other players, e.g., their "types", strategies, payoffs or their preferences.
- Perfect information games: Players are aware of the actions chosen by other players. They know who the other players are, what their possible strategies/actions are, and the preferences/payoffs of these other players. Information about the other players in perfect information game is also complete.

[3] Note that mathematical game theory can also be considered as a brand of mathematics that required mathematical background in modeling and simulation, algebra, probability, statistics and calculus.

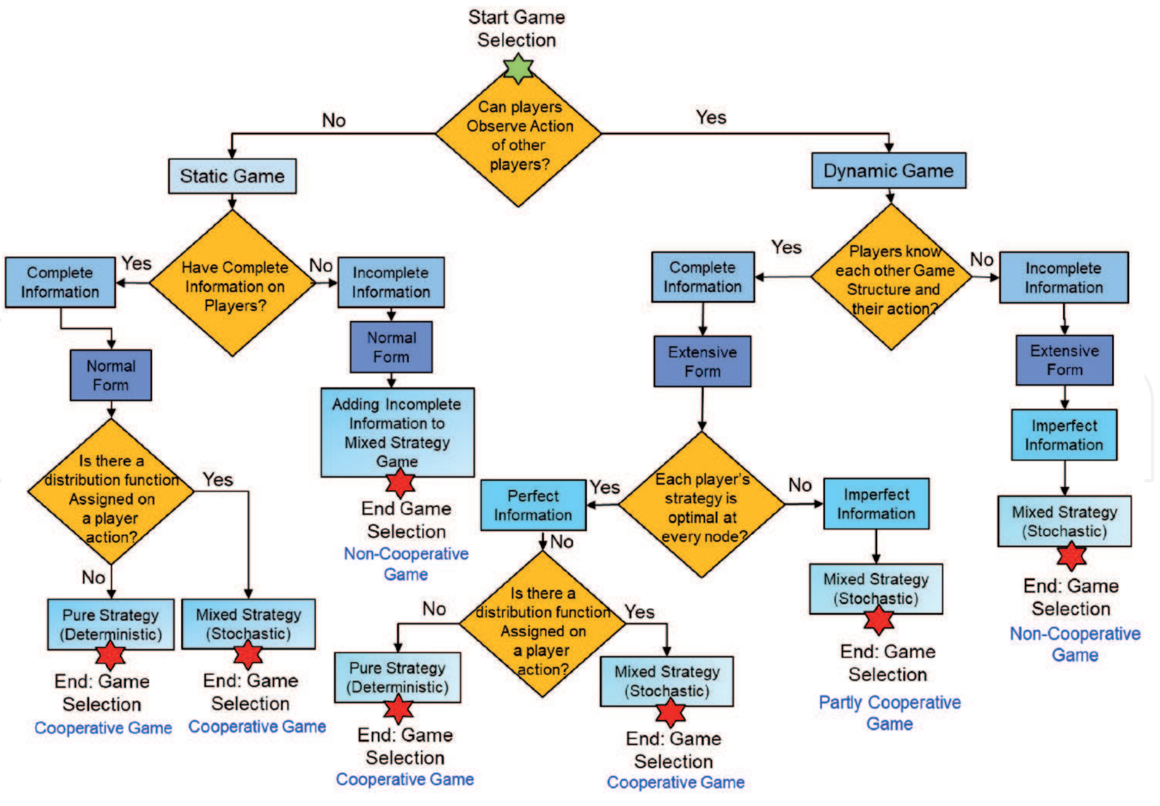

Figure 1.
Game theoretical Modeling Taxonomy from SOS perspective.

- Imperfect information games: Players simply do not know of the actions chosen by other players. However, they know who the other players are, what their possible strategies/actions are, and the preferences/payoffs of these other players. Hence, information about the other players in imperfect information game is incomplete.

Figure 1 assumes all games are Bayesian games. Basically, there are seven game models associated with static and dynamic game types. Each of these game models can be considered as a model of a "system" with a specified set of inputs and outputs and when these systems are set-up properly, together they will provide desired outcomes. Recently, References [16, 17] have used this SOS MS&A approach for developing complex computer simulation models to evaluate and develop acquisition strategy for acquiring complex space systems for U.S. DoD. Reference [18] discusses the use of decision support system to complement the game models when decisions on a selecting of a particular architecture solution does not converge. Therefore, it is probably safe to claim that the current trends for the development of complex MS&A models for game theoretical modeling applications are to use (i) SOS perspective, and (ii) Decision support system when the game model of a complex system or process does not converge.

7. Conclusion

The chapters presented in this technical book share a common thread of using Systems-of-Systems (SOS) perspectives and existing INCOSE System-of-Systems (SoS) engineering to address complex practical test-and-evaluation processes and systems, ranging from melt spinning process and space systems enterprise to medical problems. While dealing with a great level of details illustrating their thematic methodologies, the authors do take the effort in providing the readers with a thorough background discussion, making their chapters comprehensive and reachable to all. We hope that the readers will find the chapters selected for this book a source of ideas for their own work -- ideas that are anchored in a set of inter-disciplinary and, moreover, ideas are being explored or experimented in promising systems-of-systems applications.

By its very nature, an SOS evolves overtime with changing systems requirements. Looking forward, the coordination of simultaneous and real-time processes in large-scale programmable networks remains to us a major challenge in the design of SOS for complex problem solving. In spite of the recent advances in Artificial Intelligent (AI) and automated systems, the burden of managing the interfaces between distributed data repositories, disparate and incompatible model bases, and context-dependent visualization and presentation techniques, the coordination of parallel operations still depends in a large part on the SOS designers. It is a time-consuming and tedious process to conceptualize and implementing parallelism. We hope the readers find in this book concepts and ideas that would help them achieve their requirements analysis and SOS design task.

Last but not least, and although not explicitly discussed, most of the chapters in this book have judiciously pointed out the necessity of establishing a coherent set of principles for systems-of-systems engineering, including synergism, symbiosis, modularity and self-governance, conservation and reconfiguration, emergence and re-architecting, efficiency and effectiveness. If these principles are clearly defined and adopted by the community of SOS developers and users, we expect that SOS would be the next and ultimate generation of digital applications.

Author details

Tien M. Nguyen[1,2]* and Tung X. Bui[3,4,5]

1 Center for Computational and Applied Mathematics (CCAM), California State University, Fullerton (CSUF), USA

2 The Aerospace Corporation, USA

3 Shidler College of Business, USA

4 Information Technology Management, University of Hawaii at Manoa, USA

5 Haas School of Business, University of California at Berkeley, USA

*Address all correspondence to: tmnguyen57@fullerton.edu

References

[1] U.S. DOD, Systems Engineering Guide for Systems of Systems, Version 1.0, August 2008. Available at: https://acqnotes.com/wp-content/uploads/2014/09/DoD-Systems-Engineering-Guide-for-Systems-of-Systems-Aug-2008.pdf

[2] R.W. Asby, Introduction to Cybernetics. London: Routledge Kegan & Paul, 1964.

[3] L. Von Bertalanffy, General System Theory: Foundations, Development, Applications. New York: George Braziller, 1968.

[4] Tung X. Bui, Co-oP: A Group Decision Support System for Cooperative Multiple Criteria Group Decision Making. Berlin: Springer, 1987.

[5] H. Raiffa, Decision Analysis: Introductory Lectures on Choices under Uncertainty. Massachusetts: Addison-Wesley, (968.

[6] H. Simon, The Sciences of the Artificial. Massachusetts: MIT Press, 1969.

[7] R. Schlaifer, Analysis of Decisions under Uncertainty. New York: McGraw-Hill, 1969.

[8] J. Von Neumann and O. Morgenstern, Theory of Games and Economic Behavior. New Jersey: Princeton University Press, 1953.

[9] J.F. Nash, The Bargaining Problem. In Econometrica. Econometric Society. 18(2): 155-162, 1950.

[10] H.F. Von Stackelberg Market Structure and Equilibrium, 1934. 1st Edition (translation into Englsih), Bazin, Urch & Hill: Springer (2011).

[11] H. Kopetz, System of Systems Challenges. In: Schoitsch E. (eds) Computer Safety, Reliability, and Security. SAFECOMP 2010. Lecture Notes in Computer Science, vol 6351. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15651-9_35.

[12] A.M. Pendleton-Julian and J.S. Brown, Design Unbound. Massachusetts: MIT Press, 2018.

[13] INCOSE, New Standards for System-of-Systems (SoS) Engineering. Available at: https://conference. conflr.com/events/Demo/showcases/incose/INCOSE%20System%20of%20Systems%20Engineering%20Standards.pdf.

[14] Imane Cherfa, Nicolas Belloir, Salah Sadou, Régis Fleurquin and Djamal Bennouar, "Systems of Systems: From Mission Definition to Architecture Description," Wiley Systems Engineering. 2019; 1-18. Available at: https://wileyonlinelibrary.com/journal/sys.

[15] J. Gideon et al., "Taxonomy of Systems-of-Systems," Conference on Systems Engineering Research, Institute of Electrical and Electronics Engineers (IEEE), Jan 2005.

[16] Tien M. Nguyen, Andy T. Guillen, Sumner S. Matsunaga, Hien T. Tran and Tung X. Bui, War-Gaming Applications for Achieving Optimum Acquisition of Future Space Systems, Book Chapter in Simulation and Gaming Book, 2018, IntechOpen Publisher. Available at: https://www.intechopen.com /books/simulation-and-gaming/war-gaming-applications-for-achieving-optimum-acquisition-of-future-space-systems.

[17] Tien M. Nguyen, Hien T. Tran, Andy T. Guillen, Tung X. Bui, and Sumner S. Matsunaga, "Acquisition War-Gaming Technique for Acquiring Future Complex Systems: Modeling and Simulation Results for Cost Plus

Incentive Fee Contract," Journal of Mathematics, 2018. Available at: https://doi.org/10.3390/math6030043.

[18] Tien M. Nguyen, Tom Freeze, Tung Bui, Andy Guillen, "Multi-Criteria Decision Theory for Enterprise Architecture Risk assessment: Theory, Modeling and Results," 2020 SPIE Conference Proceedings, Conference: Sensors and Systems for Space Applications XIII; DOI:10.1117/12.2559317.