Architectural Complexity Analysis for Large-scale Emergency Rescue Management Systems: A Preliminary Study

Liang Gao*†, Martijn Warnier†, Sander van Splunter†, Long Chen* and Frances M.T. Brazier†
*China Research and Development Academy of Machinery Equipment
Beijing, P.R. China
Email: gaoliang_box@126.com
†Section Systems Engineering, Delft University of Technology
Delft, The Netherlands
Email: {M.E.Warnier, s.vansplunter, F.M.Brazier}@tudelft.nl

***Abstract*—Architectural complexity analysis plays an important role in the design of complex systems and System of Systems (SoS). In this process, a central problem is how to define the complexity indicator in the SoS level. This paper discusses a preliminary study on the architectural complexity analysis of the BRIDGE system, a typical large-scale emergency rescue management system. The traditional definition of architectural complexity, which has been successfully used to describe the architectural complexity of military SoS, is not suited for this case. This definition of architectural complexity, which consists of the multiplication of several components of complexity indicators, does not consider the varying importance of each component in different types of complex systems and SoS. This paper proposes a more general definition of architectural complexity by introducing additional exponential weighting factors to model such domain-related effects. Experts ratings on the relative importance of pair-wise components to the architectural complexity are used to estimate these weights. This new definition of architectural complexity facilitates the incorporation of the subjective domain-related knowledge and thereby provides a more flexible and reasonable measurement on the architectural complexity of complex systems and SoS.**

Keywords*—Complex Systems, SoS, Architectural Complexity, Large-scale Emergency Rescue Management Systems, Alternative Designing

I. INTRODUCTION

With the rapid development in systems engineering and the urgent application needs in industry, System of Systems (SoS), which can be seen as high-level complex systems, have been attracting more and more attention of researchers [1–18]. Accordingly, System of Systems Engineering (SoSE), which is described as “The design, deployment, operation, and transformation of metasystems that must function as an integrated complex system to produce desirable results” [1], has been developed as a promising response for analysis, design, and transformation of increasingly complex systems problems [2].

It is well known that complex systems problems and SoS issues are different but closely related. Boardman *et al.*[3] presented the distinguishing characters of SoS and the differences between complex systems and SoS. Keating *et al.*[2] analyzed the relationship between complex systems and SoS, along with framing questions and related topical areas. Undoubtedly, the developments in complex systems and SoS will greatly accelerate the research on each area respectively.

In the research of complex systems and SoS, a critical issue is how to supply useful and timely decision making support during the conceptual design and architecture phases by using advanced methodologies and tools. This problem has received a wide range of academic and industrial attention [14–19]. Both systems architecture designers and decision makers often have to evaluate complex systems and SoS based on certain indicators and criterion. The proposed methodology ideally will help both of these to make the design and decision process from an overall perspective more efficiently.

Due to the high complexity that complex systems and SoS possess, the proper definition and evaluation of relevant indicators from a high-level view becomes a basic and very important problem. Once such indicators are determined, they can be used to assess the complex system or SoS so that the decision making support can be improved by considering all relevant criteria. Some indicators on the system-architecture level have been successfully used to analyze a specific SoS, such as architectural complexity [14], and the total completion time to a typical kill chain [15].

Many practical examples on complex systems and SoS have been developed, from different methodology and application points of view. An example is the BRIDGE[1] system [19, 20]. This system aims to bridge different resources and agencies in large-scale emergency rescue management and to supply strong decision support to hundreds of professionals from various organizations. References [19, 20] introduce a typical incident command organization approach. A fictitious example use-case named “The Chemical Incident”, is used to explain how the BRIDGE collaboration technologies can support organizations in a large-scale crisis response scenario.

For the research on the BRIDGE system, less complexity is preferred in many cases. It is important to study how to make such decision support systems more efficient and less complex. On the other hand, the architectural complexity can also be thought of as an overall indicator and be used to evaluate different alternatives of the BRIDGE system architecture. Thus,

---

[1] http://www.bridgeproject.eu/en

how to analyze the architectural complexity of the BRIDGE system, and other emergency rescue management systems, is a fundamental problem: the problem this paper addresses. The remainder of the paper illustrates how the complexity of large-scale emergency rescue management systems can be determined using the BRIDGE system as an example.

II. THE BRIDGE SYSTEM: AN EXAMPLE OF A LARGE-SCALE EMERGENCY RESCUE MANAGEMENT SYSTEM

The BRIDGE system is developed in the BRIDGE project, which is a collaborative project co-funded by the European Commission within the Seventh Framework Programme [19, 20]. It aims to increase safety of European citizens by developing technical and organizational solutions that significantly improve the capability of crisis and emergency rescue management.

The BRIDGE system is a model-based automated support system as well as an agent-based dynamic workflow composition and communication support system. This system consists of several components, including front ends or applications, back ends and configuration tools, collaborating systems, information repositories and networking tools. By using the BRIDGE system, different people from different organizations can efficiently collaborate with each other and complete various activities in large-scale emergency rescue actions.

To demonstrate how the BRIDGE system works in a large-scale emergency rescue task, a use-case based on a fictitious emergency rescue scenario, i.e., ”The Chemical Incident”, is made [19]. Table I depicts an overview on how the BRIDGE system plays its role in the four different rescue phases, namely, initial phase, establishing phase, implementation phase and international collaboration. At different time-points, the different components of the BRIDGE system, i.e., BRIDGE Master, BRIDGE Resource Manager and BRIDGE Information Aggregator, are employed by different roles for communication and cooperation, including the Operational Management Team, the Search & Rescue Team and the Incident Commander.

Four principal activities are involved in the whole rescue workflow, i.e.

- 1.0 Victim Tracking
- 2.0 Resource Acquisition and Logistics
- 3.0 Evacuation Support
- 4.0 Assessment and Awareness Support

These activities can be further divided into a sequence of lower level functions to be completed, as illustrated in Fig. 1. The mapping of the different functions, that each available system can perform, are also presented in Table II. Listed in parentheses next to each available system is the total number of functions that each system can perform. Table II also gives the system abbreviation and system type.

In the architecture designing of the BRIDGE system, it is important to know which alternative is better under certain criteria. This requires the evaluation of alternative architectures from different system groupings. For this purpose, this paper proposes a novel measure of architectural complexity on the BRIDGE system. The details will be discussed in section III.

III. ARCHITECTURAL COMPLEXITY ANALYSIS ON THE BRIDGE SYSTEM

The following measure is defined to determine the architectural complexity of the BRIDGE system,

$$C = \left(\sum_{i=1}^{N_1} P_i^2/N_1\right)^{\alpha} \times \left(\sum_{j=1}^{N_2} ICM_j\right)^{\beta} \times (1+\log Q)^{\gamma}\,. \quad (1)$$

The meanings of symbols in Eq.(1) are given as below:

C	- Architectural complexity
α	- Exponential weighting factor
β	- Exponential weighting factor
γ	- Exponential weighting factor
N_1	- Number of functionally & physically distinct systems
N_2	- Number of network interfaces used to transmit data/information (both internal and external to the separate physical systems)
P_i	- Number of functions performed by the ith system
ICM_j	- Interface Complexity Multiplier
Q	- Cycloramic Complexity

In Eq.(1), $\sum_{i=1}^{N_1} P_i^2/N_1$ is the Node Complexity Distribution (NCD) term, $\sum_{j=1}^{N_2} ICM_j$ is the Interface Complexity (IC) term, and $1+\log Q$ is the Cycloramic Complexity (CC) term. The definition of architectural complexity in Eq.(1) is partially motivated by the complexity studies in [14, 21–23]. It is worth to point out that the complexity definition in [14] is a special case of our definition.

The purpose of introducing α, β and γ as three exponential weighting factors is to further incorporate domain-related knowledge and the experiences of experts in the measure of architectural complexity. The definition in Eq.(1) not only reflects objective evaluation factors, i.e., NCD, IC and CC, but also incorporates subjective measures that can be obtained from different stakeholders in the overall architecture design, *e.g.*, engineers from integrated and sub-systems design, the BRIDGE system users, and decision makers from different government departments. In comparison with the complexity definition in [14], Eq.(1) aims to better reflect the practical designing process. It can be used for alternatives analysis and evaluation on various types of complex systems and SoS.

In the measure of the architectural complexity of the BRIDGE system, only the NCD and the CC term are considered, since interface complexity is too complicated to be evaluated in this case. Specifically, by choosing $\beta = 0$ the IC term in this study is cancelled. Based on Table II, the NCD term of the BRIDGE system can be calculated, as follows :

$$\begin{aligned}\sum_{i=1}^{N_1} P_i^2/N_1 =&[(6^2+5^2+10^2+6^2+5^2+10^2+4^2+3^2+4^2\\ &+4^2+10^2+2^2+1^2+2^2+1^2+5^2+2^2)]/17\\ =&30.4706\end{aligned} \quad (2)$$

TABLE I. THE OVERVIEW OF THE BRIDGE SYSTEM IN THE SCENARIO

No.	Phase in the Rescue Work	Time	Components of the BRIDGE system	People in the Rescue Work
1	Initial Phase	10:02	BRIDGE Master	Operational Management Team, Search & Rescue Team
		10:05	BRIDGE Resource Manager	Operational Management Team
		10:08	BRIDGE Resource Manager	Operational Management Team, Incident Commander
		10:05-10:10	BRIDGE Information Aggregator, BRIDGE Master	Operational Management Team
		10:15	BRIDGE Planner, BRIDGE Risk Modeller	Operational Management Team
2	Establishing Phase	10:20	BRIDGE Master	Incident Commander
		10:20	BRIDGE e-triage RFID	Incident Commander
		10:20	BRIDGE Master	Incident Commander
		10:20	BRIDGE Mesh	Search & Rescue Team
		10:20	BRIDGE Master	Incident Commander
		10:20	BRIDGE Planner	Incident Commander
		10:20	BRIDGE Risk Manager	Incident Commander
		10:20	BRIDGE RescueMe	Persons in Chemo
		10:20	BRIDGE Mesh	Persons in Chemo
		10:20	Repositories of the BRIDGE System	Persons in Chemo
		10:20	BRIDGE Master	Persons in Chemo
		10:28	BRIDGE RescueMe	Incident Commander
		10:28	BRIDGE Planner	Incident Commander
		10:28	BRIDGE Risk Manager	Incident Commander
		10:30	BRIDGE Master	Incident Commander, Incident Commander Team, Operational Management Team
		10:32	BRIDGE Master	Incident Commander, Operational Management Team
		10:34	BRIDGE Master, BRIDGE Resource Manager, Risk Modeller and Planner	Incident Commander, Deputy of Incident Commander
		10:34	BRIDGE Public Information Collector Group, Risk Modeller	Operational Management Team
		10:36	BRIDGE Master	Incident Commander
		10:36	Risk Modellers	Incident Commander Team
		10:36	BRIDGE Master	Incident Commander Team
3	Implementation Phase	10:51	BRIDGE Mesh	The Technical Assistant (Incident Commander Staff)
		10:51	BRIDGE Communication Manager	The Technical Assistant (Incident Commander Staff)
		10:51	BRIDGE Information Collector Group	Incident Commander
		11:05	BRIDGE Planner	Incident Management Team
		11:10	BRIDGE Risk Modeler	Operational Management Team
		11:10	BRIDGE Distributed Expertise Integration Network	Operational Management Team
		11:10	BRIDGE Beacons	The Search and Rescue Team
		11:15	BRIDGE Expert Network Builder	Operational Management Team
		11:15	BRIDGE Risk Modeller	Experts
		11:40	BRIDGE Broadcaster	Media Officer in Operational Management Team
		11:40	BRIDGE Risk Manager	Operational Management Team
		11:40	BRIDGE Master	Incident Commander
		11:40	BRIDGE Planner	Medical Staff
		11:40	BRIDEG eTriage tools	Medical Staff
		11:40	BRIDGE Monitoring System	Medical Staff
		11:40	BRIDGE RescueMe	Commanding Staff
4	International Collaboration	14:30	Helicopter	A Task Force (The Dutch Fire Fighters)
		14:30	Repositories of the BRIDGE System BRIDGE Master	The Dutch Fire Fighters Organization
		14:30	BRIDGE Risk Manager	The Dutch Fire Fighters Organization
		14:30	BRIDGE Resource Manager	The Dutch Fire Fighters Organization
		14:30	BRIDGE Planner	The Dutch and the German Incident Commander, The Dutch and German Fire Fighting Forces
		14:30	Sensors	The Dutch Fire Fighters Organization
		14:50	Sensors	First Responder
		14:50	Gas Masks	First Responder
		14:50	Gas Detectors	First Responder

To analyze the CC term of the BRIDGE system, one first needs the program control graph for the BRIDGE system, as illustrated in Fig. 2. Each element (Row i, Column j) ($i = 1, \cdots, 19; j = 1, \cdots, 9$) represents a component of the BRIDGE system along with the corresponding sub-function that each component can perform. Each element can be considered as a node in the program control network. There exist node connections between nodes in two adjacent rows. For example, there are five edges between (Row 1, Column 1) and (Row 2, Column i) ($i = 1, \cdots, 5$) and so on. The number of total edges and total nodes are also listed in the last two columns in Fig. 2.

Using Fig. 2, the CC term of the BRIDGE system can be calculated as follows:

$$\begin{aligned} 1 + \log Q =& 1 + \log(e - n + 2p) \\ =& 1 + \log(265 - 80 + 2 \times 1) \\ \approx& 3.27 \end{aligned} \tag{3}$$

where e represents the total number of edges in Fig. 2, n is the total number of nodes in Fig. 2 and generally $p = 1$ [14].

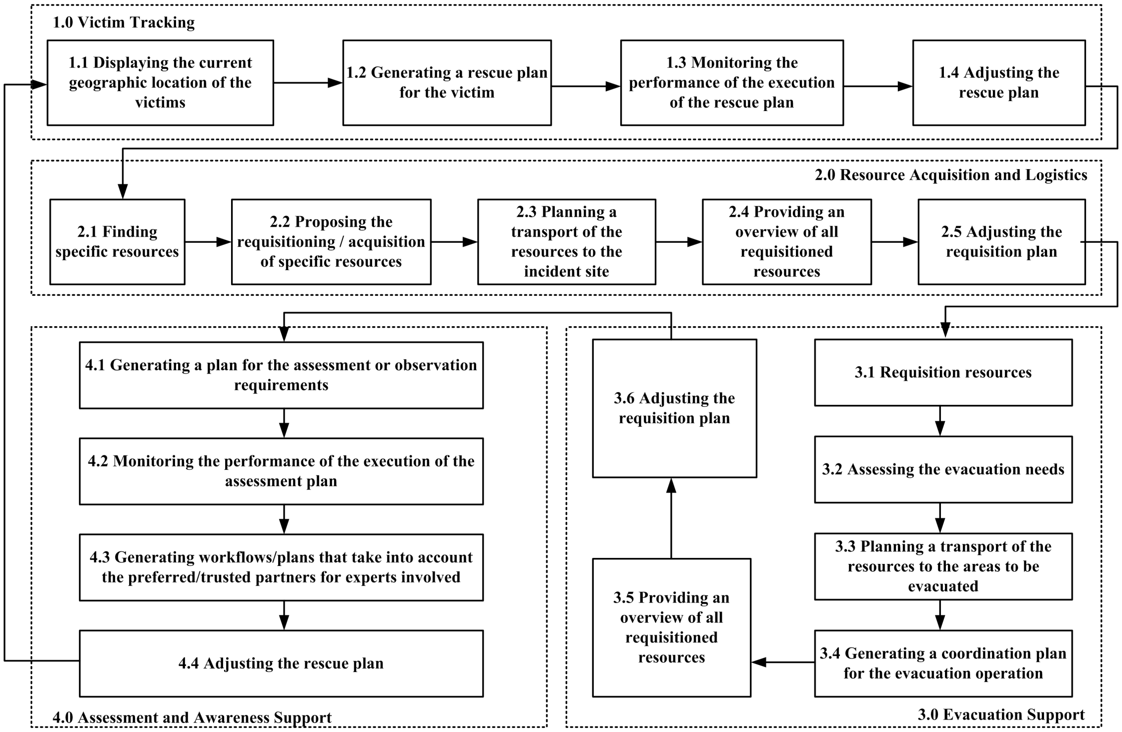

Fig. 1. Activity flow diagram for "The Chemical Incident".

TABLE II. AVAILABLE SYSTEM-TO-FUNCTION MAPPING

Available System	**System Abbreviation**	**System Type**	**Function**																		
			1.1	1.2	1.3	1.4	2.1	2.2	2.3	2.4	2.5	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	4.4
BRIDGE Resource Manager (6)	ReMa	Providing Information on Resources, Part of BRIDGE Master	X				X	X		X		X				X					
BRIDGE Information Aggregator (5)	IA	Combining Incoming Situation Reports						X		X		X	X			X					
BRIDGE Planner (10)	Planner	A Command Tool for Planning and Coordinating Personnel Tasks and Responsibilities, Part of BRIDGE Master		X		X			X		X			X	X		X	X		X	X
BRIDGE Risk Modeller (6)	RiMo	Part of BRIDGE Master		X									X	X	X			X		X	
BRIDGE e-triage RFID (5)	RFID	Collaborating System					X	X		X						X				X	
BRIDGE Mesh (10)	Mesh	Network	X	X	X		X	X		X		X	X			X			X		
BRIDGE Risk Manager (4)	RiMa	Part of BRIDGE Master						X		X			X							X	
BRIDGE RescueMe (3)	RescueMe	Collaborating System	X	X																	X
Repositories of the BRIDGE System (4)	Repositories	Information Repositories					X			X		X				X					
BRIDGE Public Information Collector Group (4)	PICG	Part of BRIDGE Master						X		X		X				X					
BRIDGE Communication Manager (10)	CM	Managing the Network and Supporting the Construction of Communication Infrastructure	X	X	X		X	X		X		X	X			X			X		
BRIDGE Distributed Expertise Integration Network (2)	DEIN	Part of BRIDGE Master													X					X	
BRIDGE Beacons (1)	Beacons	Different Sets of Integrated Sensors (Collaborating System)	X																		
BRIDGE Expert Network Builder (2)	ENB	Part of BRIDGE Master													X					X	
BRIDGE Broadcaster (1)	Broadcaster	Part of BRIDGE Master	X																		
BRIDEG eTriage tools (5)	eTriageT	Collaborating System					X	X		X						X				X	
BRIDGE Monitoring System (2)	MS	Part of BRIDGE Master			X														X		

Note: 'X' represents the system can perform the corresponding function.

To compute the architectural complexity of the system, it is also necessary to estimate the two exponential weighting factors α and γ in Eq.(1). These can be estimated from experts ratings on the relative importance of the two complexity indicators to the architectural complexity of the system. To this end, a questionnaire is used to ask experts about the relative importance of the two complexity indicators. This questionnaire is illustrated in Table III.

TABLE III. QUESTIONNAIRE FOR EXPERTS ON THE RELATIVE IMPORTANCE OF THE NCD TERM AND THE CC TERM TO THE ARCHITECTURAL COMPLEXITY OF THE BRIDGE SYSTEM.

Relative Values ($\frac{NCD}{CC}$)	2	3/2	4/3	1	3/4	2/3	1/2
Ratings	I	II	III	IV	V	VI	VII

In total, seven different ratings are used in the questionnaire, ranging from "extremely important" (Rating I, relative importance value $\frac{NCD}{CC} = 2$) to "extremely unimportant" (Rating VII, relative importance value $\frac{NCD}{CC} = 1/2$). Every expert from a related field is asked to give an assessment on the relative importance of the NCD term and the CC term to the architectural complexity of the BRIDGE system. For illustration purposes, an example rating derived from answers by seven (fictitious) experts is listed in Table IV.

Based on the experts ratings, the estimates of the exponential weighting factors α and γ associated with the two complexity indicators can be obtained. The relative importance value of the NCD term with the CC term to the architectural complexity

	Column 1	**Column 2**	**Column 3**	**Column 4**	**Column 5**	**Column 6**	**Column 7**	**Column 8**	**Column 9**	**Number of Total Edges**	**Number of Total Nodes**
Row 1	1.1 ReMa	1.1 Mesh	1.1 RescueMe	1.1 CM	1.1 Beacons	1.1 Broadcaster					6
										30	
Row 2	1.2 Planner	1.2 RiMo	1.2 Mesh	1.2 RescueMe	1.2 CM						5
										15	
Row 3	1.3 Mesh	1.3 CM	1.3 MS								3
										3	
Row 4	1.4 Planner										1
										6	
Row 5	2.1 ReMa	2.1 RFID	2.1 Mesh	2.1 Repositories	2.1 CM	2.1 eTriageT					6
										48	
Row 6	2.2 ReMa	2.2 IA	2.2 RFID	2.2 Mesh	2.2 RiMa	2.2 PICG	2.2 CM	2.2 eTriageT			8
										8	
Row 7	2.3 Planner										1
										9	
Row 8	2.4 ReMa	2.4 IA	2.4 RFID	2.4 Mesh	2.4 RiMa	2.4 Repositories	2.4 PICG	2.4 CM	2.4 eTriageT		9
										9	
Row 9	2.5 Planner										1
										6	
Row 10	3.1 ReMa	3.1 IA	3.1 Mesh	3.1 Repositories	3.1 PICG	3.1 CM					6
										30	
Row 11	3.2 IA	3.2 RiMo	3.2 Mesh	3.2 RiMa	3.2 CM						5
										10	
Row 12	3.3 Planner	3.3 RiMa									2
										8	
Row 13	3.4 Planner	3.4 RiMa	3.4 DEIN	3.4 ENB							4
										32	
Row 14	3.5 ReMa	3.5 IA	3.5 RFID	3.5 Mesh	3.5 Repositories	3.5 PICG	3.5 CM	3.5 eTriageT			8
										8	
Row 15	3.6 Planner										1
										2	
Row 16	4.1 Planner	4.1 RiMo									2
										6	
Row 17	4.2 Mesh	4.2 CM	4.2 MS								3
										21	
Row 18	4.3 Planner	4.3 RiMo	4.3 RFID	4.3 RiMa	4.3 DEIN	4.3 ENB	4.3 eTriageT				7
										14	
Row 19	4.4 Planner	4.4 RescueMe									2
	Sum									**265**	**80**

Fig. 2. Program control graphs for the BRIDGE system architecture

TABLE IV. EXPERTS RATINGS ON THE TWO COMPLEXITY INDICATORS ACCORDING TO THEIR RELATIVE IMPORTANCE TO THE ARCHITECTURAL COMPLEXITY

Expert No.	1	2	3	4	5	6	7
Ratings	IV	III	V	II	IV	IV	III

is given below:

$$\left(\sum_{i=1}^{N_1} P_i^2 / N_1 \right)^{\alpha} = r \cdot (1 + \log Q)^{\gamma}, \tag{4}$$

where r is the rating value. Taking the logarithm on both sides, gives

$$\alpha \log \left(\sum_{i=1}^{N_1} P_i^2 / N_1 \right) - \gamma \log (1 + \log Q) = \log r. \tag{5}$$

Since the expert rating can only provide a pairwise relative evaluation of the importance of the complexity indicators, an exponential weighting factor first needs to be determined as a benchmark in order to determine the others. In this case, γ is set to 1. Then, α can quickly be estimated by averaging the experts rating results in the logarithmic domain. As an example, the estimates of α and γ for the BRIDGE system can be obtained by using the experts rating results in Table IV as below:

$$\alpha = 0.3757, \gamma = 1.0000. \tag{6}$$

With α and γ available, the architectural complexity of the BRIDGE system can be calculated as:

$$C = 30.4706^{0.3757} \times 3.27^{1.0000} = 11.8043. \tag{7}$$

IV. CONCLUSIONS

This paper introduces a new approach for the analysis and evaluation of architectural complexity of complex emergency rescue management systems. The BRIDGE system, a typical large-scale emergency rescue management system, is used to illustrate this approach. A new definition of architectural complexity is thereby introduced for this purpose. This definition of complexity uses additional exponential weighting factors to model the domain-related significance of several existing complexity indicators, which are widely used for assessing the architectural complexity of various types of complex systems and SoS. These exponential weighting factors can be readily estimated from a sequence of experts ratings, thereby enable a natural way to further incorporate the subjective experiences of experts and domain-related knowledge into the definition of architectural complexity. This new definition of architectural complexity provides a tool for the design of emergency rescue management systems in the conceptual and architectural designing phases.

For future work, our aim is to use this definition of architectural complexity to compare various alternatives of emergency rescue management scenarios, so that an optimal alternative proposal selection can be made that is tailored to the crisis at hand.

ACKNOWLEDGMENT

This work was done during the visit of Dr. Liang Gao to the Section Systems Engineering at the Faculty of Technology Policy and Management at the Delft University of Technology. This work is supported in part by a scholarship under the State Scholarship Fund (Grant No. 201405290008). The authors would like to thank the BRIDGE project (FP7/2011-2015 grant agreement no 261817), and in particular B. van Veelen for his work on the example scenario used.

REFERENCES

[1] C. Keating, R. Rogers, R. Unal, D. Dryer, A. Sousa-Poza, R. Safford, W. Peterson, and G. Rabadi, “System of systems engineering,” *Engineering Management Journal*, vol. 15, no. 3, pp. 36–45, 2003.

[2] C. B. Keating, “Research foundations for system of systems engineering,” in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, 2005, pp. 2720–2725.

[3] J. Boardman and B. Sauser, “System of systems-the meaning of of,” in *IEEE/SMC International Conference on System of Systems Engineering*, 2006, pp. 6–11.

[4] B. P. Zeigler and S. Mittal, “Enhancing dodaf with a devs-based system lifecycle development process,” in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, 2005, pp. 3244–3251.

[5] H. W. Ang, D. Nicholson, and B. Mercer, “Improving the practice of dod architecting with the architecture specification model,” DTIC Document, Tech. Rep., 2005.

[6] S. Mittal, “Extending dodaf to allow integrated devs-based modeling and simulation,” *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, vol. 3, no. 2, pp. 95–123, 2006.

[7] R. Youssef, B. Kim, J. Pagotto, A. Vallerand, S. Lam, P. Pace, C. Pogue, and A. Greenley, “Toward an integrated executable architecture and m&s based analysis for counter terrorism and homeland security,” DTIC Document, Tech. Rep., 2006.

[8] T. V. Huynh and J. S. Osmundson, “A systems engineering methodology for analyzing systems of systems using the systems modeling language (SysML),” 2006.

[9] S. Mittal, E. Mak, and J. J. Nutaro, “Devs-based dynamic model reconfiguration and simulation control in the enhanced dodaf design process,” *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, vol. 3, no. 4, pp. 239–267, 2006.

[10] H.-W. Ang, F. Dandashi, and M. McFarren, “Tailoring dodaf for service-oriented architectures,” in *IEEE Military Communications Conference*, 2006, pp. 1–8.

[11] S. J. Ring, D. Nicholson, J. Thilenius, and S. Harris, “An activity-based methodology for development and analysis of integrated dod architectures,” *Handbook of enterprise Systems Architecture in practice*, vol. 85, 2007.

[12] A. AbuSharekh, S. Kansal, A. K. Zaidi, and A. H. Levis, “Modeling time in dodaf compliant executable architectures,” in *Conference on Systems Engineering Research*, 2007.

[13] E. Baumgarten and S. J. Silverman, “Dynamic dodaf and executable architectures,” in *IEEE Military Communications Conference*, 2007.

[14] J. Charles Domerçant and D. N. Mavris, “Measuring the architectural complexity of military systems-of-systems,” in *IEEE Aerospace Conference*, 2011, pp. 1–16.

[15] K. Griendling and D. N. Mavris, “Development of a dodaf-based executable architecting approach to analyze system-of-systems alternatives,” in *IEEE Aerospace Conference*, 2011, pp. 1–15.

[16] X. Pan, B. Yin, and J. Hu, “Modeling and simulation for sos based on the dodaf framework,” in *the 9th International Conference on Reliability, Maintainability and Safety*, 2011, pp. 1283–1287.

[17] R. Wang and C. H. Dagli, “Executable system architecting using systems modeling language in conjunction with colored petri nets in a model-driven systems development process,” *Systems Engineering*, vol. 14, no. 4, pp. 383–409, 2011.

[18] B. Ge, K. W. Hipel, K. Yang, and Y. Chen, “A data-centric capability-focused approach for system-of-systems architecture modeling and analysis,” *Systems Engineering*, vol. 16, no. 3, pp. 363–377, 2013.

[19] B. V. Veelen, D. Ferro, A. Salden, C. Raffelsberger, M. Wenstad, M. Stiso, A. Eide, J. de Jong, G. Pavlin, N. Wijngaards, S. van Splunter, and M. Wietek, “D07.1 - Baseline emergency reference structures, Restricted Deliverable, BRIDGE project (FP7/2011-2015 grant agreement no 261817),” 2012.

[20] S. van Splunter, B. van Veelen, A. Stam, M. Buscher, and L. Wood, “D7.3 - Workflow configuration concepts and methods, Restricted Deliverable, BRIDGE project (FP7/2011-2015 grant agreement no 261817),” 2014.

[21] L. E. Zeidner, H. M. Reeve, R. Khire, and S. Becz, “Architectural enumeration and evaluation for identification of low-complexity systems,” in *13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2010.

[22] M. J. Kinnunen, “Complexity measures for system architectures,” Master’s thesis, Massachussetts Institute of Technology, 2006.

[23] C. W. Churchman and P. Ratoosh, Eds., *Measurement: Definitions and theories*. New York: Wiley, 1960.