Implementing a MOSA Decision Support Tool in a Model-Based Environment

Michael Dai, Cesare Guariniello, and Daniel DeLaurentis

Abstract The Modular Open Systems Approach (MOSA) is a DoD initiative that requires major defense acquisition programs to employ modular architectures using widely accepted standards. In order to realize the benefits of modular and open architectures, program stakeholders must successfully navigate various technical and programmatic decisions throughout the acquisition life cycle. Our observation is that many programs do not have sufficient methods and tools to perform analysis, assess trades, and produce evidence for decisions that produce good program outcomes in general and in specific respect to modularity. This paper presents a model-based approach to rigorously collect and present acquisition context data and data from analysis tools in a Decision Support Framework (DSF). Through an example multi-domain mission engineering problem, we demonstrate how the DSF enables comparison of modular/non-modular mission architectures in terms of cost and performance. In addition, an MBSE enterprise architecture model is used to implement the DSF and is shown to (1) provide detailed visualizations of alternative architecture solutions for better comparison; (2) allow traceability between features of the architecture and organizational requirements to better document adherence to MOSA principles; and (3) lay the groundwork for continued model-based engineering development downstream of the Analysis of Alternatives activity to the rest of the acquisition life cycle.

Keywords System of systems · Model-based systems engineering · Mission engineering

M. Dai (✉) · C. Guariniello · D. DeLaurentis
Purdue University, West Lafayette, IN, USA

A. M. Madni et al. (eds.), *Recent Trends and Advances in Model Based Systems Engineering*, https://doi.org/10.1007/978-3-030-82083-1_22

1 Introduction

1.1 The Modular Open Systems Approach (MOSA)

MOSA is a strategy to implement "highly cohesive, loosely coupled, and severable modules" into system designs that can be "competed separately and acquired from independent vendors" (DDR&E(AC) n.d.). This is to be achieved using widely supported and consensus-based (i.e., "open") standards, as they are available and suitable (ODASD 2017). The Office of the Under Secretary of Defense for Research and Engineering states that the approach intends to realize the following benefits (DDR&E(AC) n.d.):

1. *Enhance competition* – open architectures with severable modules allow components to be openly competed.
2. *Facilitate technology refresh* – new capabilities or replacement technology can be delivered without changing all components in the entire system.
3. *Incorporate innovation* – operational flexibility to configure and reconfigure available assets to meet rapidly changing operational requirements.
4. *Enable cost savings/cost avoidance* – reuse of technology, modules, and/or components from any supplier across the acquisition life cycle.
5. *Improve interoperability* – severable software and hardware modules to be changed independently.

MOSA compliance has become a mandate by law for all major defense acquisition programs (10 USC §2446a). However, effective tools for DoD programs to conduct analysis and produce evidence of MOSA implementation are still lacking. This, combined with inconsistent understanding on how to balance MOSA with other trade variables, has produced a situation where many programs struggle to make effective choices and document the rationale for these decisions.

1.2 Barriers to Achieve MOSA Benefits

While MOSA promises great benefits, challenges remain to successfully realize them. Through workshops and interviews, we have interacted with expert practitioners from industry, military, and DoD to better define these challenges (see DeLaurentis et al. 2017, 2018). Practitioners identified among the challenges the need to understand how modular and open architecture alternatives modify technical trades on system life cycle cost, development schedule, performance, and flexibility toward changing mission requirements. There are also many programmatic difficulties associated with the adoption of MOSA. For example, data rights and intellectual property often incentivize vendors from sharing detailed design information that may impact other modules. Selection of working groups, compartmentalization of information, and other organizational structure features can also have a strong impact on how modular systems are successfully realized.

Thus, guidance is needed to navigate the technical trade space regarding modular architectures and the organizational requirements that would best enable modular system designs. A Decision Support Framework (DSF) is being developed to address these challenges.

2 A Decision Support Framework to Guide MOSA Implementation

Figure 1 shows the basic concept of the Decision Support Framework. The idea is to create an executable software that can provide key information to program managers and other stakeholders to guide MOSA-related decisions throughout the acquisition life cycle.

2.1 *DSF Inputs – Mission Engineering and Early-Stage Acquisition Contexts*

The inputs to the software are the parameters of the mission engineering problem that would surround an acquisition: a mission Concept of Operations, a description of capability gaps to be fulfilled, and a library of candidate systems to be selected

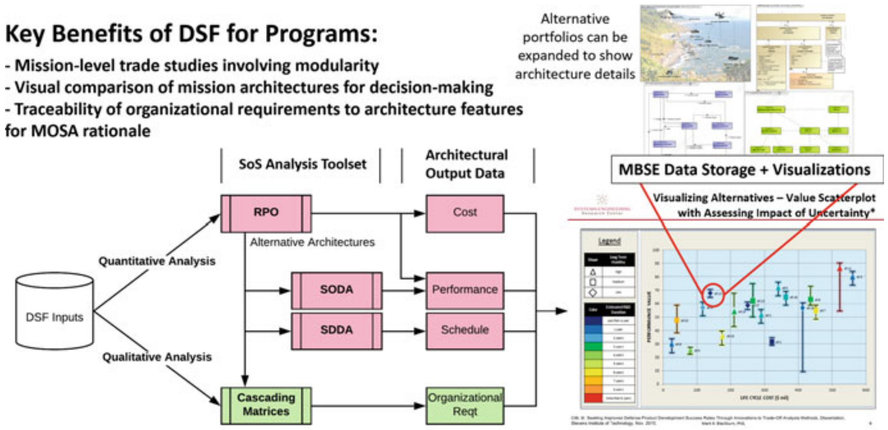

Key Benefits of DSF for Programs:
- Mission-level trade studies involving modularity
- Visual comparison of mission architectures for decision-making
- Traceability of organizational requirements to architecture features for MOSA rationale
Alternative portfolios can be expanded to show architecture details
SoS Analysis Toolset
Architectural Output Data
MBSE Data Storage + Visualizations
Visualizing Alternatives – Value Scatterplot with Assessing Impact of Uncertainty*
Legend
PERFORMANCE VALUE
LIFE CYCLE COST ($ mil)
RPO
Alternative Architectures
SODA
SDDA
Cascading Matrices
Cost
Performance
Schedule
Organizational Reqt
DSF Inputs
Quantitative Analysis
Qualitative Analysis

Fig. 1 Workflow of the Decision Support Framework concept. The decision-making scatterplot is a figure taken from SERC research efforts led by Blackburn, described in Bone et al. (2018). RPO, SODA, and SDDA are tools from the Purdue Analytic Workbench described in DeLaurentis et al. (2016)

and integrated to achieve mission objectives. This library represents the capabilities of current and to-be-acquired systems, each involving varying degrees of modularity. The alternative systems in the library will allow us to explore how the selection of modular and open systems will impact the mission architecture as a whole.

The user will also input the mission-level requirements that all solution SoS architectures must satisfy. These requirements will be based on mission capability metrics relevant to the mission category. For example, an amphibious assault mission may require SoS capabilities like naval superiority, air superiority, tactical bombardment, and land seizure. Rigorously defining metrics for this level of capability is challenging and is outside the scope of this paper.

2.2 DSF Analysis – Quantitative and Qualitative Analysis Threads

The DSF analysis is divided into two threads: a quantitative analysis addressing the technical trade-offs associated with modularity and a qualitative analysis addressing programmatic considerations for MOSA.

The quantitative analysis could eventually use a set of tools that are most familiar and trusted by a particular program, as long as they are configurable to represent choices related to modularity and openness. Our prototype DSF employs tools from Purdue's SoS Analytic Work Bench, described in DeLaurentis et al. 2016. Robust Portfolio Optimization (RPO), detailed in Davendralingam and DeLaurentis 2013, generates alternative architectures and analysis of cost and performance. In RPO, hierarchies of systems are modeled as nodes on a network that work cohesively to fulfill overarching capability objectives. Capabilities (outputs) from existing nodes connect to fulfill requirements (inputs) of other nodes, amidst compatibility constraints. The end goal is to generate a set of "portfolios" from a library of constituent systems (or components) that are pareto-optimal with respect to SoS-level performance goals and constraints, under measures of uncertainty. In its application to the DSF, the portfolios represent feasible mission architectures in terms of their constituent systems, including both modular and monolithic assets. Systems Operational Dependency Analysis (SODA) and Systems Developmental Dependency Analysis (SDDA), developed by Guariniello and DeLaurentis (2013, 2017), are AWB tools that provide additional quantitative assessment of the architectures in terms of operational and schedule risks.

Qualitative considerations on MOSA architectures are also analyzed. In many cases, our prior research has found that programs first need a way to explore and understand the various aspects of modularity, their interplay with key program cost-schedule-performance outcomes, and long-term sustainment considerations (DeLaurentis et al. 2017, 2018). This context, along with the initial sparsity of data in early life cycle phases, makes the qualitative thread important in the benefits of the DSF. The qualitative portion uses quality function deployment (QFD) techniques

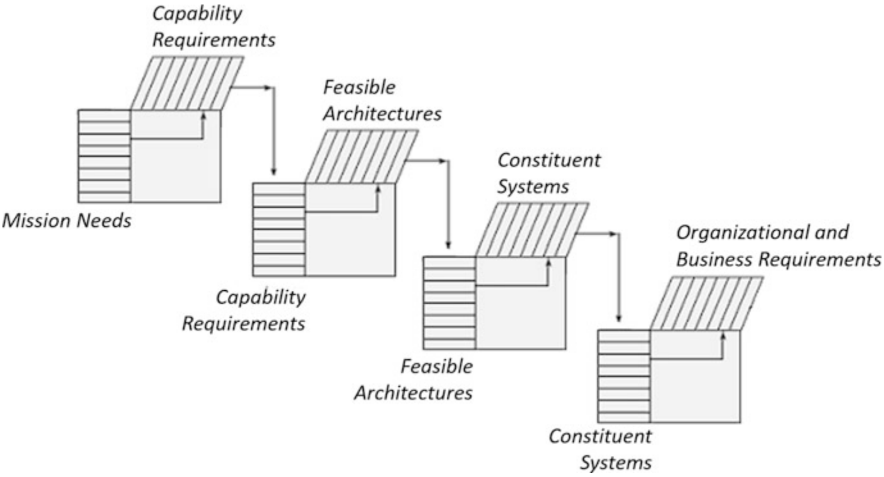


Fig. 2 Cascading matrices in the DSF are used to trace mission needs through alternative architectures to organizational and business requirements. Waterfall representation is adapted from a figure by the American Society for Quality in Revelle (2004)

and cascading matrices to trace features of alternative architectures to the ideal organizational requirements associated with them (Fig. 2). The idea is to use a series of matrices to map mission-level capability needs to certain mission-level requirements. These requirements are then mapped onto the alternative architectures (identified by RPO) which satisfy them. Finally, alternative architectures are mapped to the organizational and MOSA-related resources needed to achieve them.

2.3 DSF Outputs – Integrated Decision-Making Views in a Model-Based Environment

Finally, projected outputs of the DSF software will display the implications on cost, schedule, and risk of modular architectures and relationships between features of system solutions and the organizational structures that would best support them. We are presently assembling a comprehensive synthetic problem to exercise and demonstrate both the qualitative and quantitative tracks of the DSF. A simplified multi-domain battle scenario problem is used in this work to demonstrate the use of MBSE to support the application of RPO in the DSF.

We apply the concept of an MBSE system model with visualizations to the implementation of the DSF. Using a model-based environment allows DSF inputs and outputs to be collected and linked together in a central database. This will enable an integrated means to visualize pertinent data and facilitate DSF decision-making. Our application of MBSE to this problem domain is inspired by SERC research efforts led by Blackburn and leverages principles and best practices in model-centric engineering from his work. Bone et al. (2018) and Blackburn (2019) demonstrate how integrating data from various engineering analysis tools in a model-based workflow can be achieved and presented in a single decision-making window. His decision-making window is shown as part of in Fig. 1 as the envisioned means of displaying the DSF outputs.

3 Implementation Results Using an Example Mission Engineering Problem

In this section, the idea of implementing the Decision Support Framework using MBSE concepts is expounded upon and illustrated through a simple example problem. In it, a multi-domain mission is to be performed using five concept roles: a surveillance system, a communications system, an air superiority system, a power supply system, and a maritime superiority system. They interact as a network to achieve generic SoS capabilities.

3.1 MBSE Views Establishing Mission Context for the DSF

The general premise of the DSF is to allow the analysis and comparison of modular/non-modular architecture solutions in a given mission context. Thus, before the analysis is performed, the mission engineering problem must be clearly stated. The enterprise architecture model is therefore instantiated with high-level operational concept information expressed in OV-1 and OV-2 DoDAF models (Fig. 3). The OV-1 shows the general concept roles that will later be filled by alternative system solutions, in addition to notional dependency relationships. The OV-2 specifies the intended flow of information, energy, and material entities between the general system categories (Fig. 3b). Creating these views is helpful for validating to-be architectures against the original mission needs and assessing their ability to adapt to changes in mission configuration.

3.2 Identification of Feasible Architectures with RPO

To identify feasible architectures, a library of component systems is collected along with their individual performance, requirements, compatibility constraints, and associated uncertainties, shown in Appendix A. Among the candidate component systems are modular and non-modular options. The tool is used to generate the set

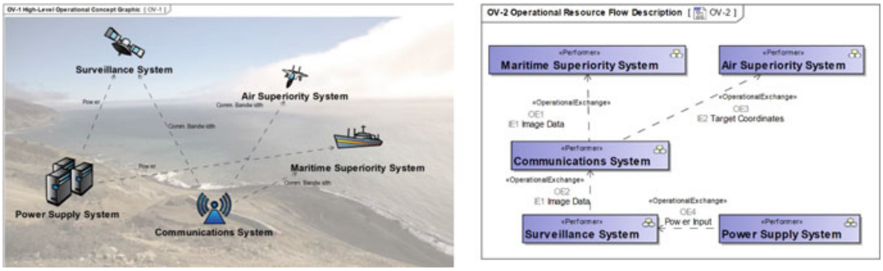

OV-1 High-Level Operational Concept Graphic [OV-1]
Surveillance System
Air Superiority System
Maritime Superiority System
Power Supply System
Communications System
Power
Comm. Bandwidth
OV-2 Operational Resource Flow Description [OV-2]
«Performer» Maritime Superiority System
«Performer» Air Superiority System
«Performer» Communications System
«Performer» Surveillance System
«Performer» Power Supply System
«OperationalExchange» OE1 E1 Image Data
«OperationalExchange» OE3 E2 Target Coordinates
«OperationalExchange» OE2 E1 Image Data
«OperationalExchange» OE4 Power Input

Fig. 3 (**a**) OV-1 high-level operational concept view of the mission and capability roles to be fulfilled. (**b**) OV-2 operational resource flow description detailing how each performer concept will exchange information, energy, and material flows (see DoD CIO 2010)

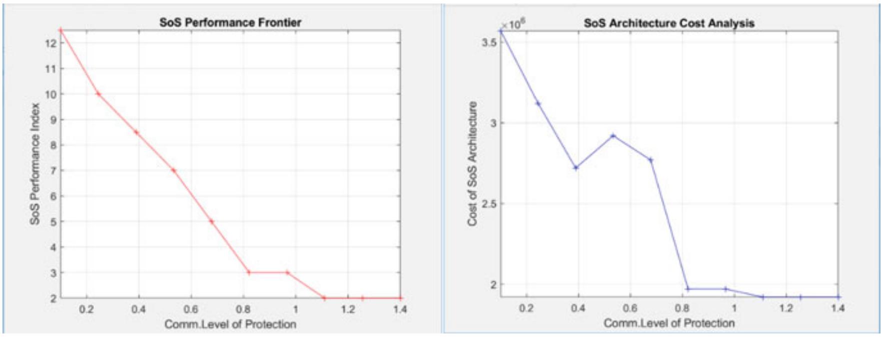


Fig. 4 Pareto frontier of SoS portfolios identified by RPO in terms of SoS performance, cost, and risk aversion toward constraint violations on the communications requirements

of architectures that are optimal with respect to SoS-level capabilities, cost, and risk protection against constraint violations.

Running the optimizer results in the pareto-optimal portfolios shown in Fig. 4. Here the SoS capabilities have been consolidated into a single “SoS Performance Index” metric. Each portfolio has a different performance index, cost, and level of protection with respect to communications constraints. This level of protection is essentially an inverted measure of how likely node-level communication bandwidth requirements are to be violated due to uncertainties in system communication capabilities.

3.3 MBSE Representations of Output Data

After feasible architectures have been identified, they are filtered through three QFD cascade matrices. The first matrix maps user-selected mission needs to mission performance requirements. The second matrix connects these SoS-level requirements

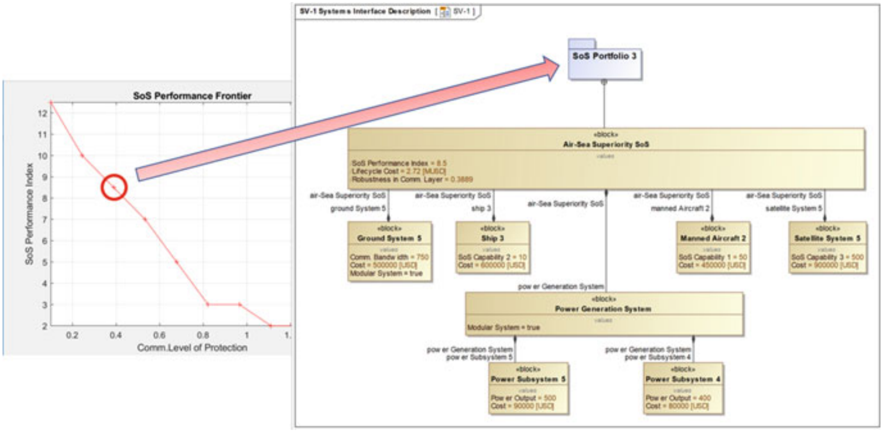


Fig. 5 The selected SoS portfolio is expanded to show its composition details in a SysML block diagram. This level of granularity reveals that the architecture incorporates a modular power generation system

to the RPO feasible architectures that can fulfill them. The third cascade maps the features of the alternative architectures to organizational requirements needed to effectively realize them. The result of the QFD cascades is a set of feasible alternative SoS architectures. Imposing a single mission requirement that the SoS Performance Index ≥ 5, the results from RPO identify four architectures on the Pareto frontier as the final set of alternatives.

The result of the analysis portion of the DSF is a listing of alternatives that can be compared in terms of architecture cost, acquisition timeframe, and performance (but here, only cost and performance). The Pareto frontier plot allows these architectures to be visually compared at a high level. More detailed architecture information can be obtained by linking each pareto-optimal point to a MBSE representation of that alternative. In Fig. 5, a block definition diagram is used to model a selected portfolio, containing information on the systems comprising it and how each collectively contributes to the SoS-performance objectives. In this example, the model and visualization were made manually for the selected portfolio. While steps toward automated diagram generation and linkage are described in the closing section, this is reserved for future work.

In addition to showing portfolio composition, MBSE can also represent the results of the QFD cascades by showing the traceability from system alternatives to organizational requirements using a SysML requirements diagram. For example, additional organizational requirements may be specifically tied to modular features of a system. Representing this traceability enables programs to provide evidence of MOSA principles in their design decisions. This is illustrated in Fig. 6. Specifying these organizational requirements will be based on MOSA case study data – however, for now, they remain notional.

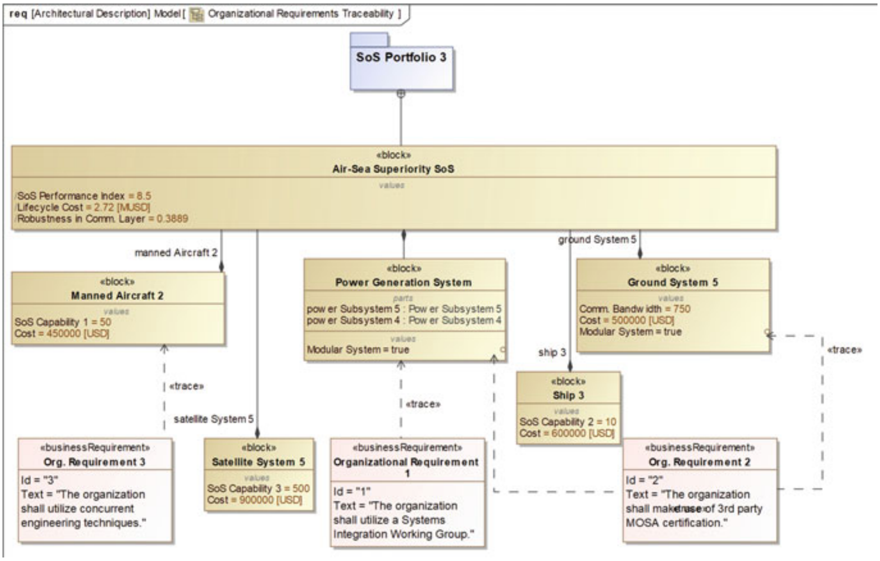


Fig. 6 Requirements diagram showing the traceability of system features to organizational requirements

4 Summary

4.1 Key Takeaways

This paper examined how the Decision Support Framework can be used as a tool toward better achieving the benefits of modularity, as motivated by the MOSA initiative. The DSF addresses key challenges concerning MOSA implementation, primarily those related to evaluating technical trades involving modularity, and tracing modular architectural features to organizational requirements needed to enable them. The simplified example problem demonstrated the use of Robust Portfolio Optimization in the DSF to enable comparison between architectures with modular/non-modular system alternatives, in terms of cost and performance. In addition, the example portrays how the DSF utilizes an MBSE enterprise architecture model to store and visualize mission context, architecture structural features, and requirement traceability.

MBSE adds value to the DSF in three ways:

1. Linked visualizations can provide more detailed architecture data upon user inquiry. This allows decision-makers to quickly jump between levels of granularity when comparing alternative architectures.
2. MBSE system models allow clear traceability between mission requirements, selection of modular/non-modular alternatives, and MOSA-relevant organiza-

tional requirements. This digital thread records the rationale behind programmatic decisions and can provide evidence for use of MOSA principles.

3. Creating a high-level architecture model capturing results from the Analysis of Alternatives (AoA) activity paves the way for continued model-based engineering throughout the acquisition life cycle. The MBSE model developed from the DSF can come to establish an authoritative source of truth, allowing more detailed system models to directly build off the mission and enterprise level models.

4.2 Future Work

There are many directions for future work on this project. One is to consider additional metrics to compare mission architectures. Other analysis tools in Purdue's AWB will be added to the DSF to assess flexibility and schedule metrics. With more dimensions of comparison, having an integrated decision-making window as a DSF output, such as that shown in Fig. 1, will become even more important.

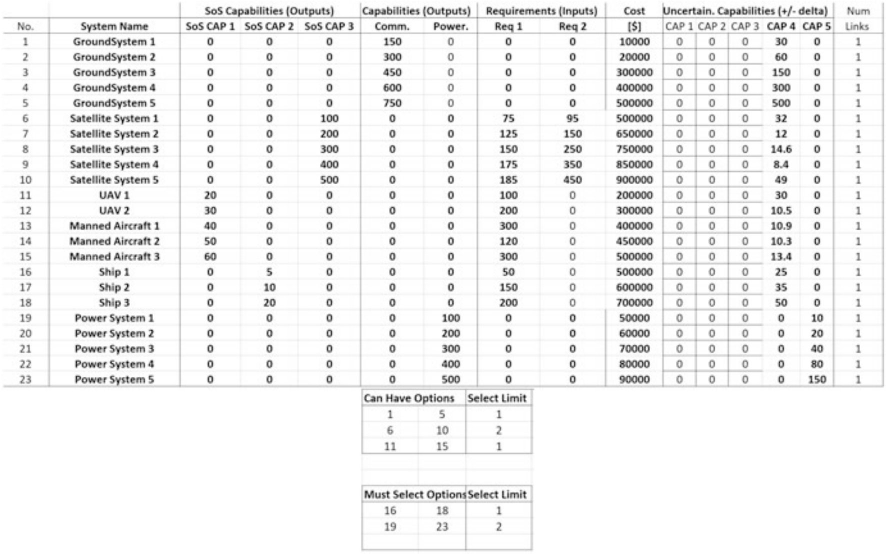
A second area of ongoing work is to understand what organizational requirements are necessary for different kinds of modular mission architectures. This work is being performed through case study analysis of successful MOSA programs and through collaboration with ongoing partner programs.

A third area of future work is in creating a digital linkage directly from RPO Pareto fronts (or other decision windows) to the MBSE portfolio visualizations. Practically, this would entail being able to click on a certain portfolio in the decision-making windows in Figs. 1 or 5 and having its visualizations directly be generated.

Acknowledgments This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contract HQ0034-19-D-0003 WRT-1002. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology. We further acknowledge the contributions of research collaborators Gary Witus, Charles Domercant, and Thomas McDermott.

A.1 Appendix A: RPO Input Data for Example Mission Engineering Problem

In this example, candidate systems are labeled generically (e.g., Satellite System 1–5) and use notional data. The data in Fig. A.1 can be read as follows: Satellite System 1 contributes 100 to "SoS Capability 3" and requires 75 [units] in communication bandwidth and 95 [units] in power input. Likewise, Power System 3 offers no SoS or communication capabilities, but is capable to supply 300 [units] of power to other systems. Each capability is subject to an uncertainty that may result in violating node input requirements. This information is reflected in a risk aversion metric shown



		SoS Capabilities (Outputs)			Capabilities (Outputs)		Requirements (Inputs)		Cost	Uncertain. Capabilities (+/- delta)					Num
No.	System Name	SoS CAP 1	SoS CAP 2	SoS CAP 3	Comm.	Power.	Req 1	Req 2	[$]	CAP 1	CAP 2	CAP 3	CAP 4	CAP 5	Links
1	GroundSystem 1	0	0	0	150	0	0	0	10000	0	0	0	30	0	1
2	GroundSystem 2	0	0	0	300	0	0	0	20000	0	0	0	60	0	1
3	GroundSystem 3	0	0	0	450	0	0	0	300000	0	0	0	150	0	1
4	GroundSystem 4	0	0	0	600	0	0	0	400000	0	0	0	300	0	1
5	GroundSystem 5	0	0	0	750	0	0	0	500000	0	0	0	500	0	1
6	Satellite System 1	0	0	100	0	0	75	95	500000	0	0	0	32	0	1
7	Satellite System 2	0	0	200	0	0	125	150	650000	0	0	0	12	0	1
8	Satellite System 3	0	0	300	0	0	150	250	750000	0	0	0	14.6	0	1
9	Satellite System 4	0	0	400	0	0	175	350	850000	0	0	0	8.4	0	1
10	Satellite System 5	0	0	500	0	0	185	450	900000	0	0	0	49	0	1
11	UAV 1	20	0	0	0	0	100	0	200000	0	0	0	30	0	1
12	UAV 2	30	0	0	0	0	200	0	300000	0	0	0	10.5	0	1
13	Manned Aircraft 1	40	0	0	0	0	300	0	400000	0	0	0	10.9	0	1
14	Manned Aircraft 2	50	0	0	0	0	120	0	450000	0	0	0	10.3	0	1
15	Manned Aircraft 3	60	0	0	0	0	300	0	500000	0	0	0	13.4	0	1
16	Ship 1	0	5	0	0	0	50	0	500000	0	0	0	25	0	1
17	Ship 2	0	10	0	0	0	150	0	600000	0	0	0	35	0	1
18	Ship 3	0	20	0	0	0	200	0	700000	0	0	0	50	0	1
19	Power System 1	0	0	0	0	100	0	0	50000	0	0	0	0	10	1
20	Power System 2	0	0	0	0	200	0	0	60000	0	0	0	0	20	1
21	Power System 3	0	0	0	0	300	0	0	70000	0	0	0	0	40	1
22	Power System 4	0	0	0	0	400	0	0	80000	0	0	0	0	80	1
23	Power System 5	0	0	0	0	500	0	0	90000	0	0	0	0	150	1

Can Have Options		Select Limit
1	5	1
6	10	2
11	15	1

Must Select Options		Select Limit
16	18	1
19	23	2

Fig. A.1 Example of RPO input data for the example problem

on the horizontal axis of Fig. 4. Finally, compatibility and selection constraints are set in the input spreadsheet as well. Here, the optimizer can select one option from systems 1–5 (Ground Systems) and 11–15 (Aerial Systems) and up to two options from systems 6–10 (Satellite Surveillance Systems). Likewise, the optimizer is constrained to select one Naval System (16–18) and two Power Systems.

References

Blackburn, M. 2019. *Transforming Systems Engineering through Model Centric Engineering*. Systems Engineering Research Center Workshop.

Bone, M., M. Blackburn, B. Kruse, J. Dzielski, T. Hagedorn, and I. Grosse. 2018. Toward an Interoperability and Integration Framework to Enable Digital Thread. *Systems* 6 (4): 46.

Davendralingam, N., and D. DeLaurentis. 2013. A Robust Optimization Framework to Architecting System of Systems. *Procedia Computer Science* 16: 255–264.

DeLaurentis, D., N. Davendralingam, K. Marais, C. Guariniello, Z. Fang, and P. Uday. 2016. An SoS Analytical Workbench Approach to Architectural Analysis and Evolution. *OR Insight* 19 (3): 70–74.

DeLaurentis, D., N. Davendralingam, M. Kerman, and L. Xiao. 2017. *Investigating Approaches to Achieve Modularity Benefits in the Acquisition Ecosystem*. (Report No. SERC-2017-TR-109). Retrieved from the Systems Engineering Research Center.

DeLaurentis, D., N. Davendraligam, C. Guariniello, J.C. Domercant, A. Dukes, and J. Feldstein. 2018. *Approaches to Achieve Benefits of Modularity in Defense Acquisition*. (Report No. SERC-2018-TR-113). Retrieved from the Systems Engineering Research Center.

Directorate of Defense Research and Engineering for Advanced Capabilties (DDR&E(AC)). (n.d.). *Modular Open Systems Approach*. Retrieved from Office of the Under Secretary of Defense for Research and Engineering: https://ac.cto.mil/mosa/

DoD Deputy Chief Information Officer. 2010. *The DoDAF Architecture Framework Version 2.02*. Retrieved from Chief Information Officer- US Department of Defense: https://dodcio.defense.gov/library/dod-architecture-framework/

Guariniello, C., and D. DeLaurentis. 2013. Dependency Analysis of System-of-Systems Operational and Development Networks. *Procedia Computer Science* 16: 265–274.

———. 2017. Supporting Design via the System Operational Dependency Analysis Methodology. *Research in Engineering Design* 28 (1): 53–69. https://doi.org/10.1007/s00163-016-0229-0.

Office of the Deputy Assistant Secretary of Defense. 2017. MOSA Definition. FY17 National Defense Authorization Act (NDDA), 10 USC §2446a.

Revelle, J.B. 2004. *Quality Essentials: A Reference Guide from A to Z*. ASQ Quality Press. Retrieved from American Society for Quality: https://asq.org/quality-resources/qfd-quality-function-deployment.