PUR-SE-22-004

ACQUISITION RESEARCH PROGRAM
SPONSORED REPORT SERIES

A System-of-Systems Approach to Enterprise Analytics Design: Acquisition Support in the Age of Machine Learning and Artificial Intelligence

November 1, 2021

Daniel A. DeLaurentis, Associate Professor

Cesare Guariniello, Research Scientist

Prajwal Balasubramani, Graduate Student

Purdue University

Disclaimer: This material is based upon work supported by the Naval Postgraduate School Acquisition Research Program under Grant No. HQ0034-19-1-0007. The views expressed in written materials or publications, and/or made by speakers, moderators, and presenters, do not necessarily reflect the official policies of the Naval Postgraduate School nor does mention of trade names, commercial practices, or organizations imply endorsement by the U.S. Government.

Prepared for the Naval Postgraduate School, Monterey, CA 93943.

The research presented in this report was supported by the Acquisition Research Program of the Graduate School of Defense Management at the Naval Postgraduate School.

To request defense acquisition research, to become a research sponsor, or to print additional copies of reports, please contact the Acquisition Research Program (ARP) via email, arp@nps.edu or at 831-656-3793.

Abstract

System-of-Systems (SoS) capability emerges from the collaboration of multiple systems, which are acquired from independent organizations. Even though the systems contribute to and benefit from the larger SoS, the data analytics and decision-making about the independent system is rarely shared across the SoS stakeholders. The objective of this work is to identify how the sharing of datasets and the corresponding analytics among SoS stakeholders can lead to an improved SoS capability. Our objective is to characterize how the sharing of connected data sets may lead to deployment of different predictive (predicting an outcome from data) and prescriptive (determining a preferred strategy) analytics and lead to better decision outcomes at the SoS level. We build and demonstrate a framework for this objective based on extensive literature review and generating appropriate predictive and prescriptive methodologies that can be used for SoS analysis: Additionally, we propose to utilize machine learning techniques to predict the SoS capability achievable by sharing pertinent datasets and to prescribe the information links between systems to enable this sharing. Two case studies demonstrate the use of the framework and prospects for meeting the objective. Highlights of our study are summarized next.

Literature Survey

Our literature survey examined three domains as they relate to our study objective. First, we surveyed research in SoS engineering and methodology to be able to properly define the problem and the activities for this research. At the same time, we looked at the current directions of research in SoS-oriented acquisition at the DoD. The second direction of literature survey identified predictive and prescriptive analytics methodologies. Predictive analytics are useful in the context of SoS acquisition, characterized by high levels of uncertainty due to complexity, interdependencies between constituent systems, and presence of multiple independent stakeholders. Prescriptive analytics have been surveyed as a potential source of decision-making support methodologies. Finally, we surveyed data analytics and Machine Learning methodologies, identifying key features and requirements of various methodologies most

promising for SoS acquisition. Due to the diverse nature of SoS problems, we cannot identify a single mathematical formulation, but rather indicate the best approach for different types of problems.

Demonstrative Example and Results

Our first case study demonstration uses our previously developed Decision Support Framework (DSF), in particular the systems portfolio optimization tool, deployed on a multi-domain battle scenario where the different stakeholders do not agree on the relative weight of the different achievable SoS capabilities. Outputs of the model include optimal SoS architectures based on different budget limits, risk aversion, and on the different perception of the SoS objectives by multiple stakeholders. Users can identify systems and subsystems that are present in various optimal architectures, as well as artifacts produced by the different perceived SoS requirements. In the second demonstration example, we model a problem in Urban Air Mobility, where a stakeholder entering the market faces uncertainty in data, both because of partial information on potential customers as well as on market decision by competitors. In this case, we use predictive and prescriptive analytics to provide support towards the best decision-making, after properly modeling the interactions due to the dynamic nature of this SoS problem.

About the Authors

Daniel A. DeLaurentis is a Professor in the School of Aeronautics and Astronautics, Purdue University where he has been on the faculty since 2004. His research and teaching interests focus on optimization and design methods for aerospace systems, system of systems, design under uncertainty and morphing aircraft.

Cesare Guariniello is a Research Scientist in the School of Aeronautics and Astronautics, Purdue University, where he received his PhD in 2016. He currently works in the Center for Integrated Systems in Aerospace (CISA) led by Dr. DeLaurentis, doing research in the fields of human space exploration, aerospace systems, System-of-Systems Engineering.

THIS PAGE LEFT INTENTIONALLY BLANK

PUR-SE-22-004

ACQUISITION RESEARCH PROGRAM SPONSORED REPORT SERIES

A System-of-Systems Approach to Enterprise Analytics Design: Acquisition Support in the Age of Machine Learning and Artificial Intelligence

November 1, 2021

Daniel A. DeLaurentis, Associate Professor

Cesare Guariniello, Research Scientist

Prajwal Balasubramani, Graduate Student

Purdue University

Disclaimer: This material is based upon work supported by the Naval Postgraduate School Acquisition Research Program under Grant No. HQ0034-19-1-0007. The views expressed in written materials or publications, and/or made by speakers, moderators, and presenters, do not necessarily reflect the official policies of the Naval Postgraduate School nor does mention of trade names, commercial practices, or organizations imply endorsement by the U.S. Government.

THIS PAGE LEFT INTENTIONALLY BLANK

Table of Contents

THIS PAGE LEFT INTENTIONALLY BLANK

Background and Scope of Research

System-of-Systems (SoS) capability emerges from the collaboration of multiple systems, which are acquired from independent organizations. The systems within an SoS serve two purposes: one is to meet their own independent objectives, and the second is to contribute some capability to the SoS from which all constituents can benefit. In recent decades, the fields of machine learning and data analytics have found widespread application in system design and acquisitions. It is unanimously understood that any organization acquiring a complex system employs some form of data analytics to assess a system's independent objectives. Even though the systems contribute to and benefit from the larger SoS, the data analytics and decision-making about the independent system is rarely shared across the SoS stakeholders. The objective of this work is to identify how the sharing of datasets and the corresponding analytics among SoS stakeholders can lead to an improved SoS capability.

Characteristics of SoS (Maier, 1998) make them quite different from simple systems and the resulting behavior of a SoS is often unpredictable just by knowing its constituent parts, due to the interactions between those parts. Given the interdependencies in SoS, when considering acquisition, it is important to recognize the stakeholders, resources, operations, policies, and economics not only of one system, but of the entire SoS. Uncertainty and possible hidden information are common in SoS acquisition, and since the SoS capability is a multi-faceted enterprise, it is hard to formulate a single set of mathematical equations that would cover all cases. Therefore, in this work we develop research towards an information-centric framework that helps inform early-stage decisions on enterprise level.

Important context for our work comes from the ambitious goals put forth in both defense and commercial sectors for Digital Engineering (DE) and its related components in various engineering functions, such as Model-Based Systems Engineering (MBSE) for the SE domain. DE and MBSE pursue the use of digital models at every phase of acquisition. However, much of the focus right now is on the "how" of DE/MBSE and the desire to have models interoperate rather than the degree to which the extended enterprise (some say the “acquisition ecosystem”) has awareness of and belief in the various datasets that underly the models and the development processes that use them. Within this context, our goal is to identify data management and analytic deployment strategies that create synergies between different enterprise entities and link the stakeholders, resources, policy, and economics between different systems. The framework focuses on examining the impact that data features (e.g., survey categories, types of variables, ownership/privacy of data etc.) have on the type and effectiveness of predictive and prescriptive analytics that can be employed and how the outcome can be shaped differently by examining the connectivity of data sets. This is particularly important for SoS acquisition where these data set exist at local system level but may not be shared at the SoS/enterprise level or vice versa. Our objective is to characterize how the sharing and the connectivity of data sets may lead to deployment of different predictive and prescriptive analytics (due to data access) and lead to better outcome at the SoS level. We do so with a mapping of useful techniques from the data science / machine learning literature as well as two illustrative examples.

After extensive literature review, we identified appropriate predictive and prescriptive methodologies that can be used for SoS analysis: predictive analytics are used to evaluate future data based on current available information; prescriptive analytics provide strategies in support of acquisition decision-making. Finally, we propose to utilize machine learning techniques to predict the SoS capability achievable by sharing pertinent datasets and to prescribe the information links between systems to enable this sharing.

Literature Review

Overview of Data Analytics

Predictive data analytics provides an ability to anticipate and predict outcomes by collecting and utilizing prior information (Waller & Fawcett, 2013) (Rehman et al., 2016) (Joseph & Johnson, 2013). Although using data to guide decision-making has been around since the Babylonian times, where data was recorded on tablets to predict harvest (Lo & Hasanhodzic, 2011), a major shift in the ability to reason over large amount of data emerged in 1940s with the advent of computer development, storage, and machine learning techniques. For application in complex systems, early usage of analytics can be traced back to the 1940s and 50s when data analytics models were used to predict outcomes for the behavior of nuclear chain reactions in the Manhattan Project and the weather forecasting using the ENIAC computer (Lynch, 2008).

Prescriptive data analytics, on the other hand, aims to provide an ability to generate/prescribe the best courses of action based on given information which may be obtained from a predictive data analytic outcome. Starting around World War II, the need to optimize courses of actions stimulated the development of operations research field (INFORMS, n.d.) which in the proceeding decades led to ‘Analytics 1.0’ for introducing data-based decision making in organizations. As the capabilities of computing and machine learning evolved to handle structured and unstructured large data sets (also known as Big Data), Analytics 2.0 became the new paradigm across most large enterprises such as Google and Amazon (Davenport, 2013). Today, the Big Data landscape is shaped by the volume, variety, velocity, and veracity of data (known as the big four Vs of data science), and an organization’s ability to include this ‘Analytics 3.0’ in the decision-making process has become fundamental to its success and profitability. It will not be a generalization to state that most successful organizations employ some form of Analytics 3.0 for business and product development.

For SoS acquisition and capability development, deployment of Analytics 3.0 provides a unique challenge where the individual organizations contributing the constituent systems individually employ a suite of predictive and prescriptive analytics tools (section 2.2 provides details on predictive machine learning techniques as applied primarily in the DoD application space). However, these analytics and the underlying data sets are rarely shared across the SoS stakeholders. Given that the SoS capability emerges from the collaboration of otherwise independent systems and considering the ever-increasing need of interoperability between systems for transitioning towards DE and MBSE, there is an imperative to connect the data sets across SoS for holistic Analytic 3.0 capability deployment. In previous work (summarized in section 2.3), we have established the significance of connecting data sets across an enterprise and our objectives with this work-in-progress is to develop this capability for SoS acquisition.

Machine Learning Techniques and Application in Defense Acquisition

To get a sense of how predictive analytics and machine learning models are used in the literature, we examine the most popular algorithms and their application. The main goal of statistical evaluation of data is to explain relationships between variables and use them to make predictive and prescriptive recommendations. Relationships between the response variable (output/target) and the independent variables (inputs/features/predictors) can be modeled using both supervised and unsupervised learning techniques. Supervised learning algorithms use predictors and a target variable to learn a function that maps the predictors to the target. It consists of regression and classification models depending on if the response variable is quantitative or categorical respectively. Unsupervised learning algorithms model the underlying structure of a data set with a set of predictors and no response variable (“Supervised Learning vs Unsupervised Learning,” 2018).

The simplest of the supervised learning models is linear regression. This type of algorithm is used to examine the linear relationship between one or more categorical and/or quantitative predictors and a continuous response variable. Linear regression uses an optimization method called “Ordinary Least Squares” (OLS), which minimizes the sum of squared error between the observed and predicted values to estimate the model parameters. (Moore & White III, 2005) combine a multivariate linear and logistic regression model to identify the root causes of procurement cost growth in engineering and manufacturing development in the DoD procurement process. They use a binary variable representing if a program will have cost growth in procurement dollars and a continuous variable of the percentage of procurement cost growth for the logistic regression and multivariate linear regression model respectively. Moore and White III's two step-process involves the prediction of the amount of cost growth a program will have using the multivariate regression model results in which those programs are identified based on how likely a program will have procurement cost growth using the results from the logistic regression in the initial step.

Linear regression is not without its disadvantages: OLS model becomes more complex when more variables are added to the model, introducing multicollinearity and overfitting. Modifications of the linear regression model, ridge and lasso, are used to address this. The parameter estimates are obtained similarly to the linear regression model with the difference being the addition of a penalization term to the loss function. Ridge regression adds the sum of squared magnitude of the coefficients (L2 norm) while lasso adds the sum of absolute value of magnitude (L1 norm). Both models include a tuning parameter, λ, in the penalization term to control the amount of shrinkage to the coefficients. The larger (smaller) the tuning parameter, the model runs the risk of under (over) fitting. If $\lambda=0$, the loss function is equivalent to OLS used in linear regression. Ridge regression is best used when the multivariate linear regression model suffers from multicollinearity while lasso is best used as a variable reduction or feature selection technique as it shrinks unnecessary coefficients to zero. To address the multicollinearity issue in defense spending, (Huang & Mintz, 1990), use ridge regression to model the relationship between military expenditures and economic growth. (Wang & Yang, 2016) used lasso regression as a variable reduction technique to select variables most relevant to supply and demand of airline tickets.

Binary logistic regression is a classification algorithm that models the relationship between a dichotomous response variable, usually denoted as "success" or "failure" and a set of categorical and/or quantitative predictors. This model commonly uses a logit link function where the purpose is to transform the linear combination of the predictor variables, which can take on any value from the real line, and convert the values between zero and one, transforming them on a probabilistic scale (MacKenzie et al., 2017). The logit link function is defined as modeling the log odds of the "success" of the outcome variable as a linear combination of the input variables. In a univariate logistic regression model, the odds increase multiplicatively by the exponential of the coefficient per every unit increase in the predictor variable. Apte et al. (2016) explore how the DoD can use information on contractor performance to identify variables that drive the success in service acquisition by using logistic regression and other big data techniques. Success or failure of a contract was used as the response variable and found that type of contract, awarded dollar value, workload (actions) by filled billets, and percentage of 1102 billets filled by contracting office, had the largest impact on a contract's success. An additional workload of 10 actions per billet is more likely to have a failed contract by 13%, and cost plus award fee (CPAF) and cost plus fixed fee (CPFF) contracts are more likely to fail than firm fixed price (FPP) contracts (Apte et al., 2016).

Support Vector Machines (SVM) is a classification technique that plots each data point in an N-dimensional space where the goal is to identify a linearly separable hyperplane that maximizes the distance (margin) between the data points of a dichotomous response variable. This algorithm uses only the set of data points, called support vectors, closest to the margin to classify the data. If the data is linearly inseparable, a kernel function is used to map the non-linear data into a high dimensional space to become linearly separable (Berwick, n.d.). (Wei et al., 2019) use SVMs to estimate the state of charge of lithium-ion batteries for unmanned aerial vehicles (UAVs).

Artificial Neural Networks (ANN) is one of the most powerful machine learning algorithms. A neural network consists of a set of inputs (input layer), interconnected nodes, and an output layer. Data from each node in the input layer is passed to a node in the hidden layer (interconnected nodes) that calculates a weighted sum (Hardetsy, 2017). The hidden layer uses an activation function (e.g., sigmoid function) to determine if the weighted sum of the inputs is passed to the next hidden layer based on if the weighted sum is greater than a threshold/bias until the data reaches the output layer. ANNs are best used when relationships are not constricted to linearity or normality assumptions, when relationships between the variables are difficult to model using traditional approaches, and to discover patterns in the data (Burger, n.d.). (Brotherton & Johnson, 2001) use a neural network to detect anomalies or unexpected faulty conditions in engine operation of advanced military aircraft.

K-Nearest Neighbors (KNN) is used to classify a data point based on the known class of its neighbors. To classify an observation in the test set, the distance between the observation and all the data points in the training set must be calculated using a distance metric (e.g., Euclidean distance) to identify the k-nearest points. Classification of a data point is assigned to one of the categories that appears the most among an observation's k-closest points if the response is categorical. If the response is quantitative, KNN becomes a regression problem, and the assigned output value for an observation is calculated using the arithmetic mean of its k-nearest points. (Xiao et al., 2006) use KNN and SVMs to classify types of military vehicles based on the acoustic and seismic signals generated.

K-means is a clustering algorithm that is used to identify K homogenous clusters in the data such that the points in each cluster are similar to each other. The algorithm estimates initial values of centroids (the average of the data points in a cluster) as a first step, and then iteratively assigns each data point to the closest centroid based on a distance metric and takes the mean of all the data points in the cluster to calculate a new centroid. The iteration of the algorithm stops when cluster centroids are stabilized. K-means ensures that data points are homogenous within and heterogeneous between clusters. The final result is the assignment of each data point to a single cluster. (Zainol et al., 2018) use K-means to uncover text patterns in military peacekeeping documents.

A summary of the above ML techniques and their application to the DoD related problem is provided in Table 1.

Table 1. Summary of ML Methodologies.

Method	Key Features	Assumptions	DoD Reference
Supervised Learning			
Linear Regression	Fits quantitative/categorical predictors and continuous response to regression line using OLS	Linear parameters, constant error variance, independent error terms, errors are normally distributed, random sample of observations, no multi-collinearity	Moore and White III (2005)
Ridge Regression	Modification of linear regression that uses L2 norm when multi-collinearity assumption in linear regression is broken	Standardization of predictors, linear parameters, constant error variance, independent errors (*Regression Analysis Software | NCSS Software*, n.d.)	Huang and Mintz (1990)
Lasso Regression	Used as a variable reduction or feature selection technique that shrinks some predictor coefficients to exactly zero to reduce overfitting from the linear regression model	Model has sparsity, irrepresentable conditions (Zhao & Yu, 2006)	Wang and Yang (2016)
Binary Logistic Regression	Models the log odds (using logit link) of a categorical binary outcome variable as a linear combination of quantitative/categorical predictors	Independent observations and errors, binomial distribution of response variable, linearity between logit of response and predictors (*Summary Points for Logistic Regression* , n.d.)	Apte et al. (2016)
Support Vector Machine	Uses a linearly separable hyperplane to classify data into two classes	Independent and identically distributed observations, margin is as large as possible, support vectors are most useful data points	Wei et al. (2006)
Artificial Neural Networks	Model consisting of interconnected nodes that receive inputs and return outputs based on an activation function	Independence of inputs	Brotherton and Johnson (2001)
K-Nearest Neighbors	Used to classify data points based on class that appears the most among neighboring points (classification) or average of classes (regression)	Similar inputs have similar outputs (*Weinberger*,2018)	Xiao et al. (2006)
Naive Bayes Classifier	Uses Bayes theorem to calculate probabilities of a class response and selects the class with highest probability as the output	Predictors are conditionally independent of each other given the response	Freeman (2013)
Decision Tree	Algorithm that recursively and iteratively partitions the data into homogeneous subsets to identify a target outcome	Entire training set is at root node, quantitative predictors must be discretized	Apte et al. (2016)
Unsupervised Learning			
K-means	Use to identify homogeneous clusters in a data set	Clusters sizes are similar and spherical in form	Zainol et al. (2017)

Decision tree analysis is a supervised learning technique that can be used for both regression and classification to visually display decisions as a tree-like diagram represented as homogeneous partitions of the data that lead to a target outcome. The structure has a root node, internal nodes, which represent a split on a predictor variable, and leaf nodes, which represents a target outcome. Every decision tree can be represented using binary decisions at each internal node (Gales, 2013). Decision trees are most commonly built using a top-down approach, which is an iterative and recursive process that selects the best predictor variable for splitting the data into disjoint subgroups based on a splitting criterion (e.g. Information gain, Gini gain) applied to each descendant node (Hand et al., 2001). Apte et al. (2016) also used a decision tree to analyze success or failure of a contract. From top-down, the decision tree first split on awarded dollar value of contract, workload (actions) by filled billets, then finally percentage of filled billets.

The objectives of this part of the literature review are two-fold: first is to identify the various ML methods which can be applied to SoS acquisition problems and map their input, output, and data requirements, and second is to assess how these methods are applied for different DoD problems. An emerging thread in the above literature review is isolated application of these methods where the outcomes along with the data sets are rarely shared across the different hierarchical levels of an organization and SoS. Moore and White III (2005) use the combination of multiple algorithms on a single data set at a local system to identify programs with increase cost growth. The objectives for our research differ in that we aim to analyze how multiple systems run their own individual predictive analytics at the local level and how to best share different data sets across SoS hierarchy to prescribe the SoS capability.

Integrating Predictive and Prescriptive Analytics for Acquisition

In previous work, we have used a conceptual problem to demonstrate the impact that even small, intuitive changes in how data is collected and shared, can result in different predictive and prescriptive analytic implementations and lead to a different outcome for SoS decision making (Davendralingam et al., 2018). Let us take a simple and conceptual example of an enterprise where the objective is to maximize profit by selling a product for which multiple independent entities such as the dealer/distributors of the product, the corporate headquarters, and market research organization must work together. Each of these entities have their own independent objectives for which they use data analytics for decision making. Consider a scenario where the Market Research Team performs a market study to understand the consumer's opinion on the product design. Now, this information can be used to support the *future* product design at the Corporate HQ by providing insights on what aspects of the design are the most crucial for the consumer. With the better understanding of the information flow, the same set of collected data can also provide insights to the Dealer/Distributor on what features of the *current* product design dictate the demand and thus, lead to higher profits. In this simplified example, this link of information flow might seem trivial but when looking at real-world system-of-systems, identifying this important link remains a challenge. In this research, we are pursuing development of a framework that will facilitate identification of such links and quantify how the SoS level capability could evolve by sharing data sets across the systems.

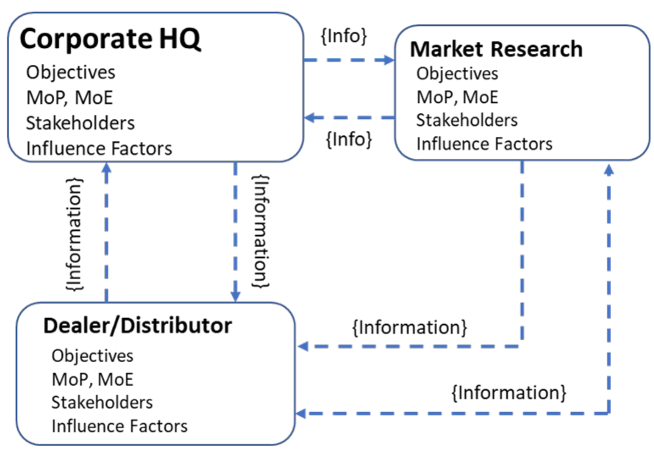

Figure 1 Conceptual problem to identify impact of data-set connectivity

Methodology

We build our approach from a conceptual model of SoS that provides a lexicon and taxonomy for representing the various SoS constructs and utilize it to examine data needs and their respective connectivity (Davendralingam et al., 2018). The framework is envisioned to assist in orchestration of analytics and data architecture components across an organization for improved enterprise level performance. The framework is comprised of three phases: Definition, Abstraction, and Implementation. The purpose of the Definition phase is to holistically identify the stakeholders, resources, policy, and economics at different hierarchal levels within an SoS. The Abstraction phase, then, develops representation of the artifacts identified in the Definition phase and recognizes the networks, and hence, interconnection of stakeholders, resources, policy, and economics. This is where the opportunity lies to identify what new connections between the artifacts could be established. Finally, in the Implementation phase, the solution to the SoS problem as defined and abstracted is investigated. Here, the focus is on identifying the right solutions methods which are tailored to the SoS problem.

Consider for example, a system-of-system capability as illustrated in Figure 2. The Definition phase identifies elements comprising the system-of-system at different hierarchy levels while the abstraction phase identifies the links between these elements. In this case, sub-systems α_1 and α_2 form the system β_1 whereas α_3 relates to the system, β_2. At the higher level, β_1 and β_2 form the system-of-system, γ_1. Now, each of the sub-system suppliers, system manager, and SoS managers have independent goals of employing data analytics to improve their figures of merit. At the sub-system level, the supplier 1 and supplier 2 may not foresee a need for data set connectivity between α_1 and α_2. However, the potential need for such connectivity becomes evident only at the $\beta_1$system. Since supplier 1, supplier 2 and sys 1 manager all become part of the same system, identifying the right information pathways and connecting data sets for predictive and prescriptive analytics becomes necessary. Similarly, the same logical formulation can be applied to the SoS-level which may demand connectivity of data sets between system β_1 and system, β_2, and subsequently implying connectivity between supplier 1, supplier

2, and supplier 3. However, it may not be pragmatic to achieve a full connectivity between all constituent systems and elements of the SoS. Therefore, identifying which datasets need to be connected by characterizing how their connectivity impacts the SoS becomes a pertinent question.

The complexity and scale of this problem for any real-world implementation prohibits an analytical solution. In this research, we address two types of problems: first, we formulate the SoS capability measure based on acquiring multiple systems within the DoD application domain and demonstrate how the SoS capability evolves due to sharing preferences between sub-hierarchical systems while maintaining the independent system objectives. Second, we model a scenario where information about the future or the choice of other stakeholders might be unavailable. In future work in this research, we aim to investigate deployment of machine learning techniques to predict and prescribe the connectivity of data sets across the different hierarchical levels.

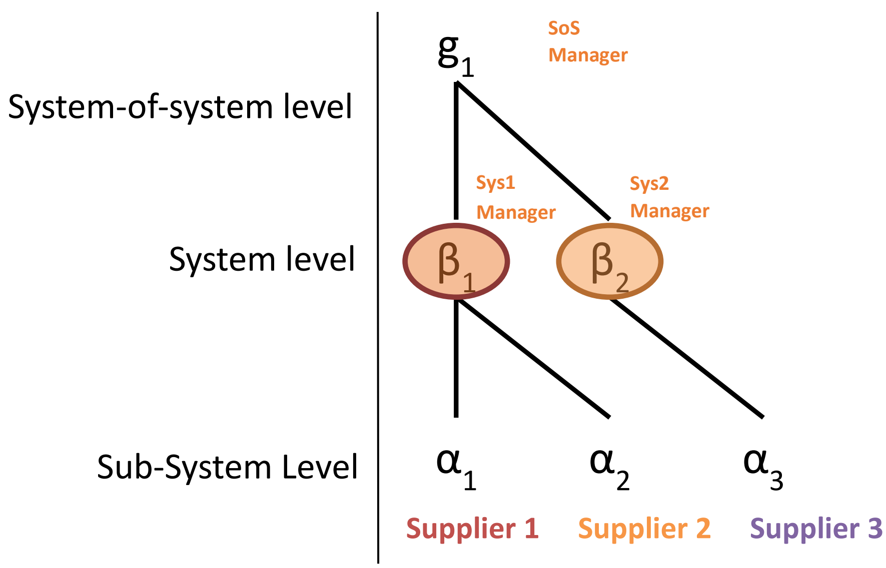

Figure 2 SoS Conceptual Example

Optimal Acquisition with uncertainty on Objectives

The Decision Support Framework (DSF) is a tool that includes various SoS analytical tools. The primary function of the DSF is to perform quantitative Analysis of Alternatives (AoA) by generating portfolios of systems that provide both the SoS capabilities of interest and the necessary logistical support for the systems included in the portfolio. This capability is accomplished by integrating a Robust Portfolio Optimization (RPO) (Davendralingam & DeLaurentis, 2015) analysis tool for SoS which evaluates not only system- and SoS-level capabilities but also the constraints imposed by interactions between systems (i.e., via support capability requirements). The DSF also performs quantitative and qualitative analysis of each architecture by generating analysis of disruptions via Systems Operational Dependency Analysis (SODA) (Guariniello & DeLaurentis, 2017), network representation of the systems and their dependencies through biographs, and cascading matrices that show how systems contribute to SoS capabilities.

For this case study, we are focusing on acquisition problems where stakeholders do not have the same perception of the relative importance of different SoS objectives. A synthetic problem was created to run simulations using the DSF software and interface with other existing System-of-Systems (SoS) analytical tools. The synthetic problem is an Amphibious Warfare Scenario, which was chosen since it is a multi-domain problem involving air, ground, naval, and space systems. The systems in Amphibious Warfare interact to provide logistical support and system-level capabilities to achieve certain SoS-level capabilities. A case study was developed for the synthetic problem that specifically defined systems from a World War II Amphibious Warfare Scenario. Use of World War II systems and context was chosen since many measures of system capabilities from this time period are public knowledge, which allowed the research team to create a case study with adequate detail. The Mission System Library (MSL) is the key means to pass user inputs into the DSF. The MSL is created in an Excel workbook, where a series of five sheets provide specific information on the problem:

1. Main Sheet: System names, support capabilities (i.e., internal logistic requirement), system capabilities, and capability uncertainties
2. SoS Capabilities: SoS capability names and sets of indices of the system capabilities that contribute to each SoS capability
3. Compatibility Constraints: matrices containing information on compatibility between systems, specification of maximum number of specific systems allowed in a portfolio
4. Must Have Systems: to indicate any mandatory systems in a portfolio
5. Conditional Must Have Systems: establishing system interdependencies for operations

Each of these sheets can be read automatically by the DSF software to run the SoS analysis tools and create the outputs. The user is expected to create their own Mission System Library for their specific problem. An example problem was created for the Amphibious Warfare Scenario to evaluate the portfolios generated under different scenarios and user-input parameters.

In this paper we are not using all the tools included in the DSF, but instead focus on those that provide results suitable for the objectives of this research on SoS and ML tools. The tool that we used to generate different portfolios based on various scenarios and input parameters is the Robust Portfolio Optimization (RPO). This is a methodology to maximize the expected performance of SoS and keep within acceptable levels of developmental risk and cost, while at the same time dealing with uncertain information. Implementation of RPO for a certain SoS design problem yields a set of Pareto optimal portfolios of cost vs. SoS performance, corresponding to a user-defined risk aversion factor. The optimization is based on a mixed integer programming technique and all the interdependency between component systems are depicted as constraints.

In the DSF, RPO has been improved with an additional layer that includes not only support capabilities and systems capabilities, but also multiple SoS capabilities that can be included in a weighted function for multi-objective optimization. Initial quantitative architecture analysis of alternatives is performed in the DSF using the RPO method. RPO generates optimized portfolios of systems, and it creates Pareto graphs to display results for SoS-level performance vs. portfolio cost. Other tools can be added for further quantitative evaluation.

The DSF runs the RPO tool using as input the system information from the MSL. The user can modify the parameters of the analysis in the DSF Main GUI. Based on the scenario loaded from the opening screen, the GUI will display the user’s list of SoS capabilities that can be selected for optimization, as well as a list of support capabilities, from which the user can select whether uncertainty needs to be considered or not. These options implement concepts of Mission-Based design, where even the same set of available systems will generate different portfolios based on different mission requirements. On the right side of the GUI, the user can define levels of risk aversion and levels of available budget, which are used later for generating Pareto frontiers. Other inputs include the importance weights for the selected SoS capabilities and the option to set the requirement to include modular systems.

Concept Application: Multi-Domain Scenario

Considering a realistic setting, where multiple officers/designers/managers are involved, in a multi-objective SoS acquisition problem, a common occurrence is differences in interpretation of the mission requirements either due to lack of communication or judgement. This study investigates how such dissimilarities in the definition of the mission requirements of one contributing individual from another affects the final SoS performance and cost.

In the previously discussed Amphibious Warfare case study, multiple systems were defined in each domain, including air, ground, naval, and space, as well as human systems (e.g., operators). In the MSL, 26 systems were defined, though only an excerpt is provided in Figure 3 and Figure 4, and then evaluated for their support and system capabilities. Five support capabilities were defined for this case study: Transport Range (measured by range in miles), Transport Capacity (measured by capacity in pounds), Refuel (measured by fuel capacity in pounds), Communication Relay (measured using a constructed rating), and Operator (measured by number of operators). Each system might have one or more support input requirements, which must be fulfilled by a system that has a matching support output capability. Therefore, two sets of columns were defined in the MSL for support capabilities: Support Input Requirement and Support Output Capability. Some systems might be only "support systems" if they only provide support output but do not provide system capabilities. Though the quantified SoS capabilities are evaluated using only the system capabilities, the Robust Portfolio Optimization tool is able to consider the support inputs and outputs by creating constraints that must be satisfied for any portfolio, making these interdependencies still critical to the architecture results.

No.	System Type	System Name	Transport Range	Transport Capacity	Refuel	Communication Relay	Operator
			Support Input Requirement				
-	-	-	Range (mi)	Capacity (lb)	Fuel capacity (lb)	Rating (n.d.)	Number of Operators
1	Air Systems	P-51 Mustang	0	2000	2795	0	1
2		B-17 Flying Fortress	0	6000	18500	0	10
3		C-47	0	0	5369	0	4
4		B-52H Stratofortress	0	60000	321000	1	5
5		B-2 Spirit	0	40000	167000	1	2
6	Ground Systems	Infantry Platoon	10	1845	0	0	42
7		M114 155mm Howitzer	12480	12480	0	0	4
8		M-4 Sherman	150	1251	869	0	5
9		M8 Greyhound	175	274	353	0	4
10		Jeep Willis	0	0	95	0	1
11		"Deuce and a half" (supply truck)	0	0	378	0	1
12		Advanced Targeting Pod	0	0	0	0	0
13		TARDEC Chassis	0	0	378	0	1
14		TARDEC Anti Air Module	100	879	0	0	4
15		TARDEC Artillery Module	100	1750	0	0	4
16		TARDEC Personal Module	100	0	0	0	0
17		Bofors 40 mm gun (L60)	100	4800	0	0	4
18		Refuel Depot	0	0	0	0	0
19		Resupply Depot	0	0	0	0	0
20	Naval Systems	Allen M. Sumner Destroyer	0	0	0	0	336
21		Higgins Boat (LCVP)	0	0	0	0	3
22		Landing Ship, Tank (LST)	0	0	0	0	140
23		Battleship	0	0	0	0	2,220
24	Space Systems	Ultrahigh Frequency Follow-on (UFO) Communication Satellite	0	0	0	0	100
25		Wideband Global Satellite Communication Satellite (WGS)	0	0	0	0	100
26	Human	General Personnel	0		0	0	0

Figure 3. List of available systems and support requirements.

No.	System Type	System Name	SC15 = Defend Ground - Against Sea	SC16 = Defend Sea Against Air	SC17 = Defend Sea Against Ground	SC18 = Defend Sea Against Sea	SC19 = Mobility Air	SC20 = Mobility Ground	SC21 = Mobility Sea	SC22 = Surveillance	SC23 = Communication	Cost	Modular System?	Transport Range	Transpo Capaci
-	-	-	Weapons Range (mi), Stopping power (n.d.), Robustness (n.d.)	Weapons Range (mi), Stopping power (n.d.), Robustness (n.d.)	Weapons Range (mi), Stopping power (n.d.), Robustness (n.d.)	Weapons Range (mi), Stopping power (n.d.), Robustness (n.d.)	Combat Radius (mi), Operational Speed (mph)	Combat Radius (mi), Operational Speed (mph)	Combat Radius (nmi), Operational Speed (knots)	Detection rating (n.d.)	Communications Rating (n.d.)	($USD 2019)	Y/N	Uncertainty (+/- delta)	Uncerta ty (+/- delta]
1	Air Systems	P-51 Mustang	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	1650, 360	0, 0	0, 0	2	1	$582,000.00	N	0	0
2		B-17 Flying Fortress	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	400, 150	0, 0	0, 0	1	1	$3,399,600.00	N	0	0
3		C-47	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	3800, 160	0, 0	0, 0	1	1	$2,173,800.00	N	0	200
4		B-52H Stratofortress	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	4400, 525	0, 0	0, 0	2	2	$78,900,000.00	N	0	0
5		B-2 Spirit	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	3450, 560	0, 0	0, 0	2	2	$3,423,000.00	N	0	0
6	Ground Systems	Infantry Platoon	1, 1, 1	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	10, 3	0, 0	1	0	$8,876.91	N	0	0
7		M114 155mm Howitzer	9, 4, 1	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	1	1	$182,581.95	N	0	0
8		M-4 Sherman	2, 3, 3	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	150, 30	0, 0	1	1	$701,849.94	N	0	0
9		M8 Greyhound	1, 2, 2	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	175, 55	0, 0	1	1	$359,147.56	N	0	0
10		Jeep Willis	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	150, 65	0, 0	1	1	$1,575.21	N	0	70
11		"Deuce and a half" (supply truck)	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	150, 45	0, 0	1	0	$19,879.13	N	0	600
12		Advanced Targeting Pod	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	2	2	$1,000.00	Y	0	0
13		TARDEC Chassis	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	100, 45	0, 0	0	0	$142,404.01	Y	0	200
14		TARDEC Anti Air Module	2, 2, 2	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	1	1	$25,000.00	Y	0	0
15		TARDEC Artillery Module	5, 2, 3	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	1	1	$25,000.00	Y	0	0
16		TARDEC Personal Module	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	0	0	$20,000.00	Y	0	100
17		Bofors 40 mm gun (L60)	3, 2, 1	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	0	0	$100,000.00	Y	0	0
18		Refuel Depot	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	0	0	$0.00	N	0	0
19		Resupply Depot	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	0	0	$0.00	N	0	100
20	Naval Systems	Allen M. Sumner Destroyer	0, 0, 0	4, 3, 3	4, 3, 3	4, 4, 4	0, 0	0, 0	3300, 20	2	1	$152,000,000.00	N	0	0
21		Higgins Boat (LCVP)	0, 0, 0	1, 1, 2	1, 1, 2	1, 1, 2	0, 0	0, 0	10, 9	0	1	$229,300.00	N	0	200
22		Landing Ship, Tank (LST)	0, 0, 0	4, 3, 3	4, 3, 3	4, 3, 3	0, 0	0, 0	10000, 12	2	1	$36,320,100.00	N	0	1000
23		Battleship	0, 0, 0	9, 3, 3	9, 3, 4	9, 4, 3	0, 0	0, 0	5900, 21	2	1	$100,238,000.00	N	0	0
24	Space Systems	Ultrahigh Frequency Follow-on (UFO) Communication Satellite	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	2	2	$382,000,000.00	N	0	0
25		Wideband Global Satellite Communication Satellite (WGS)	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	2	2	$300,000,000.00	N	0	0
26	Human	General Personnel	0, 0, 0	0, 0, 0	0, 0, 0	0, 0, 0	0, 0	0, 0	0, 0	0	0	$170.00	N	0	0

Figure 4. List of the same systems in Figure 3. Light red columns are the provided system capabilities; green column is cost; blue column indicates modularity; grey columns are uncertainties.

Next, all desired SoS Capabilities and the indices of the system capabilities that contribute to the SoS capability are defined. Each SoS capability is computed using a normalized sum of individual system capabilities in their respective domain. For this experiment, the focus will be on three SoS capabilities related to an amphibious warfare scenario – Air Superiority, Naval Superiority and Reconnaissance (shown in Table 2). DSF is also equipped to handle several more SoS capabilities, in an effort to extend its usability to a larger spectrum of acquisition problems going forward to leverage the use of machine learning techniques.

Table 2. SoS Capabilities for Amphibious Warfare case study, with system capability contributions.

No.	SoS-Capability	System-Capability Indices					
1	Air Superiority	1-6	19-27	46	47	52	53
2	Naval Superiority	13-18	37-45	50-53			
3	Reconnaissance	46-53					

A main feature of the DSF, and the one that is used to investigate our problem statement, is the ability to assign weights to the SoS capability based on the preference for the mission requirement. For example, an acquisition manager who believes that the final portfolio of systems needs to be oriented more towards Air Superiority capability over the others would assign weights accordingly (ex: Air Superiority = 0.8, Naval Superiority = 0.1, and Reconnaissance = 0.1). This then leads to the issue of conflicting objectives among the team of acquisition managers or SoS designers. In order to learn the impact of different acquisition strategies (characterized by manager expectations concerning the relative importance of SoS capabilities) on final portfolios, we run 30 cases of varying weights among the team of acquisition managers to understand the variance in portfolios, performance and cost of the SoS. Figure 5 shows the weight distribution for each of the SoS capabilities.

Weights			
Cases	Air Superiority	Naval Superiority	Reconnaissance
1	0.8	0.1	0.1
2	0.7	0.2	0.1
3	0.7	0.1	0.2
4	0.6	0.2	0.2
5	0.6	0.3	0.1
6	0.6	0.1	0.3
7	0.5	0.1	0.4
8	0.5	0.2	0.3
9	0.5	0.3	0.2
10	0.5	0.4	0.1
11	0.4	0.5	0.1
12	0.4	0.4	0.2
13	0.4	0.3	0.3
14	0.4	0.2	0.4
15	0.4	0.1	0.5
16	0.3	0.6	0.1
17	0.3	0.5	0.2
18	0.3	0.4	0.3
19	0.3	0.3	0.4
20	0.3	0.2	0.5
21	0.3	0.1	0.6
22	0.2	0.7	0.1
23	0.2	0.6	0.2
24	0.2	0.5	0.3
25	0.2	0.4	0.4
26	0.2	0.3	0.5
27	0.2	0.2	0.6
28	0.2	0.1	0.7
29	0.1	0.1	0.8
30	0.1	0.8	0.1

Figure 5. Test runs with variation in weight distribution

Running the DSF and using RPO, we collected the resulting SoS portfolios for all these cases. In each case, we obtain multiple portfolios that are feasible within given budget limits. From our runs, a total of 4 feasible instances for each case was produced and these were used to form a pareto frontier to better understand the relation between SoS capability preferences, performance, and cost

Multi-Domain Scenario – Results

For each case, the DSF produces portfolios containing data about the various possible architectures and its associated SoS performance index and cost. A portfolio is a feasible combination of systems, which includes some that provide the required capabilities and others that provide the needed support. Figure 6 is one example (case 1) for a portfolio generated for a test case. Each column is a Pareto-optimal portfolio for a given budget limit. Zeros mean that the corresponding architecture does not utilize the system in question. Ones indicate systems that are part of the architecture. Looking into this data will give the user insight on how these suggested architectures differ amongst themselves and how they compare with other cases.

SoS capability	0	0.0624	0.0691	0.0737	0.0737
Cost	$0.00	$ 145.24M	$ 297.24M	$ 597.24M	$ 679.24M
P-51 Mustang	0	1	1	1	1
B-17 Flying Fortress	0	1	1	1	1
C-47	0	0	0	0	0
B-52H Stratofortress	0	0	0	0	0
B-2 Spirit	0	0	0	0	0
Infantry Platoon	0	1	1	1	1
M114 155mm Howitzer	0	0	0	0	0
M-4 Sherman	0	1	1	1	1
M8 Greyhound	0	1	1	1	1
Jeep Willis	0	1	1	1	1
"Deuce and a half" (supply truck)	0	0	0	0	0
Advanced Targeting Pod	0	1	1	1	1
TARDEC Chassis	0	0	0	0	0
TARDEC Anti Air Module	0	0	0	0	0
TARDEC Artillery Module	0	0	0	0	0
TARDEC Personal Module	0	0	0	0	0
Bofors 40 mm gun (L60)	0	0	0	0	0
Refuel Depot	0	1	1	1	1
Resupply Depot	0	0	0	0	0
Allen M. Sumner Destroyer	0	0	1	1	1
Higgins Boat (LCVP)	0	1	1	1	1
Landing Ship, Tank (LST)	0	1	1	1	1
Battleship	0	1	1	1	1
Ultrahigh Frequency Follow-on (UFO) Communication Satellite	0	0	0	0	1
Wideband Global Satellite Communication Satellite (WGS)	0	0	0	1	0
General Personnel	0	1	1	1	1

Figure 6. Architectures in a single scenario

In this example, it is observable that when the architectures switch to a combination that includes one or more different systems, better performing yet expensive, the SoS capability improves. This possibility of various permutations of system architectures makes a portfolio-based study more relevant and accurate for SoS acquisition problems.

The data from the portfolios generated in the 30 scenarios are then used to identify the space of solutions for all the cases individually. To do so, Portfolio Performance Frontiers where the SoS Performance Index is mapped with its corresponding costs is plotted. Figure 7 is a representation of this for two cases (Case1 and Case 17 as an example).

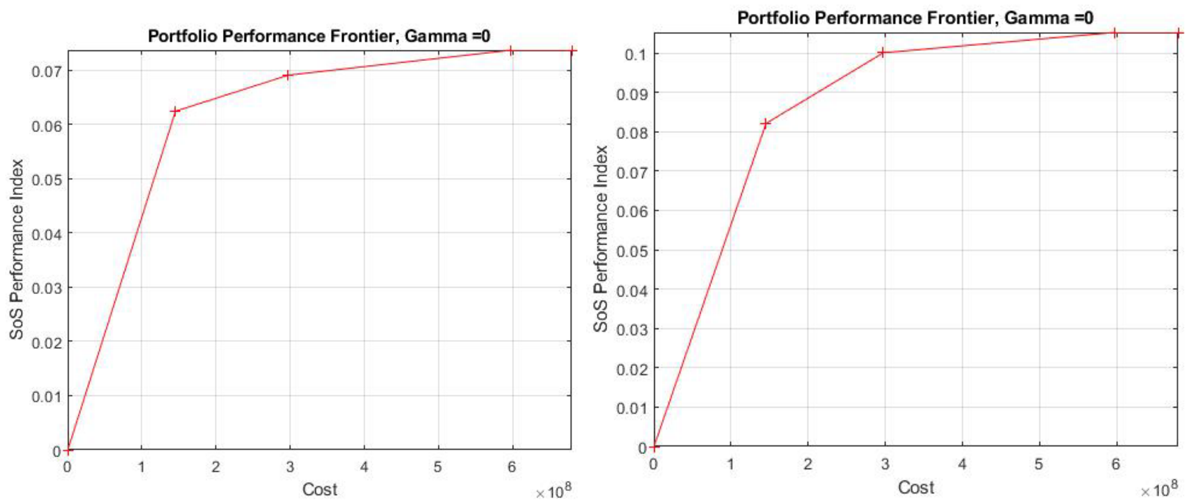

Figure 7. Test runs with variation in weight distribution

First, each of these portfolio performance frontiers identify the best possible solution (architecture) for a given cost. Every distinguishable point on the frontier is a feasible architecture (one column from Figure 6). With an increase in budget, as expected, better performing systems are acquired to form the SoS architecture. This results in better performing SoS architectures within the same scenario. Second, upon closely inspecting and comparing the two pareto frontiers it is evident that while the shape/form of the two is similar, the data points are not the same. This indicates that different weight preferences for the SoS capability produce portfolios that provide different performances. This is clearly visible when multiple pareto frontiers from various cases in the experiment are plotted in the same graph, as shown in Figure 8. We can notice that any uncertainty in SoS capability preferences (while setting up the acquisition problem) affects the resulting performance of the SoS portfolios. For example, Case 1 had a weight of 0.8 (out of 1) and Case 29 had 0.1 for Air Superiority and as stated the SoS performance index for their respective portfolios are inversely related to the value of the assigned weights. Leading to two portfolios with a sizeable difference in their performance index. Another influencing factor in any acquisition problem is the restrictive nature of the proposed budget i.e., cost. By using RPO, the accountability of cost-based comparisons is visible too. Instances where the performance index of a portfolio for one case (case 26) is higher than the other (case 22) for a specified cost value. However, with an increase in costs to a higher value, the previous trend does not hold true.

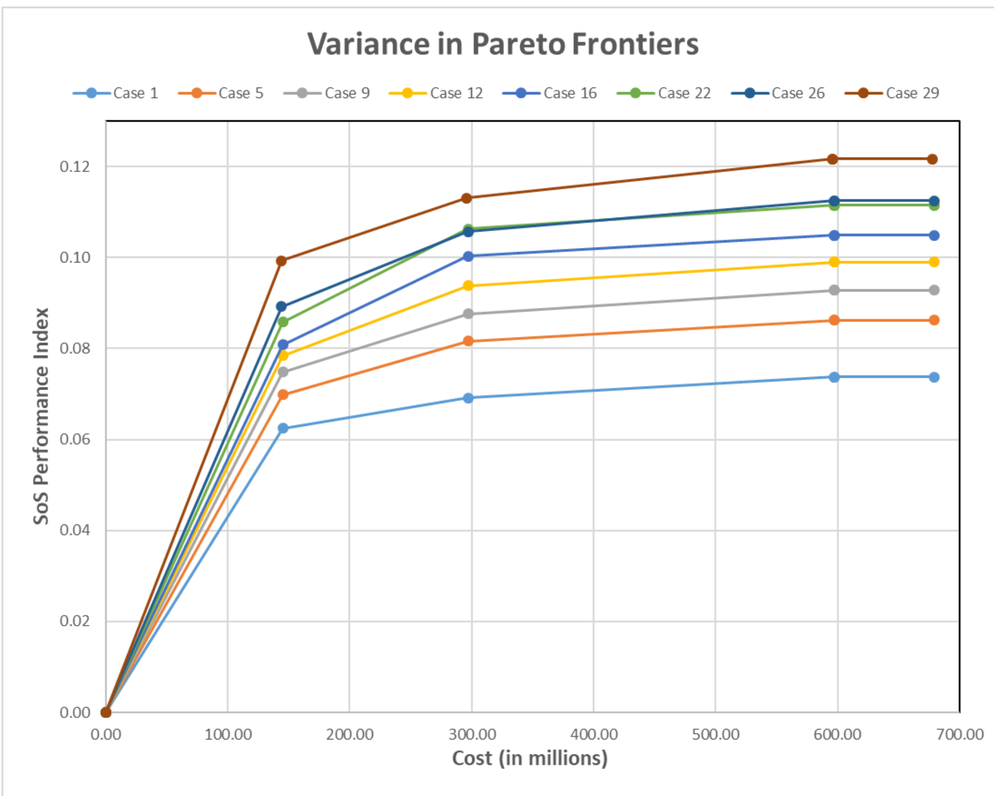

Figure 8. Variation in pareto frontiers across the cases

Optimal acquisition with uncertainty on external factors

As a proof of concept, we consider the case study of Urban Air Mobility (UAM) in the Dallas, TX region. We model the expected user traffic in this large transportation SoS by looking at previous years data from the North Central Texas Council of Governments (NCTCOG) data on population travel frequency between home and work districts. We then, with uncertainty bounds, predict the proportion of travelers willing to use the UAM SoS as an alternative to driving/public ground transport. Within this context, we consider that UAM operators might be carrying on their activities in a competitive market, where other stakeholders are present. Based on the amount of available data concerning the travelers' decision (in its turn, influenced by income, cost of commuting by UAM vehicles and perceived value of time), the presence of competition, and some degree of uncertainty, we use prescriptive techniques to identity optimal choices for UAM operators to compete in the market. The choices concern the number and type of UAM vehicles to be operated on each available travel route. In the future, ML techniques will be used to evaluate why certain strategies are optimal based on different amounts of available data, and to predict the best strategy in the presence of multiple stakeholders and data uncertainty.

Concept Application: Urban Air Mobility

Three regions of Dallas (A, B & C) are modelled in our study as potential UAM points of operation based on the NCTCOG data on population travel frequency. We model 6 routes operating between these three regions. Additionally, the stakeholder is provided with acquisition decisions for the type of UAM vehicles with varying seating capacity (1, 2 & 4 max. passengers). The stakeholder decisions to build their UAM portfolio are motivated to maximize expected total income per day. Our predictive model uses historical data from NCTCOG to predict the population travel frequency for 2022 for which the UAM network is being modelled as the acquisition problem. Then a range of feasible portfolios are built based on stakeholder constraints. These portfolios are then optimized for the highest expected income. We also perform this predictive and prescriptive analysis with a 95% confidence interval to account for uncertainty in known data. We also conduct studies involving market competition to study how that affects acquisition of UAM vehicles and structure of the network.

Urban Air Mobility – Results

In the first example, we assumed perfect information about the number of passengers willing to fly with UAM vehicles, based on their income, the alternative ground-based travel time, and the personal perception of the value of time. The only source of uncertainty in this case is due to the prediction of travel frequency for 2022, based on past travel data starting in the year 2010. Since the available dataset from NCTCOG is relatively small, we used a linear fit. We also assumed perfect knowledge about the share of the market which is already taken by existing competitors. The income is based on ticket price (varying by route and type of vehicle) and on operational costs. We ran an integer linear optimization, where the decision variables are the number of vehicles per type per route, and the number of passengers actually flying with the stakeholder's vehicles. Constraints exist on the maximum number of vehicles that can be acquired and on the maximum number of flights per vehicle per day on each route. Results of the optimization based on the expected values for flying passengers in 2022 are shown in table 3.

Table 3. Optimal choices with full data available and prediction for the year 2022 in the Dallas area.

Route	**1-passenger vehicles**	**2-passenger vehicles**	**4-passenger vehicles**
AB - BA	209	0	83
AC - CA	31	160	0
BC - CB	0	0	157
Passengers per day	2522	3840	11448
Income	$ 1,954,910.73		

Results show that on two routes with more passenger availability, larger vehicles are preferable even if they produce less income per passenger. The intermediate-sized vehicle is present only on two routes, together with the small vehicle. The expected income is about $1.955 million. However, due to uncertainty, the actual income will be slightly smaller. We ran 1000 scenarios according to the expected distribution and using the optimal choice of vehicles, which resulted in an income of $1.883 million.

As a first step towards the study of support for decision-making in SoS, we ran a scenario where the actual market share is unknown. We assumed that 33% of the passengers is available to fly with us, which is only slightly different from the actual market share that we used in the first case (ranging between 30% and 45%). We also increased the uncertainty in the predictive phase. Despite the small differences, there are already changes in the choice of optimal fleet, as shown in table 4. The routes AB and BA saw an increase in the number of small vehicles, while more large vehicles are acquired for the routes BC and CB.

Table 4.Optimal choices with partial data available and prediction for the year 2022 in the Dallas area.

Route	1-passenger vehicles	2-passenger vehicles	4-passenger vehicles
AB - BA	218	0	67
AC - CA	22	160	0
BC - CB	0	0	173
Passengers per day	2488	3840	11832
Income	$ 1,956,211.60		

The expected larger market share along the routes BC and CB suggests an income slightly larger than the previous case. However, despite the very small differences, the presence of incomplete data causes suboptimal acquisition. When running 1000 scenarios according to the expected distribution of flying passengers, the resulting income is $1.858 million, lower than the income achieved with the optimal choice in the previous case.

Further study in this area includes stochastic optimization, to minimize the risks in case of incomplete information, and use of ML techniques to identify the patterns of optimal acquisition choices.

Outcomes of Acquisition Decision-making from Concept Problem Demonstrations

Results from our concept problems demonstrate the following contributions of our research:

1. In the first problem, we demonstrated the use of a Decision Support Framework to optimize portfolios of systems that can maximize the required SoS capabilities with constraints of budget limit and risk aversion.

2. The quantitative framework is based on a highly flexible mathematical modeling technique that can incorporate many types of connectivity behaviors between complex system modules (here, individual systems).
3. The comparison of optimal portfolios in cases where the perceived relative importance of SoS capabilities is different allows the user to recognize commonalities between the different solutions and to guide acquisition when multiple stakeholders do not agree upon the objectives.
4. In the second problem, we showed how the presence of incomplete data, together with uncertainty, can produce suboptimal choices in acquisition. Preliminary results showed promising directions for the use of predictive and prescriptive analytics to address this type of problems.

THIS PAGE LEFT INTENTIONALLY BLANK

Potential Next Steps

We suggest the following steps as fruitful for additional research:

- Further research and expansion of the case studies, to include non-recurrent and recurrent costs, and stochastic optimization in support of acquisition decision-making.
- Use of Machine Learning techniques to identify patterns that characterize optimal choices in presence of uncertainty and incomplete information. This has a twofold goal: modeling the results of optimal and suboptimal choices, and informing stakeholders on data sets that can produce the highest improvements to the SoS capabilities.

THIS PAGE LEFT INTENTIONALLY BLANK

References

A Basic Introduction To Neural Networks. (n.d.). Retrieved April 9, 2020, from http://pages.cs.wisc.edu/~bolo/shipyard/neural/neural.html

Apte, U., Rendon, R., & Dixon, M. (2016). *Big data analysis of contractor performance information for services acquisition in DoD: A proof of concept*. Naval Postgraduate School Monterey United States.

Berwick, R. (n.d.). *An Idiot's guide to Support vector machines (SVMs)*. 28.

Brotherton, T., & Johnson, T. (2001). *Anomaly detection for advanced military aircraft using neural networks*. *6*, 3113–3123.

Davendralingam, N., & DeLaurentis, D. A. (2015). A robust portfolio optimization approach to system of system architectures. *Systems Engineering*, *18*(3), 269–283.

Davendralingam, N., Maheshwari, A., Raz, A. K., & DeLaurentis, D. A. (n.d.). *Predictive and Prescriptive Analytics via System of Systems Engineering: Informational Determinants for Enterprise-Level Effectiveness*.

Davenport, T. H. (2013, December 1). Analytics 3.0. *Harvard Business Review*, *December 2013*. https://hbr.org/2013/12/analytics-30

Explained: Neural networks. (n.d.). MIT News. Retrieved April 9, 2020, from http://news.mit.edu/2017/explained-neural-networks-deep-learning-0414

Freeman, C. E. (2013). *Multivariate and Naïve Bayes text classification approach to cost growth risk in Department of Defense acquisition programs*.

Guariniello, C., & DeLaurentis, D. (2017). Supporting design via the system operational dependency analysis methodology. *Research in Engineering Design*, *28*(1), 53–69.

Hand, D. J., Mannila, H., & Smyth, P. (2001). *A Systematic Overeview of Data Mining Algorithms*.

Huang, C., & Mintz, A. (1990). Ridge regression analysis of the defence-growth tradeoff in the United States. *Defence and Peace Economics*, *2*(1), 29–37.

INFORMS. (n.d.). *History of OR: Useful history of operations research*. INFORMS. Retrieved April 9, 2020, from https://www.informs.org/ORMS-Today/Public-Articles/June-Volume-42-Number-3/History-of-OR-Useful-history-of-operations-research

Joseph, R. C., & Johnson, N. A. (2013). Big Data and Transformational Government. *IT Professional*, *15*(6), 43–48. https://doi.org/10.1109/MITP.2013.61

Lect12.2up.pdf. (n.d.). Retrieved April 9, 2020, from http://mi.eng.cam.ac.uk/~mjfg/local/4F10/lect12.2up.pdf

Lecture 2: K-nearest neighbors / Curse of Dimensionality. (n.d.). Retrieved April 9, 2020, from https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02_kNN.html

Lo, A. W., & Hasanhodzic, J. (2011). *The Evolution of Technical Analysis: Financial Prediction from Babylonian Tablets to Bloomberg Terminals*. John Wiley & Sons.

Lynch, P. (2008). The ENIAC Forecasts: A Re-creation. *Bulletin of the American Meteorological Society*, *89*(1), 45–56. https://doi.org/10.1175/BAMS-89-1-45

MacKenzie, D. I., Nichols, J. D., Royle, J. A., Pollock, K. H., Bailey, L., & Hines, J. E. (2017). *Occupancy estimation and modeling: Inferring patterns and dynamics of species occurrence*. Elsevier.

Maier, M. W. (1998). Architecting principles for systems-of-systems. *Systems Engineering*, *1*(4), 267–284. https://doi.org/10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D

Moore, G. W., & White III, E. D. (2005). A REGRESSION APPROACH FOR ESTIMATING PROCUREMENT COST. *Journal of Public Procurement*, *5*(2).

Naive Bayes—Scikit-learn 0.22.2 documentation. (n.d.). Retrieved April 9, 2020, from https://scikit-learn.org/stable/modules/naive_bayes.html

Regression Analysis Software | Regression Tools | NCSS Software. (n.d.). Retrieved April 9, 2020, from https://www.ncss.com/software/ncss/regression-analysis-in-ncss/#Ridge

Rehman, M. H. ur, Chang, V., Batool, A., & Wah, T. Y. (2016). Big data reduction framework for value creation in sustainable enterprises. *International Journal of Information Management*, *36*(6, Part A), 917–928. https://doi.org/10.1016/j.ijinfomgt.2016.05.013

Summary Points for Logistic Regression | STAT 504. (n.d.). Retrieved April 9, 2020, from https://online.stat.psu.edu/stat504/node/164/

Supervised Learning vs Unsupervised Learning—Best 7 Useful Comparison. (2018, April 29). *EDUCBA*. https://www.educba.com/supervised-learning-vs-unsupervised-learning/

Waller, M. A., & Fawcett, S. E. (2013, June 1). *Data Science, Predictive Analytics, and Big Data: A Revolution That Will Transform Supply Chain Design and Management*. Journal of Business Logistics. https://onlinelibrary.wiley.com/doi/abs/10.1111/jbl.12010

Wang, R., & Yang, W. (2016). Estimation of demand and supply in US airline industry. *Unpublished Working Paper*.

Wei, K., Wu, J., Ma, W., & Li, H. (2019). State of charge prediction for UAVs based on support vector machine. *The Journal of Engineering*, *2019*(23), 9133–9136.

Xiao, H., Cai, C., & Chen, Y. (2006). Military vehicle classification via acoustic and seismic signals using statistical learning methods. *International Journal of Modern Physics C*, *17*(02), 197–212.

Zainol, Z., Azahari, A. M., Wani, S., Marzukhi, S., Nohuddin, P. N., & Zakaria, O. (2018). Visualizing military explicit knowledge using document clustering techniques. *International Journal of Academic Research in Business and Social Sciences*, *8*(6), 1127–1143.

Zhao, P., & Yu, B. (2006). On model selection consistency of Lasso. *Journal of Machine Learning Research*, *7*(Nov), 2541–2563.

Acquisition Research Program
Graduate School of Defense Management
Naval Postgraduate School
555 Dyer Road, Ingersoll Hall
Monterey, CA 93943

www.acquisitionresearch.net