A METHOD FOR VISUALIZING THE STRUCTURAL COMPLEXITY OF ORGANIZATIONAL ARCHITECTURES

A Thesis

presented to

the Faculty of California Polytechnic State University,

San Luis Obispo

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Aerospace Engineering

by

Jacob King

March 2021

COMMITTEE MEMBERSHIP

TITLE:	A METHOD FOR VISUALIZING THE STRUCTURAL COMPLEXITY OF ORGANIZATIONAL ARCHITECTURES
AUTHOR:	Jacob King
DATE SUBMITTED:	March 2021
COMMITTEE CHAIR:	Dianne DeTurris, Ph. D. Professor of Aerospace Engineering
COMMITTEE MEMBER:	Paul Choboter, Ph. D. Assistant Professor of Mathematics
COMMITTEE MEMBER:	Aaron Drake, Ph. D. Professor of Aerospace Engineering
COMMITTEE MEMBER:	Andrew Palmer Engineer Associate – Systems, Ball Aerospace

ABSTRACT

A Method for Visualizing the Structural Complexity of Organizational Architectures

Jacob King

To achieve a high level of performance and efficiency, contemporary aerospace system must become increasingly complex. While complexity management traditionally focuses on a product's components and their interconnectedness, organizational representation in complexity analysis is just as essential. This thesis addresses this organizational aspect of complexity through an Organizational Complexity Metric (OCM) to aid complexity management. The OCM augments Sinha's structural complexity metric for product architectures into a metric that can be applied to organizations. Utilizing nested numerical design structure matrices (DSMs), a compact visual representation of organizational complexity was developed. Within the nested numerical DSM are existing organizational datasets used to quantify the complexity of both organizational system components and their interfaces. The OCM was applied to a hypothetical system example, as well as an existing aerospace organizational architecture. Through the development of the OCM, this thesis assumed that each dataset was collected in a statistically sufficient manner and has a reasonable correlation to system complexity. This thesis recognizes the lack of complete human representation and aims to provide a platform for expansion. Before a true organizational complexity metric can be applied to real systems, additional human considerations should be considered. These limitations differ from organization to organization and should be taken into consideration before implementation into a working system. The visualization of organizational complexity uses a color gradient to show the relative complexity density of different parts of the organization.

Keywords: Structural Complexity, Design Structure Matrix, Organizational Metric

ACKNOWLEDGMENTS

I would like to thank Dr. DeTurris and the rest of my committee for being patient and working with me through the highs and lows of my thesis. Also, I would like to thank my parents for continuing to support my endeavors. I would not be in the position to write a master's thesis without them providing me with every opportunity to pursue my academic goals.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

$A_{i,j}$	adjacency matrix of i, j elements
C	total structural complexity
C_1	component complexity
C_2	interface complexity
C_3	topological complexity
F_c	complexity factor
$R^{(i)}$	vector of component characteristic ranks
$X_k^{(i,j)}$	vector of component characteristics
$c^{(k)}$	interface type characterization
f_i	percentile rank of the i^{th} value (discrete)
f_j	percentile rank of" j^{th} value (continuous)
$r_j^{(i)}$	percent rank w.r.t. variable $x_j^{(i)}$
$w_j^{(i)}$	weight assigned to j^{th} factor for i^{th} component
$x_j^{(i)}$	characteristics vector of interface i, j
$y_{i,j}$	development cost/performance of i^{th} and j^{th} component
$z_{i,j}^{(k)}$	interface development cost/performance of i^{th} and j^{th} interface type
α_i	component complexity of component i
$\beta_{i,j}$	interface complexity of interface i, j
σ^2	standard deviation
μ	distributions mean
$E(A)$	matrix energy of matrix A
L	number of data values that are less than the j^{th} value
N	number of total data values
TRL	technology readiness level

Chapter 1
INTRODUCTION

Aerospace systems provide a unique challenge in the systems engineering domain as they can be complex. Managing complexity has become a key focus of development teams, as there is a direct relationship between developing complex systems and cost overrun. System complexity is derived from a system's large number of components and their interwoven interactions. As aerospace systems grow larger, system complexity inherently increases. In 2014, Dr. Kaushik Sinha addressed quantitative complexity analysis by considering system structure and arrangement to develop a structural complexity metric. Sinha's structural complexity metric considers the size and geometry of the information interfaces, as well as the complexities associated with individual subsystems/components (Sinha, 2014). Additionally, Sinha's structural complexity metric considers the modularity of system structures and accounts for their integrations by showing the scalability of the complexity metric (Sinha, 2014).

This thesis expands upon Sinha's product architecture structural complexity metric to develop an organizational complexity metric (OCM). With the aim of managing complexity throughout the life cycle process, the organization of human resources plays an essential role in system conception, development, and utilization. While Sinha used engineered products as the basis for the metric, many parallels can be drawn between product architectures and organizational architectures. Having teams consisting of complex human individuals interacting within diverse organizational structures, mismanaging information flow could contribute to overall system complexity. Using a human-centered approach along with the application of the modified complexity metric to organizational systems could help manage the ever-growing complexity of modern aerospace systems.

1.1 Statement of Purpose

The purpose of this thesis is to investigate the legitimacy of applying a modified complexity metric to organizational architectures. With differing team organization methodologies varying from industry to industry, this thesis aims apply the organizational complexity metric (OCM) to organizations of differing sizes and arrangements. This thesis will test the legitimacy of taking an organizations topology, interfaces, and individuals into account when attempting to analyze complexity.

1.2 Purpose of Study

The underlying premise of this thesis is the relation of large costs overruns and overall system complexity. With system complexity studies focusing on the engineered product itself, there is an equal importance tied to the organizational architecture of the engineering teams that develop these products. With human interactions showing complex properties that reflect those of the products they are developing, understanding human influence is essential to analyzing system complexity. This thesis will investigate the need for human and organizational considerations when attempting to understand system complexity.

Chapter 2

LITERATURE REVIEW

Complexity management will be discussed in the context of engineering systems. More specifically, structural complexity in an organizational context and how structural complexity is managed will be explored. An overview of Bearden's ranking system and Sinha's product architecture complexity metric will supplement the proposed OCM methodology in Chapter 3. Lastly, an important aspect of the OCM is how a system is to be represented. For the OCM, nested numerical DSMs will be implemented to better communicate the relative complexity densities throughout the organization. These areas of high complexity provide insight into potential problem areas within an organization. Once these problem areas are identified, it is the job of systems engineers to monitor and support these areas as needed.

2.1 Complexity in Engineering Systems

As stated before, the challenges with contemporary systems are a result of being complex. However, complexity in a systems sense is more than sheer size and scale. Complexity in modern engineering systems can be attributed to the interdependence of components and their interactions. In the context of this thesis, a system that is large in scale, interlaced, and can exhibit unpredicted behavior is considered complex. For example, an automobile, while complex, shows fewer unpredictable tendencies than a system such as an extremely intelligent software system. While an automobile is made up of numerous components from differing disciplines, the operation of an automobile is relatively linear and predictable. As for a grand software system, different modules have the capability of running in parallel, while exchanging information between the modules. This nonlinear interdependence is a large contributor to unpredictability and, therefore, is considered complex. This complexity grows in aerospace systems, as the systems are an amalgamation of both complex physical and cyber systems. Because of the scale of these systems, only the organization of individuals (or teams) can bring systems such as these into existence. Complexity is defined by the unpredictability or emergence of certain system behaviors due to the interwoven components and their overlapping responsibilities. This emergent behavior makes it difficult to predict system development and performance contributing to reductions in reliability and robustness (Sinha, 2014).

System complexity is not inherently negative. System complexity can improve performance and system robustness and is often necessary to meet minimum acceptable operational requirements. This can be seen directly with the advancement in technology over the past century. As time passes, systems are asked to do more tasks more effectively than ever before and are becoming more complex. Much like other engineering metrics, balancing complexity with performance and functionality is assigned to the systems engineer. A systems engineer must manage complexity within a system while meeting a minimum performance requirement. Complexity can be directly related to organizational environments, as an increase in organizational complexity could affect communication capability. The connection between communication and increased complexity, as well as exploring ways of quantifying and contextualizing this complexity are the focus of this thesis.

2.2 Structural Complexity in Organizational Systems

Along with the need for managing the technical elements of engineering systems, interfaces and relationships between people must be managed. The limitations and structures of development teams typically echo the structures of the products they are developing (Madni, 2011). Therefore, having inefficient organizational structures can lead to inefficient product architectures. While managing product architectures has its own challenges, due to products having technical rationale driving design, managing humans within an organization is far more difficult. Although humans can be modelled as logical beings, this is not necessarily the case. Humans rely on both conscious and subconscious thought processes that drive decisions, making emergent behavior a concern when conceptualizing numerous interdisciplinary development teams (Madni, 2011). Individuals within a team have differing cognitive capacities, biases, and access to information that can contribute to the structural inefficiency of these organizations (Madni, 2011).

With access to information never truly being complete, cognitive capabilities having limits and time constraints, decisions are forced to be made before an individual is ready. This issue is only compounded as complexity increases, since the amount of necessary information increases with system complexity (Madni, 2011). An increase in information leads to an increase in time needed and as a result, decisions made are less likely to be supported with sufficient backing. As a result, it becomes challenging to intuitively manage complex problems (Madni, 2011).

2.3 Managing Structural Complexity Through Modularity

One way of addressing system complexity is through proper system representation. System elements and organization elements can be modelled through network graphs, matrices, and other numerical and graphical representations to better understand the interconnections between these elements. Even a simple method of graphical representation provides an avenue for managing this complexity. Specifically, interdependence and modularity become easily measurable. These interdependencies and modular characteristics directly correlate to technical subsystem and organizational development team conception.

Modularity is an essential tool for managing complexity. Modularity is defined as a continuum describing the degree to which a system's components may be separated and recombined (Schilling, 2000). In an organizational context, modularity "within and among organizations mirror the degree of product modularity, with the main consequence that independent companies may develop, produce and deliver selfcontained modules consistent with the scope and depth of their core competences" (Campagnolo and Camuffo, 2010). This emphasizes that modularity is not only a product characteristic, but also an organizational characteristic. Efforts have been made to improve modularity and manage complexity through the implementation of distributed organizational hierarchies and interface mechanisms. This is further emphasized by Campagnolo and Camuffo:

> *Integral products should be developed by integral organizations (tightly connected organizational units to maximize ease of communication and minimize the risk of opportunism). Modular products should be developed by autonomous, loosely coupled, easily reconfigurable organizations. Indeed, the adoption of standards reduces the level of asset specificity (Argyres, 1999) and, in turn, the need to exercise managerial authority. Product modularity also reduces the need for communication due to information hiding, whereby knowledge about the 'interior' of each module does not need to be shared. (Campagnolo and Camuffo, 2010).*

This highlights that the management of modularity is an underlying characteristic of existing tools, such as, work breakdown structures. However, the challenges associated with modularity can lead to inefficiencies and unpredictability. If a system's modules are not correctly interwoven, these inefficiencies begin to surface as developmental and operational shortcomings.

2.4 System Representation Through Design Structure Matrices (DSMs)

An obvious contributor to product conception, development, and utilization success are adequate project and program managers facilitating information across different development teams (Heaslip, 2015). Managers must struggle with the balance of allowing the free flow of information while avoiding information overload. For example, some employees are asked to manage hundreds of communication channels from varying sources. This presents a potential disconnect between the sender and the receiver of the information. The sender may assume they completed an information exchange that was fully understood, while the receiver may have missed the message and/or read and misinterpreted the information. In some ways, this form of communication disconnect can be more harmful to system development, as a misunderstanding is buried in the interwoven nature of organizational teams (Heaslip, 2015). One way to combat this disconnect are frequent meetings; however, meetings can be time consuming and interrupt productive working hours. As a result, it is important for managers to design organizational architectures with purposeful structure. Properly managing these information channels and anticipating potential communication shortcomings could be contextualized through complexity management (Heaslip, 2015).

2.4.1 Introduction to Organizational Architecture DSMs

An organizational architecture is the structuring of people to work together to accomplish tasks (Eppinger, 2012). Figure 1 shows how organizational architectures can be broken down into three components: business units, departments, and individuals. Traditionally these components are modelled in organization charts or by an organization breakdown structure (OBS); however, OBSs primarily focus on the decomposition units, leaving out valuable information regarding the rest of the organization structure (Eppinger, 2012). DSMs provide the benefits of decomposition and defined roles while maintaining unit and lateral relationship information. Organizational DSMs can capture key workplace breakdown information, as well as interunit relationships while maintaining visibly efficient interpretation (Eppinger, 2012).

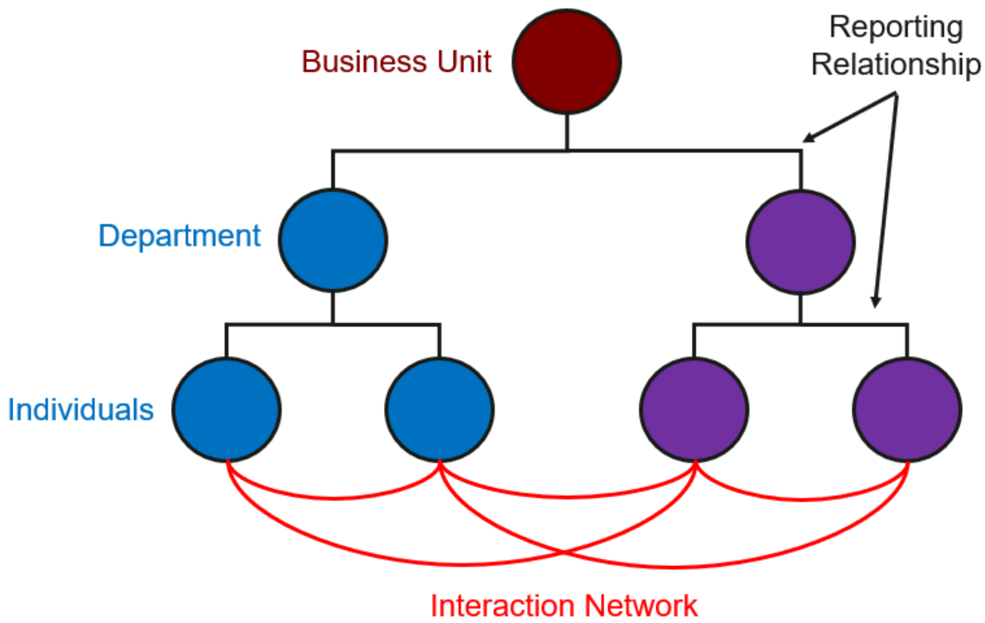


Figure 1. Organizational Architecture Representation (Eppinger, 2012)

Figure 2 shows an organizational architecture captured within a DSM. Within the DSM, the rows (i) and columns (j) of the matrix are filled with individuals (i = j) with their given communication interfaces (i, j) represented with a binary assignment. Additional department or development team modulization is captured through the pink boundary encapsulating the individuals.

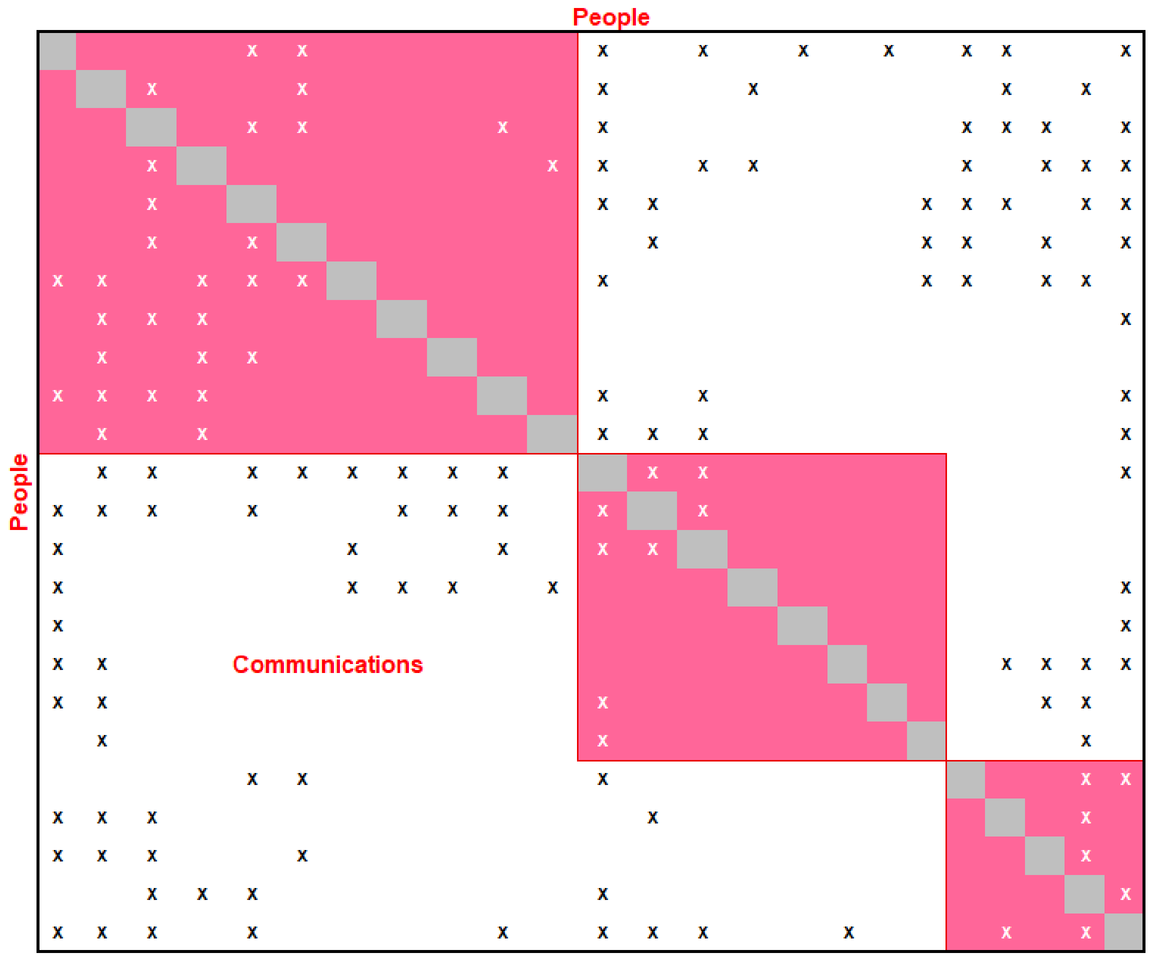


Figure 2. Organizational Architecture within a DSM (Eppinger, 2012)

These boundaries have subjectivity, as it is the job of the systems engineer to represent the organization effectively using a DSM. Although subjective, the nature of the DSM is the highlighting of organizational interactions. A DSM can be sequenced and modularized to better group individuals based on the frequency of interaction. In other words, those found within a module interact with one another more frequently than those outside of the module. This characteristic of DSMs provides valuable insight in an organizational context, as communication across these module boundaries carry significantly more weight than internal communications. To better explain the significance, imagine an individual communicates with their team daily, but meets biweekly with an individual with another team to discuss integration. If the individual were to miscommunicate within the team, a problem can be fixed (hypothetically) within the next day. As for a miscommunication with an individual in another module, a mistake might not be acknowledged until the next meeting (or longer). The increase in time between meetings compounds the miscommunication. As a result, the organization loses out financially, due to reworks and schedule delays. Because of this extreme weight associated with inter-team, interdisciplinary, and interorganizational communication, having a management tool to signal these potential problem areas is extremely useful. Traditional organizational structures tend to promote internal communication but struggle to define external communication pathways. The ability to model and manage the internal and external communication channels and their integrations are essential as it allows for the modification of existing structures to account for deficiencies (Eppinger, 2012). This is a strength of the DSM, which can be scaled up or down depending on the desired fidelity. Additionally, DSMs promote modular representation and allow for additional analysis of internal-external relationships.

Communication interfaces within these organizational networks greatly outpace the number of individuals leading to the management of these interfaces becoming increasingly challenging. The conception of misinformation can be attributed to these interfaces and only grows with complexity (Eppinger, 2012). This misinformation can lead to delays, recalls, and even cancellations. Building on the DSM model, managing organizational complexity could become more procedural for a systems engineer or project manager.

2.4.2 Implementation of Nested Numerical DSMs

The DSM presents key organizational information needed for the calculation of organizational complexity. A system's topological arrangement, interfaces, and individual components are all presented in an easily understandable matrix.

Compounding the initial benefits a DSM provides for system representation, additional numerical information can be stored within these matrices for further organizational analysis. To maintain the locality of data and its ease of access, nested matrices will be implemented. Nested numerical DSMs allow for surface level complexity metrics to be presented, with the option to show a more detailed view of complexity using a denser DSM. An example of this method can be seen in Figure 3 where an individual 'A' has its overall complexity contribution displayed; however, within this matrix representation are higher fidelity matrices showing the sub-contributors of complexity. In other words, an "average" complexity of individual 'A' can be viewed on the surface, while the complexity portfolio can be viewed within the cell, upon further inspection.

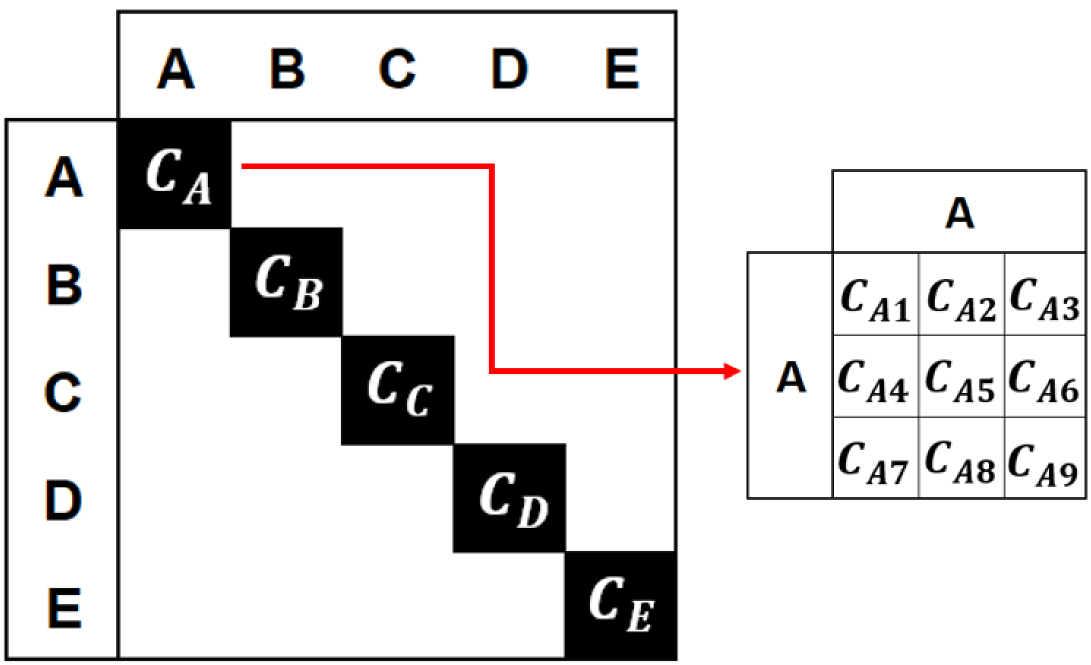


Figure 3. Example of Nested DSM for Data Storage and Representation

2.5 Bearden's Complexity Index and Ranking System

Bearden's complexity index uses performance, mass, power, and technological choices to determine the representation of a system's complexity for purposeful comparison (Sarsfield, 1996). The complexity contribution uses a matrix of system characteristics and performance data to develop a complexity measure based on the baseline data (Bearden, 2000). Data within the matrix can be both subjective and objective in nature. Everything from reusability to mass is quantifiable and is only limited by the amount of data available.

At its core, the complexity contribution is simply defined. The process for measuring a system's complexity is as such (Aerospace Corp, 1977):

1. *Identifying the parameters that drive spacecraft design.*
2. *Quantify the identified parameters.*

The quantification of these parameters is determined by whether the data is continuous or discrete. If a system must decide on a finite system characteristic (such as fuel type, number of engines, et cetera), there are a discrete number of possibilities that can be summed and averaged based on their assigned values. Discrete choices are defined as:

$(0, 1)$ For two options

$f_i = \left(0, \frac{1}{2}, 1\right)$ For three options, et cetera

$i = 1 \dots m$ m ≡ the number of discrete choices

f_i ≡ percentile rank of i^{th} value (discrete)

For continuous datasets, a percentile rank calculation is employed. This function is used to determine the relative standing of the value within the dataset. Continuous choices are defined as:

$$f_j = \left(\frac{L_j}{N-1}\right)(100) \; ; \; j = 1 \dots N$$

f_j ≡ percentile rank of j^{th} value (continuous)

L ≡ number of data values that are less than the j^{th} value

N ≡ number of total data values

3. *Combine these parameters into an aggregated complexity contribution.*

Combining both discrete and continuous datasets define the complexity factor, F_c. Eq. (1) defines this factor as:

$$F_c = \left(\sum f_i + \sum f_j\right) / (m + n) \tag{1}$$

The mean of the complexity factor is determined by the average of individual factors, while the minimum and maximum values are taken from the extremums in the dataset. Each datapoint is weighted equally to remove a parameter from dominating complexity but can be expanded upon. This factor uses the minimum and maximum values to produce a normalized complexity metric between 0 and 1.

The uses of the complexity contribution are flexible and can be applied to a myriad of predictive functions. However, Bearden's complexity contribution will be expanded upon in the following section and will define the use of the complexity contribution and ranking system in terms of structural complexity.

2.6 Existing Structural Complexity Metric for Product Architectures

As alluded to previously, product complexity and organizational complexity mirror one another, due to the close relationship between those developing the systems and the systems themselves. In this section, Dr. Kaushik Sinha's metric will be discussed in detail, as Sinha's metric acts as the basis for the proposed OCM.

2.6.1 Overviewing Structural Complexity Quantification of Product Architectures

Structural complexity is a measurable characteristic and is attributed to the internal complexities of system components, the interactions between these components, as well as the organization and arrangement of these elements, and their connections. Sinha's metric for estimating structural complexity is defined as such:

$$C = (C_1) + (C_2)(C_3) \tag{2}$$

Component Complexity, C_1, relates to the component engineering activity within a system development effort and does not involve architectural information (Sinha, 2014). Interface Complexity, C_2, represents the number of pair-wise interactions, along with the number of these interactions and how it relates to interface management (Sinha, 2014). Lastly, Topological Complexity, C_3, expresses the arrangement of interfaces with respect to the system's top-level architecture. C_3 is used to simulate the difficulties associated with system integration (Sinha, 2014).

2.6.2 Functional Form of the Structural Complexity Metric

Expanding Eq. (2), the analytical form of the structural complexity metric is as follows:

$$C(n, m, A) = \left(\sum_{i=1}^{n} \alpha_i\right) + \left(\sum_{i=1}^{n}\sum_{j=1}^{n} \beta_{i,j} A_{i,j}\right)\left(\frac{E(A)}{n}\right) \quad (3)$$

The component complexities, α_i, are attached to their corresponding compositional elements; therefore, localized to their element. $\beta_{i,j}$ represents the pair-wise interfaces and their complexities. The third term, $E(A)$, represents the underlying connectivity structure through the implementation of the matrix energy (Sinha, 2014) associated with the system's adjacency matrix (A) of n components with m interfaces. This connectivity complexity is defined as topological complexity, which generally scales with architecture size and integration. Higher topological complexity will likely signal the lengthening of system integration efforts and represents a global property of the system.

2.6.2.1 Estimating Component Complexity, C_1

When characterizing component complexity, it is important to note that the localized complexity of a system's elements is domain dependent. What this indicates is a need for domain information, data, or other forms of insight to make correct estimations. Component complexity is dependent on the technical design and development difficulty with the component alone, not including its interfaces (Sinha, 2014). As the available data varies from project to project, estimation methodologies depend on the readily available information. As a result, three methods are suggested: 1) estimation based on technological maturity, 2) estimation of component complexity with expert opinion, and 3) estimation of component complexity with data analytics (Sinha, 2014).

2.6.2.1.1 Estimating Component Complexity Using Technological Maturity

Using technological maturity as a metric assumes the proportionality of technology readiness to component complexity (α_i). If a technology has been developed over a relatively long period of time, it is assumed that the processes and operating principles have been matured as well (Sinha, 2014). This leads to the proposed implementation of a scaled technology readiness level (TRL) (Sadin et al., 1988):

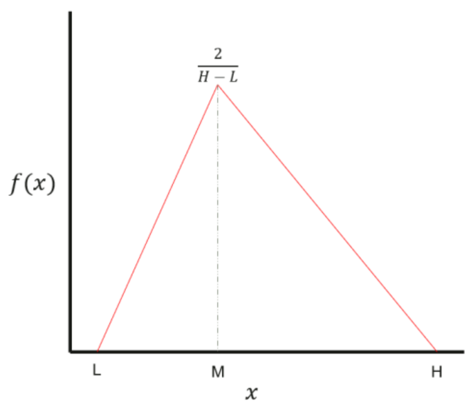
$$\alpha_i = 5\left(\frac{TRL_{max} - TRL_i}{TRL_{max} - TRL_{min}}\right) \tag{4}$$

With the interval defined as [TRL_{min}, TRL_{max}] for calculating the ith component in a system. Component complexity is then scaled from the TRL to a continuous interval of [0, 5] with a higher component complexity value denoting a higher component complexity (Sinha, 2014). This method for defining component complexity is beneficial specifically for companies with rigorous definitions of TRL.

2.6.2.1.2 Estimation Component Complexity Using Expert Opinion

Using expert opinion to estimate component complexity is valuable when no significant databases are available. When eliciting expert opinions, a structured approach to capturing the subject matter expert's knowledge quantitatively is needed.

With only a limited data sample, a triangular distribution is useful when an expert has a certain confidence level for estimations (Sinha, 2014). Information gathered from the expert can be captured in the triangular distribution given a range of minimum (L) and maximum values (H) and a most likely case (M). Figure 4 represents that distribution and is supplemented by Eq's. (7) and (8) for the distributions mean (μ) and standard deviation (σ^2).



$$f(x) = \frac{2(x - L)}{(M - L)(H - L)};\ L \leq x \leq M \qquad (5)$$

$$f(x) = \frac{2(H - x)}{(H - M)(H - L)};\ M \leq x \leq H \qquad (6)$$

$$\mu = \frac{L + M + H}{3} \qquad (7)$$

$$\sigma^2 = \left(\frac{L^2 + M^2 + H^2 - LM - LH - MH}{18}\right) \qquad (8)$$

Figure 4. Triangular Distribution Using Expert Elicitation (Sinha, 2014)

2.6.2.1.3 Estimation Component Complexity Using Data Analytics

Lastly, if sufficient data is available, the implementation of component data would be implemented. Applications of Sinha's method use the statistical model of form $\alpha = f(X) = f(x_1, \ldots, x_i)$ relating component complexity with a vector of component characteristics, X (Sinha, 2014). This vector can be adjusted from system to system, but is suggested as follows (Sinha, 2014):

1. **Measure of performance tolerance, x_1:** Components with extremely tight performance tolerance requirements tends to have increased complexity.
2. **Measure of performance level, x_2:** A higher level of component performance introduces higher levels of complexity in its components.

3. **Component "size" indicator, $\boldsymbol{x_3}$:** Components that are large in "size" typically indicate higher complexity. However, this should be taken within the correct context, as hardware and software size-complexity correlations do not necessarily relate.
4. **Number of coupled disciplines involved, $\boldsymbol{x_4}$:** If a component involves multiple disciplines, it typically is more complex.
5. **Number of variables and physical processes involved, $\boldsymbol{x_5}$:** An increase in variables and physical processes typically lead to an increase in complexity.
6. **Component reliability measure, x_6:** Components with high reliability typically indicate higher complexity.
7. **Existing knowledge of operating principles, $\boldsymbol{x_7}$:** Existing knowledge about an operating procedure reduces complexity.
8. **Extent of Reuse/heritage indicator, $\boldsymbol{x_8}$:** Reusability of an existing component reduces the complexity of that component.

Building off the weighted, rank measure developed by Bearden (Bearden, 2000, 2004), the component characteristic vector, X, is combined with Bearden's procedure for computing ranks. This combination creates a new vector of component characteristic ranks, $R^{(i)}$, and is defined as follows:

$$R^{(i)} = \left\{ \begin{array}{c} \text{performance tolerance rank} \\ \text{performance level rank} \\ \text{component 'size' rank} \\ \text{coupled disciplines rank} \\ \text{variables involved rank} \\ \text{component reliability rank} \\ \text{existing knowledge rank} \\ \text{extent of reuse/heritage rank} \end{array} \right\} = \left\{ \begin{array}{c} r_1^{(i)} \\ r_2^{(i)} \\ r_3^{(i)} \\ r_4^{(i)} \\ r_5^{(i)} \\ r_6^{(i)} \\ r_7^{(i)} \\ r_8^{(i)} \end{array} \right\}$$

$$r_j^{(i)} \equiv \text{percent rank wrt variable } x_j^{(i)}$$

From this list, weights are assigned to each component characteristic used to compute the component complexity with:

$$w_j^{(i)} = \frac{\text{representative coefficient of } x_j^{(i)}}{\min\left[\text{representative coefficient of } x_j^{(i)}\right]} \tag{9}$$

$$w_j^{(i)} \equiv \text{weight assigned to j}^{\text{th}} \text{ factor for i}^{\text{th}} \text{ component}$$

Using the weight and rank definitions, component complexity can be computed as:

$$\alpha_i = \frac{1}{m}\sum_{j=1}^{m} w_j^{(i)} r_j^{(i)};\ m \equiv \text{number of ranks in vector } R^{(i)} \tag{10}$$

2.6.2.2 Estimating Interface Complexity, C_2

The interface complexity metric, $\beta_{i,j}$, is a function of the component complexities of the interfacing elements, as well as the type of interface (k). This interface complexity can be represented as:

$$\beta_{i,j} = f(c^{(k)}, \alpha_i, \alpha_j) \tag{11}$$

To better apply this metric into the overall complexity metric, the functional equation for interface complexity is defined as follows:

$$\beta_{i,j} = \frac{\max(\alpha_i, \alpha_j)}{c^{(k)}} \tag{12}$$

$c^{(k)} \equiv$ interface type characterization

Estimating the interface type characterization leverages existing data regarding development cost or performance. Interface type characterization can be defined as:

$$c^{(k)} = mean\left[\frac{\max(y_i, y_j)}{z_{i,j}^{(k)}}\right] \tag{13}$$

$y_{i,j} \equiv$ development cost/performance of i^{th} and j^{th} component

$z_{i,j}^{(k)} \equiv$ interface development cost/performance of i^{th} and j^{th} interface type

In the original metric, the suggested primary interface types and suggested subtypes are found in Table 1.

Table 1. Primary Product Interfaces and Subtypes (Sinha, 2014).

Primary Interface Types	**Interface Subtypes**
Physical	Load transfer, translational, spatial, alignment, positional proximity
Flow	Fluid flow, solid flow, mixture flow, plasma flow
Energy	Mechanical, thermal, hydraulic, elastic, pneumatic, electrical, magnetic, electromagnetic, acoustic, chemical, biological, human
Information	Control signal, status signal, information processing

These interface types may not be applicable to all scenarios and act as a suggestion.

Vector, $X_k^{(i,j)}$ is a collection of interface ranks placed in a vector for convenient representation. Like the component complexity's use of existing datasets, the interface complexity leverages a characteristics vector, $X_k^{(i,j)}$, and is suggested as follows (Sinha, 2014):

$$X_k^{(i,j)} = \left\{ \begin{array}{c} \text{magnitude of 'entity' transfer} \\ \text{tolerance requirement indicator} \\ \text{knowledge of interface mechanism} \\ \text{disciplines involved} \\ \text{reliability requirement indicator} \\ \text{extent of reuse/heritage rank} \end{array} \right\} = \begin{Bmatrix} x_1^{(i,j)} \\ x_2^{(i,j)} \\ x_3^{(i,j)} \\ x_4^{(i,j)} \\ x_5^{(i,j)} \\ x_6^{(i,j)} \end{Bmatrix}$$

1. **Magnitude of 'entity' transfer, $\boldsymbol{x_1^{(i,j)}}$:** Interfaces with large 'entity' transfer are typically more complex.
2. **Interface tolerance requirement, $\boldsymbol{x_2^{(i,j)}}$:** Interfaces with tighter tolerance requirements tend to have higher complexity.
3. **Existing knowledge of interface mechanism, $\boldsymbol{x_3^{(i,j)}}$:** Existing knowledge of an interface typically lowers complexity.
4. **Number of disciplines involved, $\boldsymbol{x_4^{(i,j)}}$:** Typically, the more disciplines involved, the higher the complexity.
5. **Interface reliability requirement, $\boldsymbol{x_5^{(i,j)}}$:** Interfaces with high reliability are typically more complex.
6. **Extent of Reuse/heritage indicator, $\boldsymbol{x_6^{(i,j)}}$:** Any extent of reusability reduces complexity.

2.6.2.3 Quantifying Topological Complexity, C_3

As discussed with the use of DSMs, any system with components and connections can be represented graphically (Ulrich 1995, Lindemann et al. 2008). These interactions between components influence a system's behavior and the architectural pattern leads to the inherent structural complexity of that system. A systems architecture is an abstract representation of the compositional entities and their interactions. These interactions are dependent on the constraints and requirements assigned to a system to satisfy. This architecture can be represented in a variety of ways, either via node diagrams, DSMs, or other graphical methods. This architectural pattern is a system characteristic that can be measured.

When contextualizing topology, low topological complexity implies a more centralized scheme while those with high topological complexity imply a decentralized scheme. To meet the modern demands of aerospace systems, a minimum system complexity is needed. An example of this increase in baseline complexity is reflected in the evolution of jet engine architectures (Frey et al. 2007) showing how the rise in demands is met while maintaining performance levels. While an increased topological complexity may help to meet the demands of modern engineered systems, the increased complexity of more decentralized structures may prove more harmful than beneficial.

Translating graph structures to system architectural patterns, the topological complexity metric can indicate the arrangement based on the adjacency matrix energy value. Figure 5 supplements this characteristic of the metric by representing the relationship between differing structural arrangements and their place on the topological spectrum. The topological complexity spectrum is a gradient measure of a system's structure and can represent the system in one of the three distinct categories, as well as in an intermediate/transition classification. In other words, the topological complexity spectrum defines where system's classification generally falls, either centralized, hierarchical, or decentralized. This is due to the numerical nature of topological complexity, as all complexity considerations are taken from the topological value and implemented into structural complexity calculation.

As an organizational structures increases its interdependencies, complexity also increases. Having the ability to determine a large organizations structure through a single metric provides a simple and easy way to trade organizational structures, as well as assist complexity management. Topological complexity (C_3) is an essential characteristic of the OCM, as it is a modifier of the interface complexities (C_2) within the system $[C = (C_1) + (\boldsymbol{C_2})(\boldsymbol{C_3})]$.

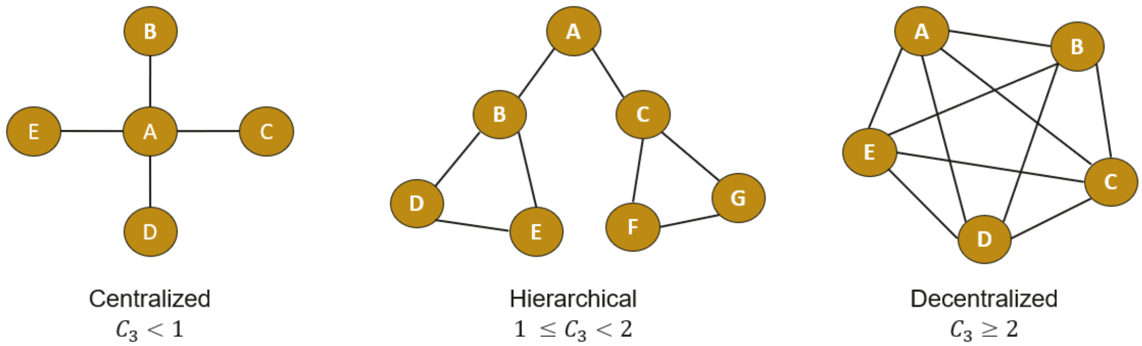


Figure 5. How to Characterize Varying Topological Arrangements

Chapter 3

METHODOLOGY

Complexity theory in an organizational context highlights that organizations must adapt to uncertain environments. The environment both internal and external to the organization plays a role in complexity and prioritizing adaptable structures helps with unpredictability. This is contrasted by processes that are more rigid and structured in mature and well-established organizations. Since complexity management and organizational success go hand in hand, it is important to find tools that improve, augment, and/or supplement an organization's structure, topology, and reporting relationships.

Before developing the base ranks for the structural complexity metric in organizations, it is important to look at how certain aspects of organizations affect complexity. Of course, similarities can be drawn between the complexity realizations found in product architectures and organizational architectures; however, discussion is still necessary, as organizational architectures involve their own unique challenges.

Some of the translatable characteristics from Sinha's product architecture metric include a system's size, modularity, and reliability. While not a direct comparison to products, organizational elements share these properties at varying semblances. For example, the 'size' of an architectural element is essentially identical between products and organizations, while quantifying the reliability of a product component and an individual pose their own challenges but have the same effect on complexity. These similarities will be discussed case-by-case in the following section.

In terms of uniquely organizational considerations, component support and interfacing can be built off the benefits of topological arrangement. While empowerment and adaptability were addressed with topology, an organization's structure needs other processes and measures to enact these characteristics at a component and interface level. In other words, arranging an organization to be adaptable opens the possibility for an empowering environment and to effectively measure its complexity, key aspects should be captured in an analytical ranking system.

When discussing the internal or component processes and characteristics that promote productivity and effectiveness, having a proper support system is essential. Ways of quantifying such support can be found with assessing available tools, training and career progression aid, implementation of incentives, and the level of understanding shared within an organization (Hoopes and Postrel 1999; Browning et al. 2006). As for the external interactions, organizational interface mechanisms provide valuable means for quantifying intercommunication effectiveness. Assessing the overall visibility of an organization's project goal, the effectiveness of coordination, boundary objects, and common processes, as well as the use of interface mediators (Star and Griesemer 1989; Bernstein 2001; Steward 2000; Browning et al. 2006) provide the flexibility necessary to manage system complexity. As no two complex systems are identical, having the ability to adapt the organizational complexity metric (OCM) to differing datasets and techniques is essential. In addition to the need for strong datasets as the foundation for the OCM, the OCM should reflect the iterative nature of the organizations it is modelling. As an organization's structure forms and changes through the conceptualization, production, and operational phases of the lifecycle, the OCM should reflect these changes. The OCM was designed to model a structural snapshot of an organization and should be iterated on. The frequency of this iteration is at the will of the systems engineer and should be considered on a team to team, project to project, and organization to organization basis.

3.1 Visualizing Structural Complexity in Organizations

Building off Sinha's original metric and the implementation of nested numerical DSMs, a method for visualizing complexity within organizations is proposed. Figure 6 represents the flowchart of the proposed OCM. Beginning with an allocation of available human resources, a systems engineer and/or program manager will construct an organizational arrangement. Within this structure, topology is inherently defined. Using the suggested method for data collection or organization-specific data definition, interface and human resource information is stored in a convenient, singular organizational DSM. From this singular location, the structural complexity of the organization can now be analyzed. At this stage, the OCM is executed on the nested numerical DSM. The output from this process will be a metric to measure the current structural complexity based on the chosen hierarchy, available data, and interface definition. From here, the output metric can be used to trade between differing organizational arrangements with the same organizational dataset or the implementation of Sinha's "complexity pool" can be used to optimize an organizations arrangement based on a pre-allocated complexity value.

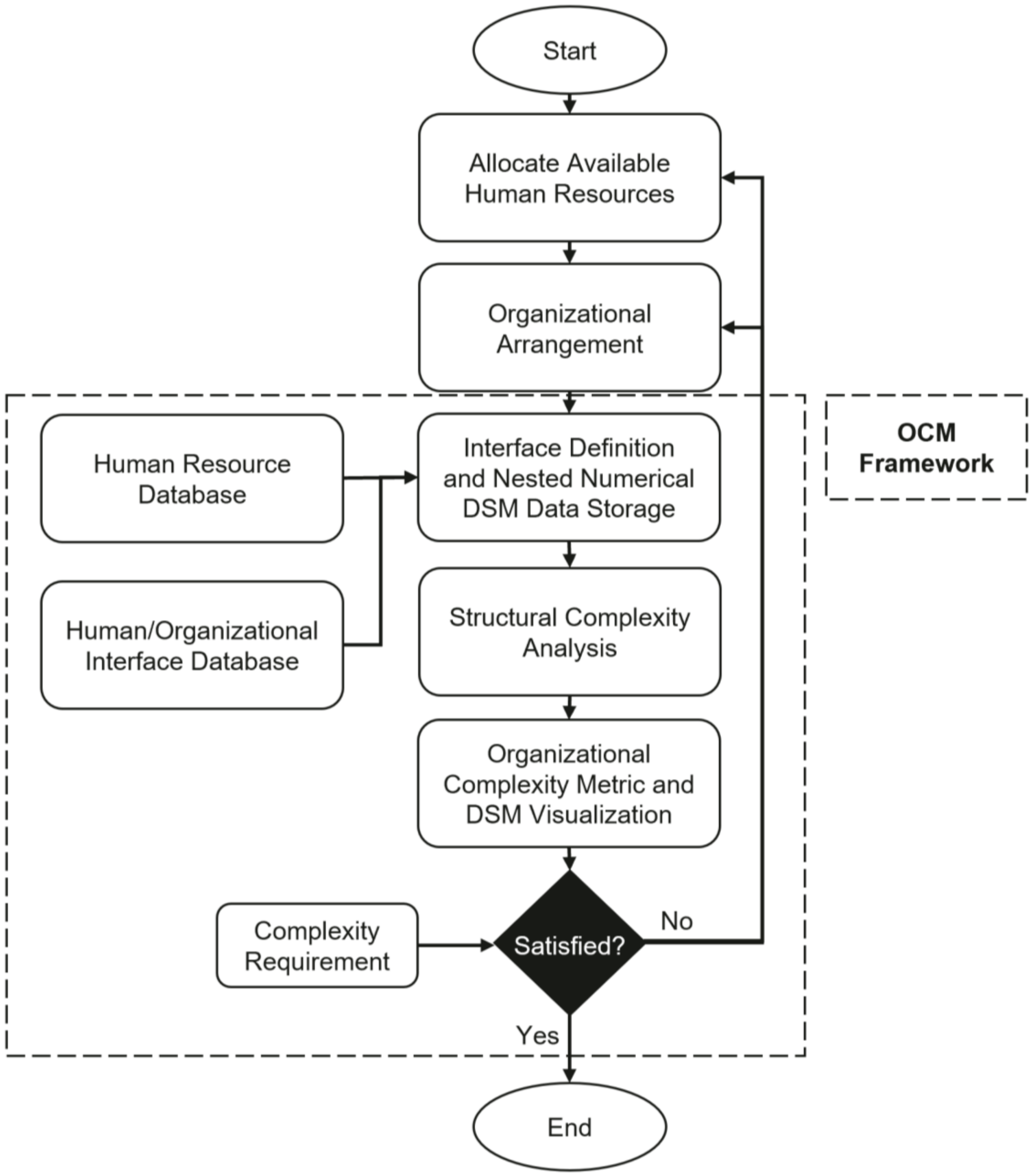


Figure 6. Methodology for Visualizing Structural Complexity within an Organization

3.2 Relating Organizational Architectures to the Product Architecture Metric

To translate a product architecture structural complexity metric to organizational architectures, the three complexity categories must be altered to conform to the unique challenges associated with organizational architectures. System topology translates directly, as it is an inherent property of the systems adjacency matrix and is independent of the system observer. Additionally, topology can be directly related to organizational hierarchies and development team structures. Component and interface complexity do not transfer as easily, and characterization should be evaluated from system to system. As expected, attempting to characterize individuals is nearly impossible to do at the most detailed level. From this lack of full definition, a single methodology for defining individual complexity cannot be achieved. Data collection and/or performance measurements must be tailored to meet the desired scope of complexity for a given system's complexity analysis. As for the interface complexity, unless there are rigid organizational procedures that make the flow of information easy to define and trace, a series of interface mechanisms will be introduced to combat inherent unpredictability. From these mechanisms, a systems engineer can appropriately characterize communication interfaces from project to project. As a result, the complexity metric is not meant for comparing organizations with differing methodologies, rather the OCM is meant for trading different arrangements within a project.

In this section, each complexity metric will be discussed in an organizational context. This modification of Sinha's structural complexity metric is presented as a metric for organizational complexity quantification and is meant to guide complexity management and design exploration. For individual and interface complexity, some solutions for defining complexity are presented, but again, should be evaluated from system to system.

3.3 Introduction to Organizational Complexity Metric Example

To better communicate the OCM methodology, a hypothetical example was developed. Within the example, each aspect of the structural complexity metric will be calculated and contextualized. This example will consist of five individuals apart of ‘Team Y’ within ‘Organization Z’ and is structured as seen in Figure 7.

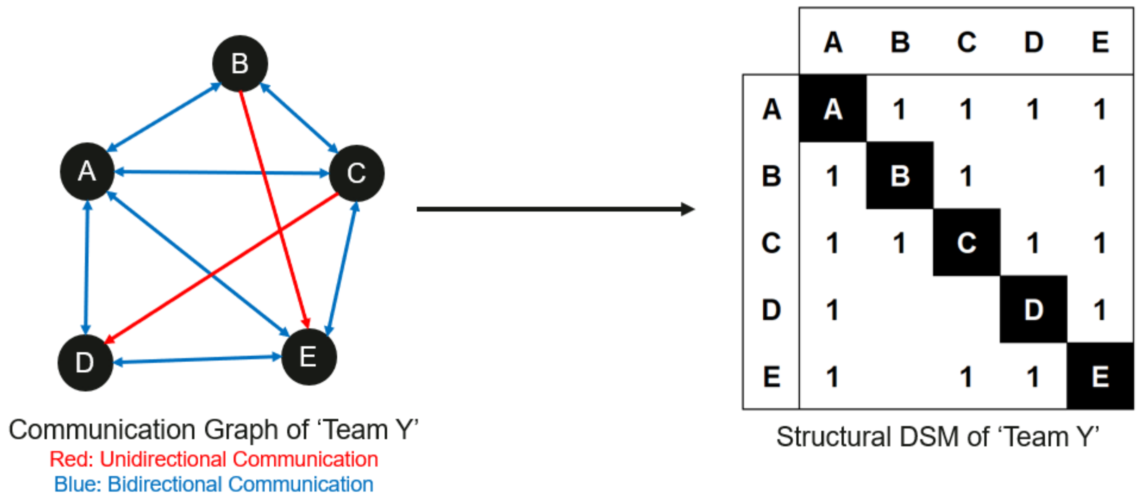


Figure 7. Organizational Structure of Team ‘Y’

Team Y is involved in an internal, interdisciplinary engineering project. Section 3.5 will walk through defining component complexity through created datasets represented by six ‘rank’ categories:

$$R^{(i)} = \begin{Bmatrix} \text{communication pathways rank} \\ \text{individual performance rank} \\ \text{coupled disciplines rank} \\ \text{organizational processes rank} \\ \text{indiviudal reliability rank} \\ \text{indiviudal experience rank} \end{Bmatrix} = \begin{Bmatrix} \begin{pmatrix} r_1^{(i)} \\ r_2^{(i)} \\ r_3^{(i)} \\ r_4^{(i)} \\ r_5^{(i)} \\ r_6^{(i)} \end{pmatrix} \end{Bmatrix}$$

A similar process will be conducted for interface definition and is explored in Section 3.6. Interface ranks are defined as such:

$$X_k^{(i,j)} = \left\{ \begin{array}{c} \text{information of critical importance} \\ \text{deadline tolerance/leadtime sensitivity} \\ \text{interorganizational interfacing} \\ \text{interdisciplinary communications} \\ \text{project goal commonality} \end{array} \right\} = \left\{ \begin{pmatrix} x_1^{(i,j)} \\ x_2^{(i,j)} \\ x_3^{(i,j)} \\ x_4^{(i,j)} \\ x_5^{(i,j)} \end{pmatrix} \right\}$$

These ranks represent data that Organization Z finds valuable and essential to approximating system complexity. This collection of characterized component properties can be presented as either a vector or, for the sake of the OCM, distributed throughout the nested numerical DSM. An example of how these ranks is presented within a nested numerical DSM can be found in Figure 8 and Figure 9. Figure 8 shows a snapshot of an organizational DSM and focuses on the interactions between individual ‘A’, individual ‘B’, and individual ‘C’. Within the white and gray squares (or interfaces) are the interface characteristic ranks, while the black squares represent the internal component/individual characteristic ranks. To better communicate each individual rank throughout the example, each rank is presented in a singular DSM that only contains the data relevant to that rank.

	A	B	C
A	Individual A	A-to-B Interface	A-to-C Interface
B	B-to-A Interface	Individual B	B-to-C Interface
C	C-to-A Interface	C-to-B Interface	Individual C

...

⋮

Figure 8. Individual and Interface Ranks Within a Nested Numerical DSM (Low-Fidelity)

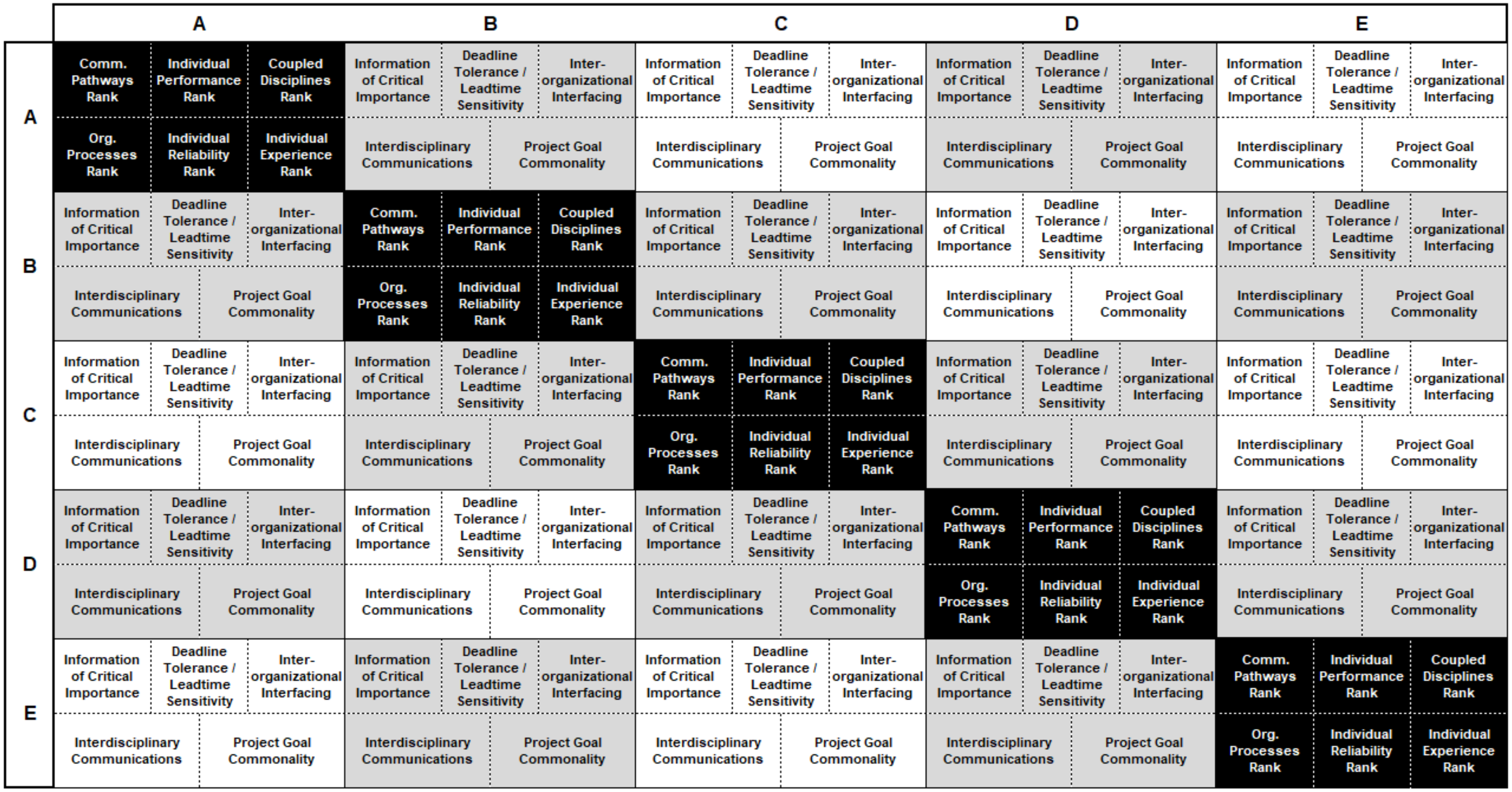


Figure 9. Individual and Interface Ranks Within a Nested Numerical DSM (High Fidelity)

3.4 Estimating individual Complexity in Organizational Architectures

In this section, the proposed example component complexity ranks will be expanded upon and applied to Team Y. These component ranks are based on fictional, existing datasets belonging to Organization Z. The six datasets are then evaluated and converted to a complexity metric to be used to aid complexity management. The six ranks are discussed as such:

1. *Individual 'Communication Pathways' Rank*

When considering a component's system influence in an organizational context, it is important to note the communication load an individual is experiencing. The easiest way to quantify this reach and influence is the number of communication pathways (such as the use of communication tools, engineering software tools, team meetings, et cetera between the same two or more individuals) an individual has responsibility for. While it appears that this characteristic could be associated with the interface(s) of the component, that is not necessarily true. The number of pathways does not relate to the interface, directly, rather, the number of pathways associated with an individual reflects on the complexity of the processes assigned to the individual. An individual with many communication pathways has a higher organizational complexity requirement. This can be contributed to a potential loss in information within these pathways and an increase in pathways compounds this complexity.

Figure 10 shows Team Y and its individuals assigned with the number of communication pathways. For simplicity, each component only has the capability of having a maximum of two pathways between other components. An empty cell denotes a nonexistent pathway, while a '1' or '2' denote the number of communication pathways existing. For example, individual A has 2 outgoing pathways to individual B, as well as 2 incoming pathways from the same individual.

	A	B	C	D	E
A	A	2	1	1	1
B	2	B	2		
C	1	2	C		1
D	1			D	
E	2		2	1	E

Figure 10. Hypothetical Organizational DSM (Communication Pathway Definition)

For the first rank, the method for converting the available data to its contribution to complexity is shown as such:

Step 1: Define minimum and maximum 'size' values.

A: $\boldsymbol{sum(i,:) + sum(:,j) = 11}$ **(maximum)**

B: $sum(i,:) + sum(:,j) = 8$

C: $sum(i,:) + sum(:,j) = 9$

D: $\boldsymbol{sum(i,:) + sum(:,j) = 3}$ **(minimum)**

E: $sum(i,:) + sum(:,j) = 7$

Step 2: Use Bearden's system to rank each datapoint.

$$f_A = \frac{4}{4} = 1.0$$

$$f_B = \frac{2}{4} = 0.5$$

$$f_C = \frac{3}{4} = 0.75$$

$$f_D = \frac{0}{4} = 0.0$$

$$f_E = \frac{1}{4} = 0.25$$

Step 3 and 4: Create aggregated complexity contribution and scale to [0, 5] to establish numeric contribution to complexity.

Employing Eq. (1), the complexity contribution is as follows:

Table 2. Communication Pathways Ranks and Complexity Contribution

Individual	**Communication Pathways**	**Rank (f_j)**	**Complexity Contribution**
A	11	1.0	5.0
B	8	0.5	2.5
C	9	0.75	3.75
D	3	0.0	0.0
E	7	0.25	1.25

This scaling factor of 5 is carried over from Sinha's complexity calculations, as it is the suggested value to better integrate with the other aspects of the structural complexity metric. As a result, the contribution to complexity of each rank is five times the percent rank. Additionally, scaling the ranks in accordance with weight could be employed here, but is not present in the example.

2. Individual Performance Rank

An obvious contributor to system performance, efficiency, and complexity is the performance of the system constituents, themselves. While product architectures have requirements, tolerances, and other metrics to help align subsystem performance with desired system performance, managers use similar benchmarks to measure employees. For example, Apple Inc. uses a simple performance measuring strategy. Each employee is assessed within three categories: teamwork, innovation, and results. Within these categories, an employee is assigned a performance level of either “needs improvement”, “met expectations”, or “exceeded expectations” (Anonymous, 2015). For the sake of the existing example, each of the five individuals was assessed using Apple’s 3-tier assessment. With a decrease in performance contributing to an increase in unpredictability, risk, and potential for mistakes, complexity scales inversely to performance. Therefore, an employee who “needs improvement” will garner a ‘3’, while an “exceeds expectation” grants a ‘1’. An equally weighted average was taken between the three categories. Table 3 shows the results of the assessment and how they can be stored within a DSM in Figure 11. Figure 11 shows the ability of displaying different data information based on how deep in the nested DSM the observer is located. In other words, the average performance assessment score is displayed at the surface level, but the more comprehensive information is located within the averaged cell (black). The results of the complexity contribution are found in Table 4.

Table 3. Individual Performance Assessment Results

Individual	**Teamwork**	**Innovation**	**Results**
A	1	1	2
B	2	1	3
C	2	2	1
D	1	3	3
E	1	1	1

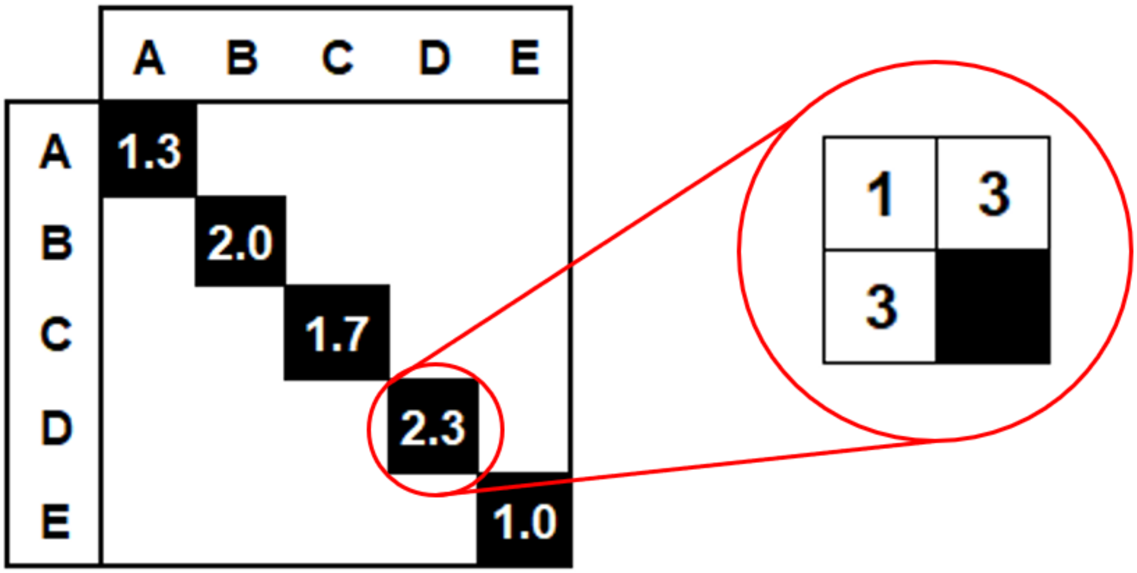


Figure 11. Individual Performance DSM Data Storage

Table 4. Individual Performance Ranks and Complexity Contribution

Individual	Value	Rank	Complexity Contribution
A	1.3	0.25	1.25
B	2.0	0.75	3.75
C	1.7	0.5	2.5
D	2.3	1.0	5
E	1.0	0.0	0

3. Coupled Disciplines Rank

The communication between differing disciplines parallels that of product architecture. The more disciplines involved, as well as the subject matter gap between disciplines implies an increase in complexity. The identification of coupled interdisciplinary resources could lead to the implementation of integration specialists or other aids. Individuals within a team could face interdisciplinary communication both internally and externally. Characterizing individual roles and classifications defines the interdisciplinary boundary, as a result. This rank is directly related to the preparation needed to communicate between interdisciplinary boundaries and remains local in terms of complexity calculation. For implementation into the ranking system developed by Bearden and implemented by Sinha, the initial definition of an interdisciplinary boundary would be binary with bounds [0, 1]. If an interdisciplinary boundary exists, the rank associated would be a value of 1 and a nonexistent boundary would equate to a 0. The coupled disciplines rank would similarly reflect a system's adjacency matrix and is symmetric; however, certain relationships will be excluded, since not all relationships involve interdisciplinary communication. Additionally, this metric could be expanded up to include the number of interdisciplinary relationships, strength of the relationship, et cetera.

Figure 12 represents these interdisciplinary relationships within the hypothetical system. The results of the ranking and associated complexity contribution can be seen in Table 5.

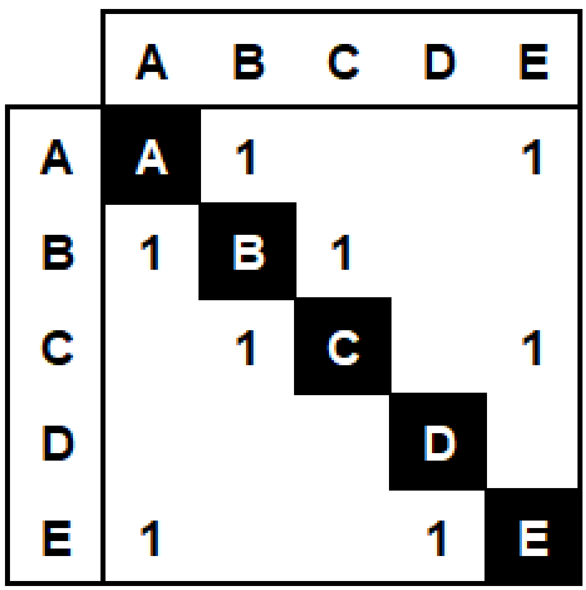


Figure 12. Hypothetical Organizational DSM (Interdisciplinary Couples)

Table 5. Interdisciplinary Couples Ranks and Complexity Contribution

Individual	Value	Rank	Complexity Contribution
A	4	0.5	2.5
B	4	0.5	2.5
C	3	0.25	1.25
D	1	0.0	0.0
E	4	0.5	2.5

4. Organizational Processes Rank

One can equate the number of processes to organizational processes, checkpoints, et cetera to an organization’s complexity. Many organizational processes vary from position to position along with scope of work and the responsibilities associated with an individual. In a broader context, the flow of information within an individual’s scope of work may differ based on how that information is being transferred. For example, information that is being prepared to travel upwards within a reporting relationship may have to undergo a different set of processes, as compared to information transferred between a horizontal interaction network. A direct example of this could be the idea of decision gates (INCOSE, 2015). Although a team’s collaborative contribution develops the information to be discussed at a decision gate, there are differing processes needed to prepare the information. For this, someone higher in the organizational hierarchy may have more interconnected processes and challenges than a technical engineer or other component of an organization. These challenges and intricacies contribute to the potential for emergent behaviors to appear within the organizational system, resulting in an increase in complexity. A parallel example can be drawn between an increase in intricateness within software processes and an increase in complexity (Banker, 1993).

Applying this to Team Y, a hypothetical numerical system is developed to determine the difficulty associated with of information transfer in the example organizational architecture. This scale ranges from 0 to 3 with an increasing number denoting a higher difficulty associated with the processes involved in the transfer of information. This difficulty can be associated with the potential of information loss or risk due to the processes the information must undergo. An increase in difficulty can be associated with an increase in complexity, as miscommunications are more likely to occur. Figure 13 represents a numerical DSM of the five components and the respective process difficulty. The results of the ranking and associated complexity contribution can be seen in Table 6.

	A	B	C	D	E
A	A	3	3	3	3
B	3	B	1		1
C	3	2	C	1	1
D	3			D	1
E	3		2	1	E

Figure 13. Hypothetical Organizational DSM (Process Definition)

Table 6. Organizational Processes Ranks and Complexity Contribution

Individual	Value	Rank	Complexity Contribution
A	24	1	5
B	10	0.25	1.25
C	13	0.75	3.75
D	9	0	0
E	12	0.5	2.5

5. Individual Reliability Rank

Miscommunication and information loss can be contributed to structural arrangements and processes; however, the responsibility of properly interpreting information and communicating it clearly ultimately falls on the individual. With the increase in system demand, those developing the system are demanded more, as well. Without proper management of information by the individual, these demands can become overwhelming and lead to information being buried. Individual performance within an organization is dynamic and difficult to quantify. Reliability is often viewed narrowly and do not include a wide breadth of antisocial metrics that indicate potential unreliable behavior. Reliability changes from day-to-day and is determined by an individual's engagement in a process or project. Although dynamic, ways of measuring this reliability can be found with metrics, such as, read rates, communication reach, et cetera.

For Team Y, email read rates were assessed and converted into complexity contributions. Figure 14 represents the read rates within a DSM while Table 7 represents the resulting ranks and complexity contributions. The higher the response rate contributes to a lower complexity contribution and vice versa.

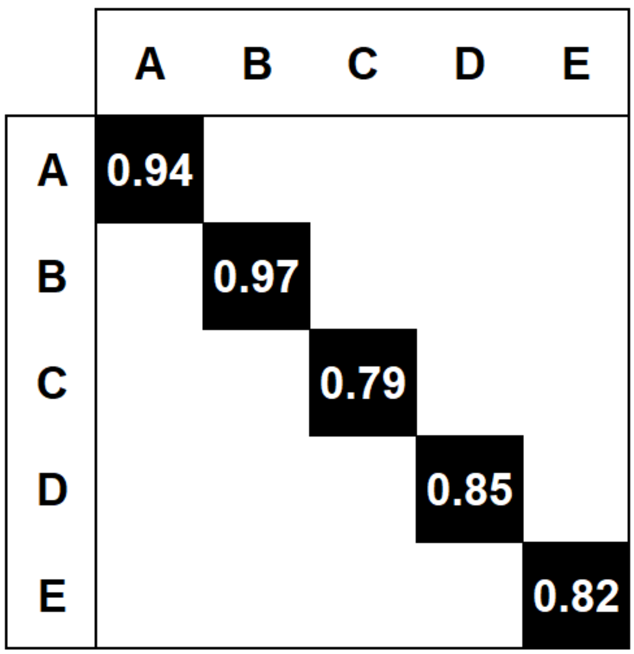


Figure 14. Individual Read Rates

Table 7. Individual Reliability Rank and Complexity Contribution

Individual	Value	Rank	Complexity Contribution
A	0.94	0.25	1.25
B	0.97	0	0.0
C	0.79	1	5.0
D	0.85	0.50	2.5
E	0.82	0.75	3.75

6. Existing Knowledge Rank

Existing knowledge of operating procedures refers to the organizations preexisting experience with organizational procedures. In an organizational context, specific years of experience can be substituted for a numeric representation of prior knowledge. An individual with more years of experience with a process implies a more predictable working environment. In terms of communication, having an overall greater scope of understanding reduces some of the communication challenges someone without that perspective may face. This would lead to a decrease in complexity.

Applying this to Team Y, each of the five individuals have varying experience with the project. Figure 15 represents the numerical DSM storing the years of experience associated with similar interdisciplinary projects. Results of the rankings can be found in Table 8.

	A	B	C	D	E
A	25				
B		17			
C			13		
D				1	
E					7

Figure 15. Years of Experience with Processes Associated with the Project

Table 8. Years of Experience Ranks and Complexity Contribution

Individual	Value	Rank	Complexity Contribution
A	25	1.0	5
B	17	0.75	3.75
C	13	0.5	2.5
D	1	0.0	0.0
E	7	0.25	1.25

Following the individual complexity quantification, each complexity characterization can be tabulated and stored in a nested numerical DSM for ease of representation. Table 9 represents the tabulated results of the individual complexity study of Team Y. These results are then stored in the organizational complexity metric DSM shown in Figure 16.

Table 9. Summary of Component Complexities

	$C_{1,pathways}$	$C_{1,perf}$	$C_{1,coup}$	$C_{1,OP}$	$C_{1,reliability}$	$C_{1,exp}$	$C_{1,i}$
A	5.0	1.25	2.5	5	1.25	5	**20.0**
B	2.5	3.75	2.5	1.25	0.0	3.75	**13.75**
C	3.75	2.5	1.25	3.75	5.0	2.5	**18.75**
D	0.0	5	0.0	0.0	2.5	0.0	**7.5**
E	1.25	0	2.5	2.5	3.75	1.25	**11.25**

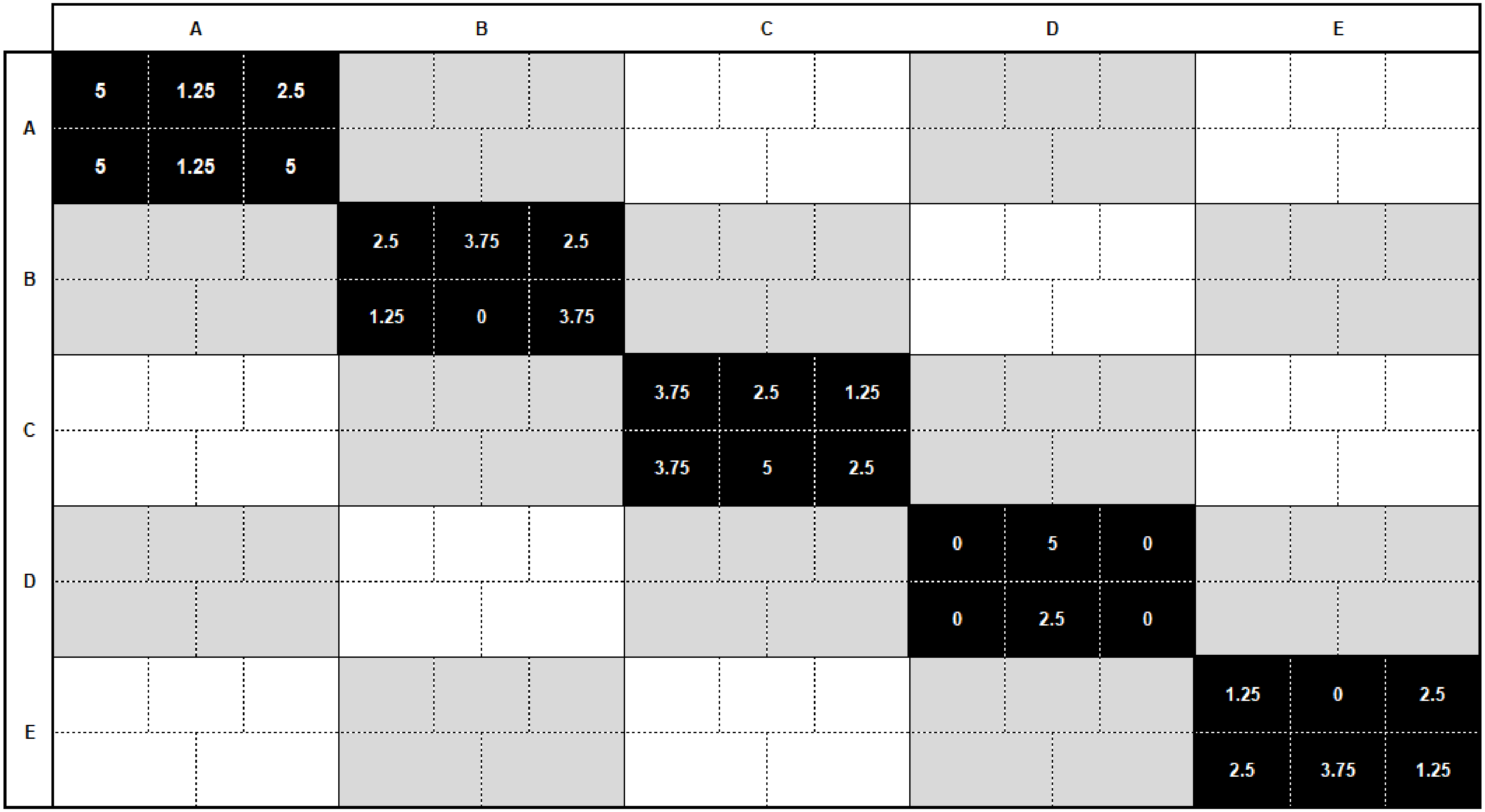


Figure 16. Individual Complexity Contributions of Team Y

3.5 Estimating Interface Complexity in Organizational Architectures

The main challenge with managing interpersonal and organizational interfaces is the lack of true consistency throughout a system. Modern aerospace systems are developed by large groups of interdisciplinary teams, organizational departments, and companies collaborating with one another. This variety brings the potential for conflicting interests, project goals, and communication styles. Sinha's original methodology for quantifying interface complexity within a product architecture was to use cost and performance characteristics of typical interfaces normalized with the cost and performance metric of the components the interface was 'connecting'. Revisiting Eq. (13), the methodology for quantifying organizational interfaces can be translated using the same performance comparison. Much like the original metric, interface definition is highly dependent on available data and/or the employment of expert opinions. Much like the individual ranking system, unique organizational interface characteristics are suggested as such:

1. *Information of Critical Importance,* $x_1^{(i,j)}$

Defining the information that is flowing between individuals through the interfaces is essential for determining the complexity of the organizational processes involved. Information that is more critical to project success tends to have greater security measure and involve additional processes. These additional steps lead to an increase in complexity that needs to be managed.

2. *Deadline tolerance / Leadtime sensitivity,* $x_2^{(i,j)}$

Certain information is more sensitive to deadlines and require close monitoring to avoid delays and overrun. Information that is more sensitive to these deadlines tend to increase the complexity within an organizational structure.

3. Interorganizational Interfacing, $x_3^{(i,j)}$

Alluded to previously, information that crosses department or organization boundaries tend to require additional preparations or security considerations. Additionally, miscommunication and information loss are more likely when crossing an organizational boundary. Therefore, information crossing these boundaries tend to increase the complexity of the system.

4. Interdisciplinary Communications, $x_4^{(i,j)}$

Much like product interfaces, organizational interfaces that involve more than one discipline involved in the interaction tend to increase the complexity of the system.

5. Project Goal Commonality, $x_5^{(i,j)}$

Project goals and requirements need to be aligned to ensure proper communication and understanding exists between the interfaces. Visibility and understanding of these goals can become clouded or obscured both internally and externally to the organization. The more indifferences between two individuals leads to an increase in an organization's complexity.

Continuing with the application of the OCM to Team Y, the different interface types were defined between the five individuals. Table 10 represents the interface characterizations, as well as important performance metrics used to quantify the interfaces. All interface types are present within Team Y; however, as they are working intra-organizationally, no complexity due to interorganizational communication is present. After calculating the interface complexity, Figure 17 represents the interface definition DSM and their corresponding complexity contributions.

Table 10. Example Interface Definition

Interface Type	$\boldsymbol{c^{(k)}}$
Information of Critical Importance	0.67
Leadtime Sensitivity	2.14
Interdisciplinary Communication	6.2
Interorganizational Communication	4.42
Project Goal Commonality	1.22

These complexity contributions are calculated by taking the maximum individual complexity within the row and column and dividing it by the interface that is being analyzed. A sample calculation for the interface of 'Information of Critical Importance' between Individual 'A' going to Individual 'B':

$$C_{2,(i,j)} = \frac{\max\left(C_{1,i}, C_{1,j}\right)}{c^{(k)}}$$

$$C_{2(A,B)} = \frac{\max(20.0,\ 13.75)}{0.67}$$

$$\boldsymbol{C_{2(A,B)} = 29.85}$$

Using the hypothetical interface characteristics and their quantities, each interface's complexity contribution is calculated and stored in the nested numerical DSM found in Figure 17.

<table>
<tr><th></th><th colspan="6">A</th><th colspan="6">B</th><th colspan="6">C</th><th colspan="6">D</th><th colspan="6">E</th></tr>
<tr><th rowspan="2">A</th><td colspan="6" rowspan="2">$C_{1,A} = 20.0$</td><td colspan="2">30</td><td colspan="2">9.3</td><td colspan="2">3.2</td><td colspan="2">30</td><td colspan="2"></td><td colspan="2"></td><td colspan="2">30</td><td colspan="2"></td><td colspan="2"></td><td colspan="2">30</td><td colspan="2"></td><td colspan="2">3.2</td></tr>
<tr><td colspan="3">N/A</td><td colspan="3">16.4</td><td colspan="3">N/A</td><td colspan="3">16.4</td><td colspan="3">N/A</td><td colspan="3">16.4</td><td colspan="3">N/A</td><td colspan="3">16.4</td></tr>
<tr><th rowspan="2">B</th><td colspan="2">30</td><td colspan="2">9.3</td><td colspan="2">3.2</td><td colspan="6" rowspan="2">$C_{1,B} = 13.75$</td><td colspan="2">28</td><td colspan="2">8.8</td><td colspan="2">3.0</td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2">6.43</td><td colspan="2"></td></tr>
<tr><td colspan="3">N/A</td><td colspan="3">16.4</td><td colspan="3">N/A</td><td colspan="3">15.4</td><td colspan="3">N/A</td><td colspan="3">11.3</td><td colspan="3">N/A</td><td colspan="3">11.3</td></tr>
<tr><th rowspan="2">C</th><td colspan="2">30</td><td colspan="2">9.3</td><td colspan="2"></td><td colspan="2">28</td><td colspan="2">8.8</td><td colspan="2">3.0</td><td colspan="6" rowspan="2">$C_{1,C} = 18.75$</td><td colspan="2"></td><td colspan="2">8.8</td><td colspan="2">3.0</td><td colspan="2">28</td><td colspan="2"></td><td colspan="2">3.0</td></tr>
<tr><td colspan="3">N/A</td><td colspan="3">16.4</td><td colspan="3">N/A</td><td colspan="3">15.4</td><td colspan="3">N/A</td><td colspan="3">15.4</td><td colspan="3">N/A</td><td colspan="3">15.4</td></tr>
<tr><th rowspan="2">D</th><td colspan="2">30</td><td colspan="2">9.3</td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="6" rowspan="2">$C_{1,D} = 7.5$</td><td colspan="2"></td><td colspan="2"></td><td colspan="2">1.8</td></tr>
<tr><td colspan="3">N/A</td><td colspan="3">16.4</td><td colspan="3">N/A</td><td colspan="3">11.3</td><td colspan="3">N/A</td><td colspan="3">15.4</td><td colspan="3">N/A</td><td colspan="3">9.2</td></tr>
<tr><th rowspan="2">E</th><td colspan="2">30</td><td colspan="2"></td><td colspan="2">3.2</td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2">28</td><td colspan="2">8.8</td><td colspan="2"></td><td colspan="2">17</td><td colspan="2"></td><td colspan="2">1.8</td><td colspan="6" rowspan="2">$C_{1,E} = 11.25$</td></tr>
<tr><td colspan="3">N/A</td><td colspan="3">16.4</td><td colspan="3">N/A</td><td colspan="3">11.3</td><td colspan="3">N/A</td><td colspan="3">15.4</td><td colspan="3">N/A</td><td colspan="3">9.2</td></tr>
</table>

Figure 17. Interface Complexity Contributions of Team Y

3.6 Quantifying Topological Complexity in Organizational Architectures

Aside from the metric application, organizational topology can provide more insight about the structure of the organization, the empowerment of its individuals, as well as how adaptable the structure is. While not the only organizational characteristics affected by the topological structure of the organization, these characteristics have shown benefits to organizational success and efficiency.

When discussing individual empowerment, this refers to giving individuals the ability to make decisions related to their tasks. This could include personal and collective decision-making, access to information for decision-making, and promoting engagement, education, and the exchange of information (Adler, 1993; Nonaka, 2007). A common strategy for promoting empowerment within an organization is through the arrangement of the organization, itself. By shifting traditional hierarchical structures towards a more decentralized arrangement, individuals are granted the ability to collaborate and communicate on a level interaction network. An additional benefit of decentralization is an organizations ability to adapt. With modern markets and technology evolving rapidly, uncertainty arises leading to the need for adaptable organizations. Again, decentralization is a suggested strategy within organizational structures (Foss, 2003; Heckscher and Donnellon, 1994; Ouchi, 1980; Torbert, 1974; Volberda, 1996) and its effect on the flexibility of the structure. As hypothesized, this adaptability promotes organizational effectiveness in everchanging environments (Birkinshaw, Hamel, & Mol, 2008; Foss, 2003; Zenger and Hesterly, 1997).

While complete flexibility and empowerment are not the only properties of successful organizations, both are important characteristics to manage and analyze. However, the idea of decentralization brings more complexity into a system. Much like other aspects of structural complexity, this must also be balanced to maximize effectiveness. Using topological complexity as identifier within the structural complexity, these two characteristics become easier to visualize. Figure 18 relates differing organizational arrangements and their effects on complexity. With an increase in decentralization comes an increase in topological complexity. Although topological complexity translates directly from Sinha's original metric (as it is an inherent characteristic of a system's adjacency matrix), the organizational implications discussed above provide new insights that were not considered in the original product architecture application.

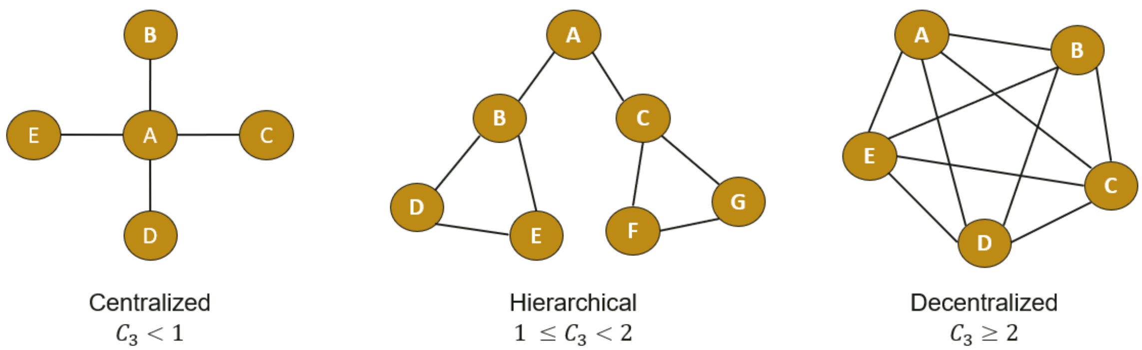


Figure 18. Structural Arrangement and Complexity Relationship

Applying the topological complexity metric to Team Y, the matrix energy of the organization's adjacency matrix was calculated and divided by the number of individuals.

3.7 Compiling the Organizational Complexity Metric of the Example Organization

With all the complexity components accounted for, Table 11 shows the summary of the completed structural complexity metric.

Table 11. Summary of Example Complexity Calculations

Individual	C_1	$\sum C_2$	C_3
A	20.0	**761.9**	**1.46**
B	13.75		
C	18.75		
D	7.5		
E	11.25		
Organization	**71.25**		
Structural Complexity		**1184**	

Reviewing the results shown in Table 11 provides an additional tool for trading and organizing team structures and human resources. Taking aspects from networking diagrams and existing organization breakdown structures, the complexity metric and its constituent components provides a simple, numerical visualization of the traded organization. Looking at the component complexities (C_1) attributed to the individuals within the organization allows for a snapshot of the complexity distribution throughout the organization. This allows for potential rework or additional resource allocation considerations to support more 'complex' areas of an organization. As for the interface complexities (C_2) and topological complexity (C_3), these metric are representative of the organization's arrangement and how much influence this arrangement has on the chosen cost/performance metrics that define them. With a 1.46 value for topological complexity, this indicates a more traditional hierarchical organizational structure ($1 \leq C_3 < 2$). This indicates a stronger influence of interfacing on the complexity metric. This information could lead to the implementation of additional interface mechanisms and processes to relieve the challenges associated within complex interface and communication challenges.

Applying this tabulation of organizational complexity into a DSM, two DSMs with differing fidelity are available to aid complexity analysis and management. Figure 19 represents the overall complexities of each individual and their interfaces, while Figure 20 represents a breakdown of each complexity contributor and characterization. In other words, Figure 19 show the sum of the more detailed Figure 20. To strengthen the visual nature of the DSM, a color gradient is applied to the DSM to highlight areas of high complexity within the organization. Within both DSMs, outliers are highlighted in red color tones, while lower complexity interfaces are colored in the green scale. This color scale is self-referential and does not indicate that an interface of green is not highly complex, rather, comparatively, it is less complex than other aspects of the organization. This DSM is meant to bring attention to extreme cases, allowing for the systems engineer or program manager to reallocate support and resources to these areas of complexity.

	A	B	C	D	E
A	20.0	58.8	46.2	46.2	49.5
B	58.8	13.8	55.1	11.3	17.7
C	55.6	55.1	18.8	27.2	46.4
D	55.6	11.3	15.4	7.5	11.0
E	49.5	11.3	52.1	27.8	11.3

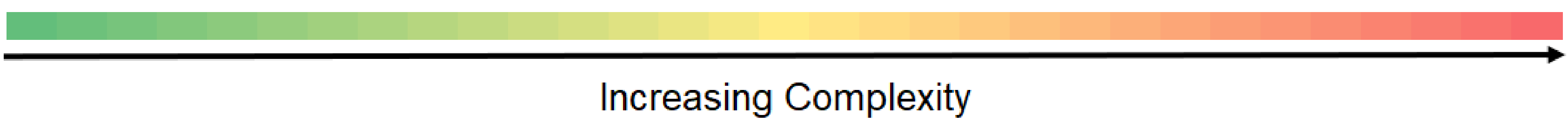


Figure 19. Visualizing the Relative Complexity Density of Team Y (Low-Fidelity)

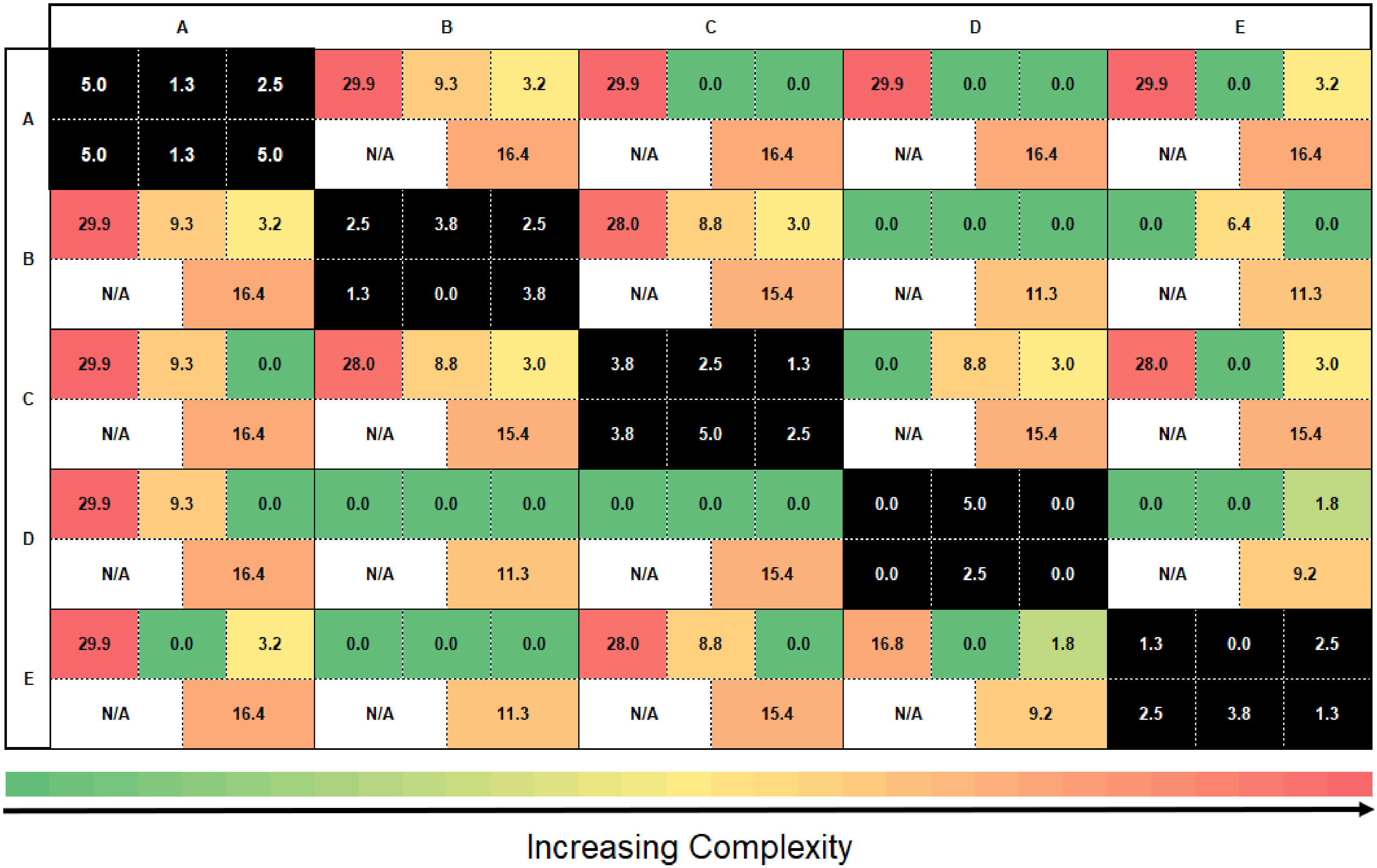


Figure 20. Visualizing the Relative Complexity Density of Team Y (High-Fidelity)

Chapter 4

ORGANIZATIONAL COMPLEXITY METRIC APPLIED TO NASA ISS

In this section, an existing organizational structure will be explored and expanded upon using the proposed OCM methodology. The organizational architecture is in the desired numerical DSM format with data relatable to the proposed organizational characterizations. However, due to the lack of full definition in terms of available organization analytics, the implementation of the structural complexity metric is limited to the small dataset. The relevance of each dataset and how it relates to an organization's complexity will be discussed and interpreted to fit the proposed methodology for quantifying structural complexity. The addition of the complexity metric within the context of the system is meant to supplement system representation and the already existing numerical DSM format. This additional example will provide insight on how the methodology scales with an organization of different size, arrangement, and data collection methodology.

4.1 Program Overview - NASA ISS Sustaining Engineering Example

The International Space Station (ISS) began construction in 1998 and has been continuously inhabited with human operators since November 2000. As a result of the ISS's continued success, NASA began planning necessary sustaining operations needed to maintain the station. In 2003, Tim Brady was assigned the task of developing a list of the necessary engineering sustainment tasks needed to achieve NASA's goal.

4.1.1 Data Collection - NASA ISS Sustaining Engineering Example

Over the course of 4 months, Brady reviewed ISS documentation and interviewed current and former ISS engineers to develop a list of tasks based on necessary skills needed to maintain the ISS. The following DSMs represent organizational responsibilities, information sharing, knowledge capture, and the interactions between teams that support the ISS.

4.1.2 Organizational Model - NASA ISS Sustaining Engineering Example

Thirty-six critical functions performed by various teams within the ISS organization were identified to represent the scope of effort to support operations (Eppinger, 2012). These operations were then placed on a DSM and their interdependencies were defined within the interconnecting cells. A ‘0’ defined no dependency, a ‘1’ for moderate dependency, and a ‘2’ for high dependency. Figure 21 represents the initial dependency DSM. To better help visualize the differences in dependencies, a ‘0’ cell is white, a ‘1’ cell is purple, and a ‘2’ cell is light blue.

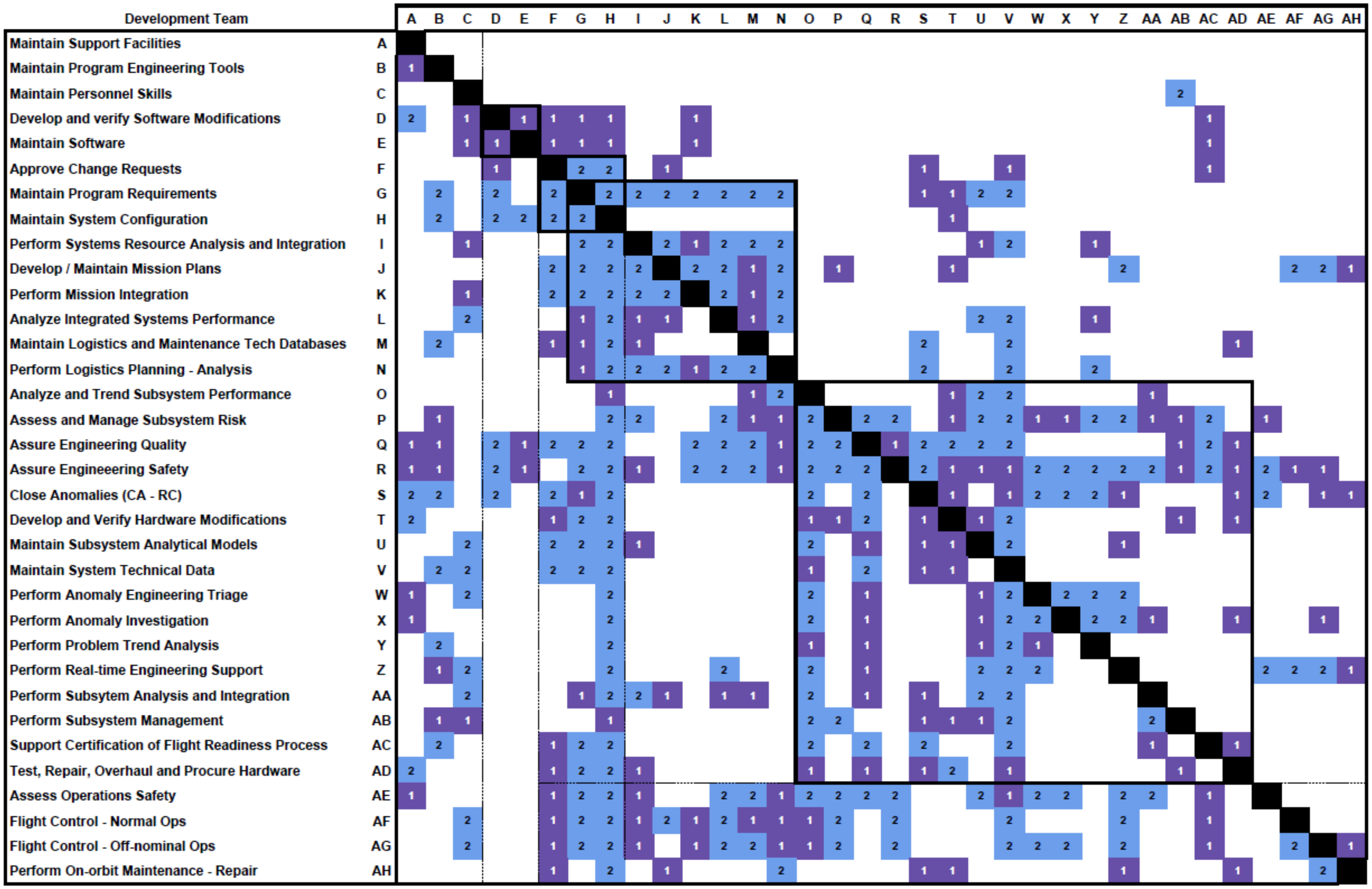


Figure 21. Organization DSM for ISS Sustaining Engineering Operations (Eppinger, 2012)

A second DSM was developed to capture the importance of critical skills retention within the organization. Each of the 36 functions was assigned a weighting factor representing the critical skills value. A ‘1’ defines a function that requires general engineering or project skill, a ‘2’ requires skills unique to NASA, while a ‘3’ requires an ISS-unique skillset (Eppinger, 2012). Figure 22 represented the critical skills distribution within a DSM. Within Figure 22, the color scheme reflects the severity of the critical skills needed. In other words, if a cell is green, comparatively, the skills needed is lower than a cell that is yellow, as well as a cell that is orange requires more skill than that of a yellow cell.

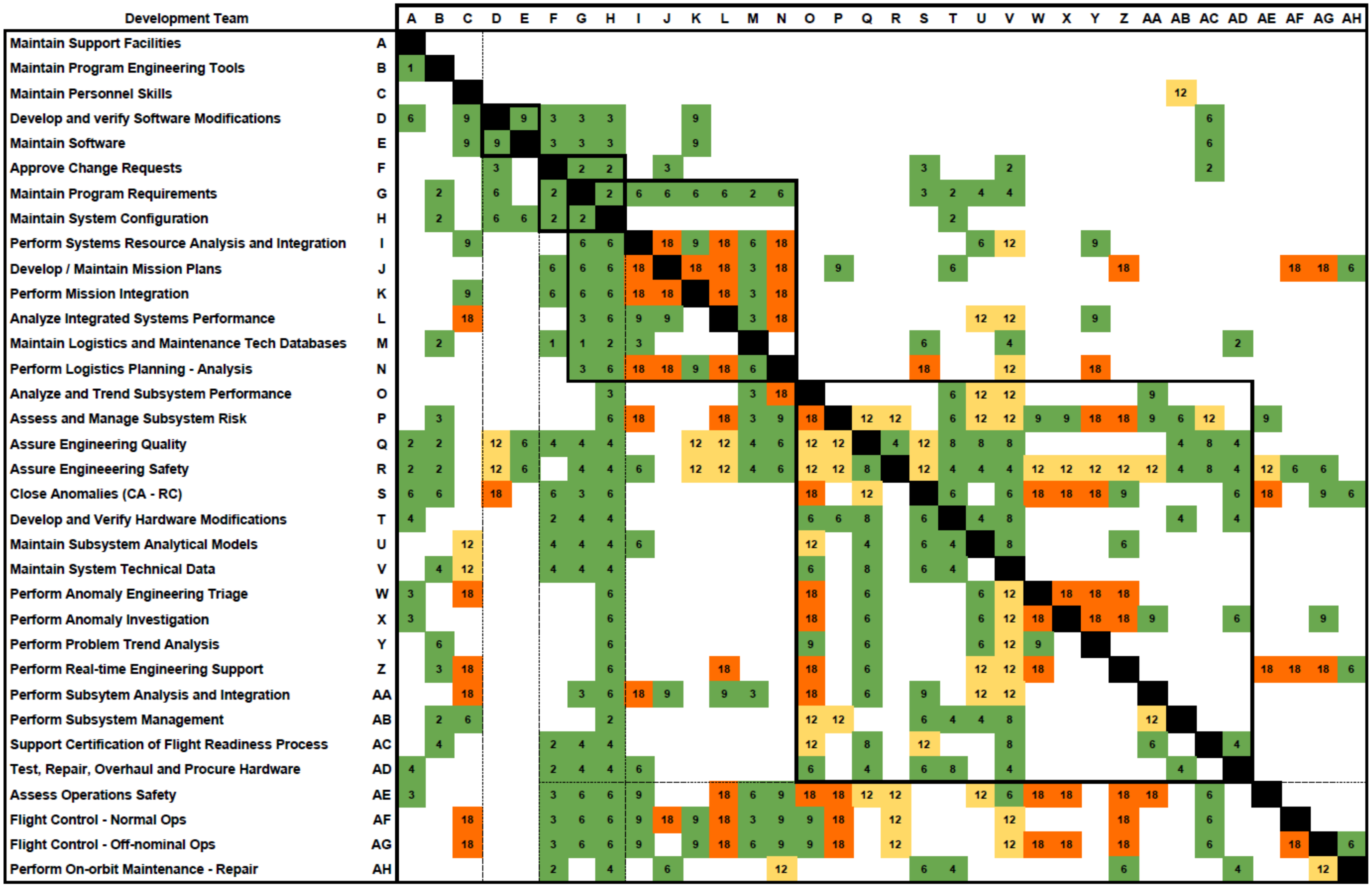


Figure 22. Critical skills DSM for ISS Sustaining Engineering Operations (Eppinger, 2012)

Lastly, a second analysis was conducted on the ISS organization and defined potential communication and coordination issues between functions. Like the critical skills valuation, a communication penalty was designated to each of the 36 functions. With eight different organizational units involved in the ISS's maintenance, if a function were performed within the same unit, the value would remain the same as Figure 22. If the function were performed between differing units, the value would be multiplied by 5. A cell that is colored red has a significant communication penalty, while a green cell has less of a communication penalty, comparatively. Figure 23 represents the communication penalty within an organizational DSM.

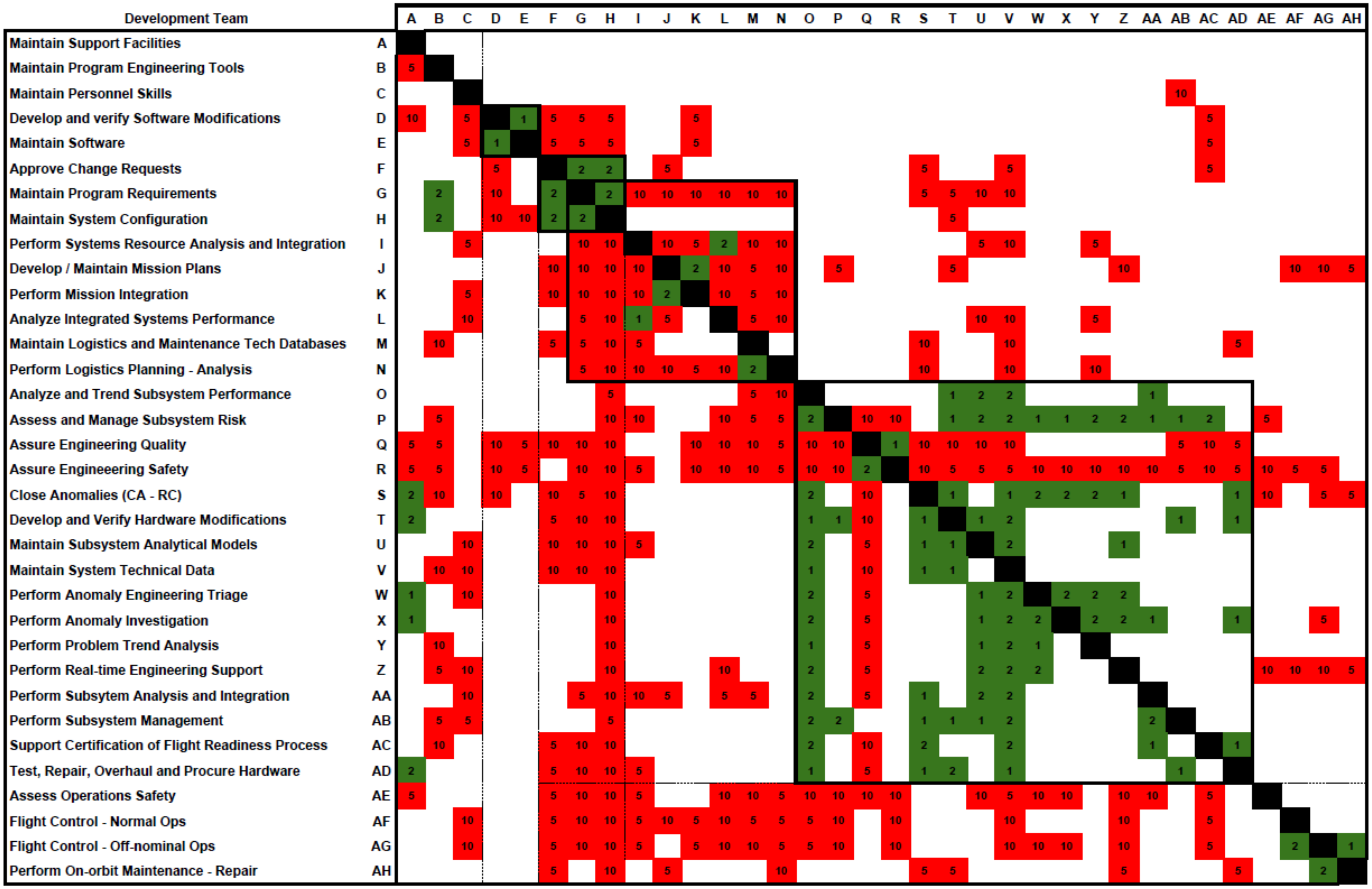


Figure 23. Communication Penalty DSM for ISS Sustaining Engineering Operations (Eppinger, 2012)

4.1.3 Visualizing Complexity in the NASA ISS Maintenance Organization

For the data presented in the ISS study to conform to the visualization method, interpretations on how each dataset affects complexity must be established. When categorizing the presented data, determining a correlation with complexity is needed. Additionally, the datasets must be designated as either component or interface. For this example, the breakdown of complexity definitions and correlations are summarized in Table 12. Since the OCM methodology does not employ weights, each dataset will be weighed equally. Again, due to the ISS study only having three datasets, the ISS example is meant to provide an additional example of application and may show skewed complexity results.

Table 12. Summary of Dataset Assignment

Dataset	Characterization	Correlation
Function Dependency	Interface	Positive
Critical Skills Value	Component	Positive
Communication Penalty	Interface	Positive

Starting with Figure 21 and the communication dependency dataset, the relationship between the metric assigned to dependency can be positively correlated to structural complexity. This inference is based on a study done by Alison M. Konrad and Deborah W. Brown (Brown, 1996) that showed a positive relationship between task instability and organizational dependency. As for the characterization of the data, function dependency is independent of internal function characteristics and should be characterized as interface-valuable data. For the critical skills value dataset, it is implied that an increase in specific engineering skills needed for a function denotes an increase in complexity within that function’s tasks. As inferred, this dataset will be characterized as component-valuable data that is positively correlated with complexity. Lastly, the communication penalty highlights the potential for miscommunication, information loss, et cetera. From this, another positive relationship between the penalty dataset and complexity can be inferred, as these communication mistakes increase the chance of emergent behaviors and therefore complexity. As the communication penalty defines the penalty between two functions, the dataset will be viewed as interface-valuable data that has a positive relationship.

Figure 24 represents a section of the nested numerical DSM and how the information can be stored in a single, centralized DSM. Reflecting the color schemes from the existing ISS DSMs, the purple cells identify the ‘dependency’ category, the red cells identify the ‘communication penalty’ data, and the black represent the ‘critical skills needed’ for the position. Below the initial DSM is the complexity contributions of that section of the organization. The full component complexity contributions due to the organizational functions can be seen in Table 13 and the summary of all complexity contributions can be seen in Table 14. Lastly, similarly to the example shown in Chapter 3, the final snapshot of the OCM DSM is shown in Figure 25 with the overall visualization of complexity shown in Figure 26. These DSMs follow the same coloring methodology, as the red color scale shows self-referencing, high complexity interfaces, while the green color scale show lesser complex interfaces.

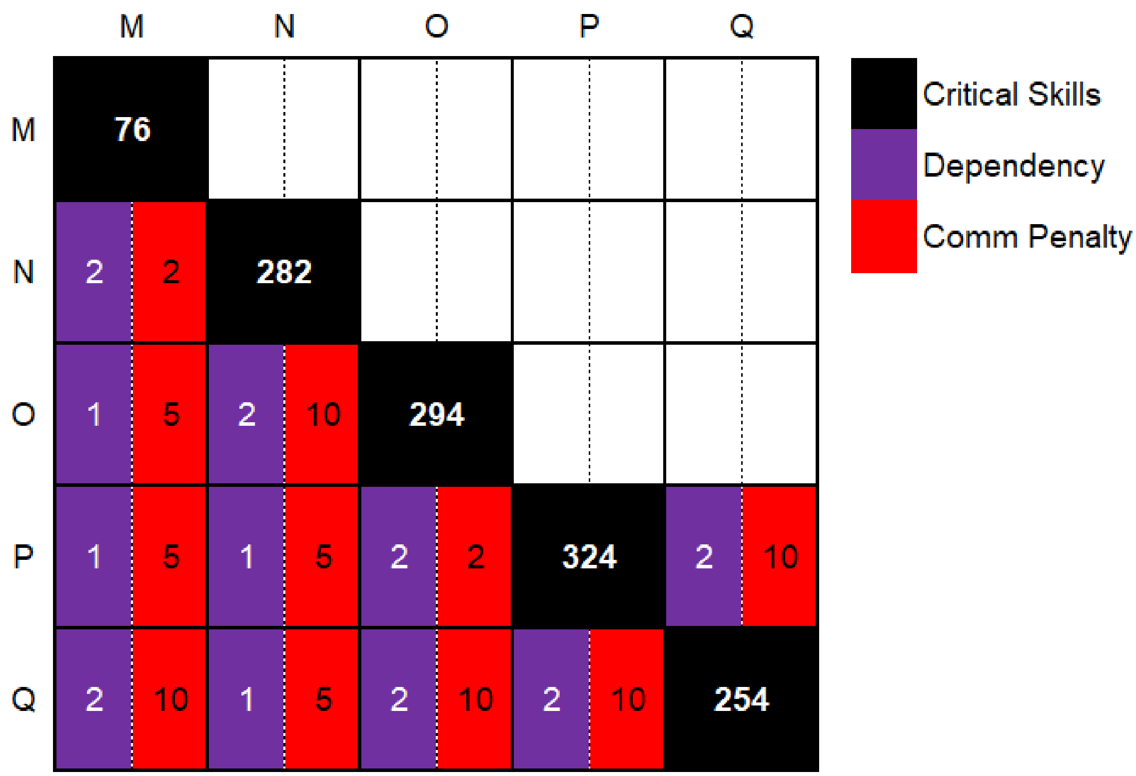


Figure 24. Snapshot of Nested Numerical DSM with Original Dataset

Table 13. Summary of Function (Critical Skills Value) Complexity Contributions

Function	Rank	Complexity Contribution
A	0	0
B	0.03	0.15
C	0.424	2.12
D	0.242	1.21
E	0.061	0.305
F	0.091	0.455
G	0.333	1.665
H	0.364	1.82
I	0.697	3.485
J	0.727	3.635
K	0.515	2.575
L	0.909	4.545
M	0.121	0.605
N	0.818	4.09
O	0.879	4.395
P	0.97	4.85
Q	0.636	3.18
R	0.788	3.94
S	0.909	4.545
T	0.273	1.365
U	0.455	2.275
V	0.667	3.335
W	0.606	3.03
X	0.545	2.725
Y	0.394	1.97
Z	1	5
AA	0.485	2.425
AB	0.212	1.06
AC	0.273	1.365
AD	0.182	0.91
AE	0.727	3.635
AF	0.576	2.88
AG	0.848	4.24
AH	0.152	0.76

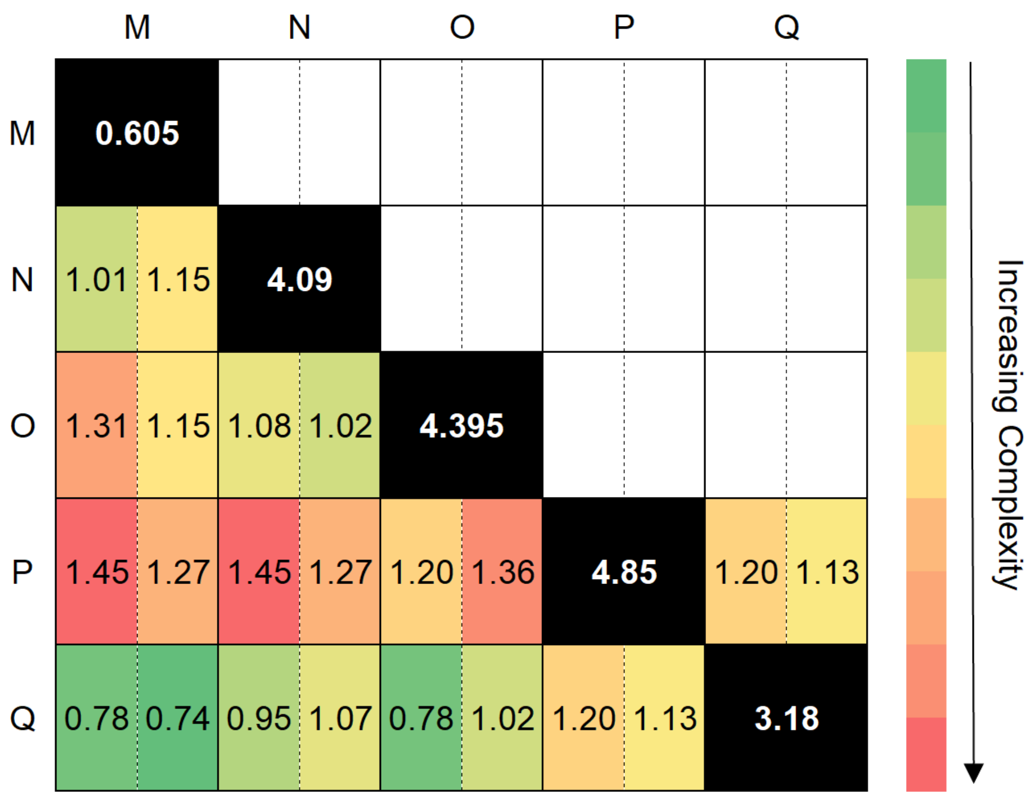


Figure 25. ISS Complexity Contributions (M to Q Snapshot)

Reviewing the results shown in Table 14 provides another example of the OCM being applied to an organizational architecture with appropriate datasets. While the NASA ISS Maintenance Program highlighted a single organizational configuration, the potential for trading and additional organization structures and human resources is made simpler with comparable complexity metrics. The complexity metric provides a simple, numerical visualization of the any organization and its arrangement. Again, looking at the component complexities (C_1) attributed to the functions provides an easily understandable metric to help aid in decision making. As for the interface complexities (C_2) and topological complexity (C_3), these metrics are representative of the organization's arrangement and how much influence this arrangement has on the chosen cost/performance metrics that define them. The topological complexity value of 2.67 strongly indicates a decentralized organizational structure ($C_3 \geq 2$). While not inherently a negative organizational characteristic, this information reinforces the need for interface management through interface mechanisms. As expected with decentralized systems, the interface complexity overwhelms the component complexities (359, 344.4 > 84.5). One thing to note is that the interface datasets outweigh the component datasets 2 to 1. With an increase in available datasets, as well as, the creation of datasets tailored for the OCM, a more robust and complete organizational analysis can be made.

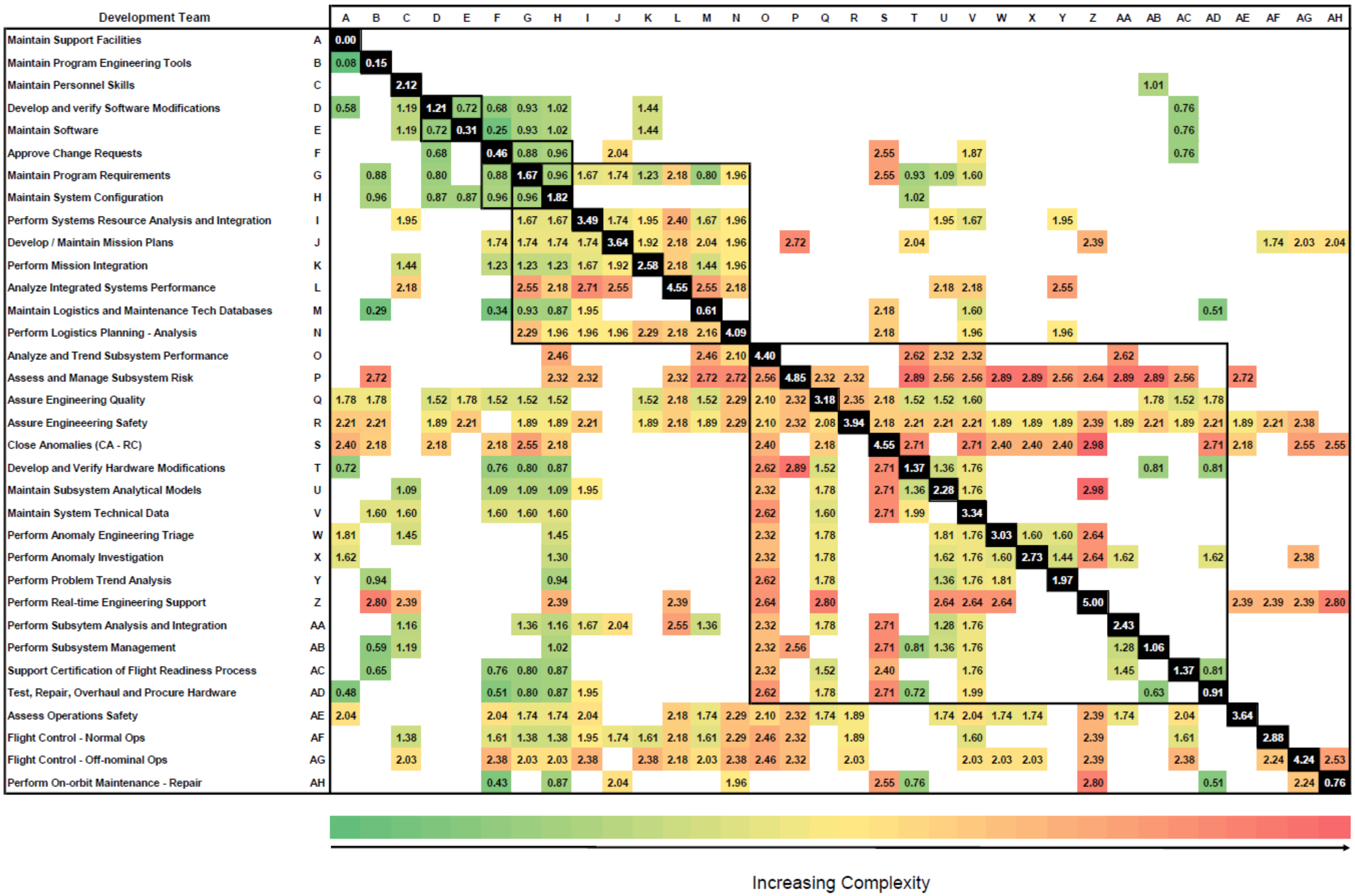


Figure 26. ISS Complexity Visualization

Table 14. Summary of the Structural Complexity of the ISS Maintenance Program

<table>
<tr><th>Function</th><th>C_1</th><th>$\sum C_{2.1}$</th><th>$\sum C_{2.2}$</th><th>C_3</th></tr>
<tr><td>A</td><td>0</td><td rowspan="35">359.0</td><td rowspan="35">344.4</td><td rowspan="35">2.67</td></tr>
<tr><td>B</td><td>0.15</td></tr>
<tr><td>C</td><td>2.12</td></tr>
<tr><td>D</td><td>1.21</td></tr>
<tr><td>E</td><td>0.305</td></tr>
<tr><td>F</td><td>0.455</td></tr>
<tr><td>G</td><td>1.665</td></tr>
<tr><td>H</td><td>1.82</td></tr>
<tr><td>I</td><td>3.485</td></tr>
<tr><td>J</td><td>3.635</td></tr>
<tr><td>K</td><td>2.575</td></tr>
<tr><td>L</td><td>4.545</td></tr>
<tr><td>M</td><td>0.605</td></tr>
<tr><td>N</td><td>4.09</td></tr>
<tr><td>O</td><td>4.395</td></tr>
<tr><td>P</td><td>4.85</td></tr>
<tr><td>Q</td><td>3.18</td></tr>
<tr><td>R</td><td>3.94</td></tr>
<tr><td>S</td><td>4.545</td></tr>
<tr><td>T</td><td>1.365</td></tr>
<tr><td>U</td><td>2.275</td></tr>
<tr><td>V</td><td>3.335</td></tr>
<tr><td>W</td><td>3.03</td></tr>
<tr><td>X</td><td>2.725</td></tr>
<tr><td>Y</td><td>1.97</td></tr>
<tr><td>Z</td><td>5</td></tr>
<tr><td>AA</td><td>2.425</td></tr>
<tr><td>AB</td><td>1.06</td></tr>
<tr><td>AC</td><td>1.365</td></tr>
<tr><td>AD</td><td>0.91</td></tr>
<tr><td>AE</td><td>3.635</td></tr>
<tr><td>AF</td><td>2.88</td></tr>
<tr><td>AG</td><td>4.24</td></tr>
<tr><td>AH</td><td>0.76</td></tr>
<tr><td>Org.</td><td>84.5</td></tr>
<tr><td colspan="3">Structural Complexity</td><td colspan="2">1962.6</td></tr>
</table>

Chapter 5

CONCLUSION AND FUTURE WORKS

Aerospace systems prove to be difficult to manage when it comes to complexity. While efforts have been made to manage the product complexity, the human element present in these systems must also be accounted for. From the need for a better way of visualizing the complexity within organizational structures, the proposed organizational complexity metric methodology was developed and presented. The method takes advantage of quantifiable human and interface datasets and relates them to complexity. This complexity metric is normalized to better compare, and trade differing organizational arrangements and is visualized in a single, centralized DSM for ease of representation. The Organizational Complexity Metric (OCM) DSM that leverages a color gradient was developed to aid in complexity analysis and management. This all-encompassing DSM shows quantifiable human data, as well as the complexity of the individuals and their communication interfaces. Along with the nested DSM visual aid, applying the OCM allows for the arrangement and complexity of any system to be displayed and represented with a concise complexity table.

Applying the OCM to the ISS, the ISS's organizational arrangement is now easily discernible. With a topological complexity of 2.67, the ISS organization is easily seen as a decentralized architecture ($C_3 \geq 2$). This arrangement information would normally be difficult to discern with a convoluted network diagram and even a DSM. The ability to quickly determine whether a system's organization is centralized, hierarchal, or decentralized allows systems engineers and program managers to better manage human resources and communication interfaces. The OCM also allows systems engineers to easily visualize the relative complexity density throughout the organization. The centralized storage of complexity information, whether it is individual contributions or interface contributions, provides a single location for organizational complexity information.

To expand upon the presented OCM it is suggested that a further understanding of quantifiable organizational datasets should be explored. This thesis assumes that all data presented is statistically sufficient and has a strong correlation to system complexity. Additionally, the implementation of weights is suggested if there is sufficient research and backing to support the data's relation to complexity. Weighting component and interface characterizations would allow for refinement when calculating complexity. In its current state, the OCM assumes that all complexity contributors are viewed as equally important in complexity quantification.

REFERENCES

Adler, Paul S. *The Learning Bureaucracy: New United Motor Manufacturing, Inc. Research in Organizational Behavior*, 1993, pp. 15, 111.

Argyres, Nicholas S. “The Impact of Information Technology on Coordination: Evidence from the B-2 ‘Stealth’ Bomber.” *Organization Science*, vol. 10, no. 2, 1999, pp. 162–180., doi:10.1287/orsc.10.2.162.

Banker, Rajiv D., et al. *Software Complexity and Maintenance Costs*, 1993.

Bearden, David A. “Complexity-Based Cost Estimating Relationships for Space Systems.” *2004 IEEE Aerospace Conference Proceedings (IEEE Cat. No.04TH8720)*, doi:10.1109/aero.2004.1368229.

Bearden, David A. *A Complexity-Based Risk Assessment of Low-Cost Planetary Missions: When Is a Mission Too Fast and Too Cheap?*, 2 May 2000.

Bernstein, Joshua I. *Multidisciplinary Design Problem Solving on Product Development Teams*, 2001.

Birkinshaw, J., et al. *Management Innovation*, 2008. *Academy of Management Review*.

Browning, Tyson R., et al. “Key Concepts in Modeling Product Development Processes.” *Systems Engineering*, vol. 9, no. 2, 2006, pp. 104–128., doi:10.1002/sys.20047.

Brown, Deborah Wright, and Alison M. Konrad. “Task Complexity and Information Exchange: The Impact of Nurses Networking Activities on Organizational Influence.” *Sociological Focus*, vol. 29, no. 2, 1996, pp. 107–124., doi:10.1080/00380237.1996.10570634.

Campagnolo, Diego, and Arnaldo Camuffo. “The Concept of Modularity in Management Studies: A Literature Review.” *International Journal of Management Reviews*, 2009, pp. 269–274., doi:10.1111/j.1468-2370.2009.00260.x.

Eppinger, Steven D., and Tyson R. Browning. “Design Structure Matrix Methods and Applications.” 2012, doi:10.7551/mitpress/8896.001.0001.

Foss, Nicolai J. “Selective Intervention and Internal Hybrids: Interpreting and Learning from the Rise and Decline of the Oticon Spaghetti Organization.” *Organization Science*, vol. 14, no. 3, 2003, pp. 331–349., doi:10.1287/orsc.14.3.331.15166.

Frey, Daniel, et al. “Part Count and Design of Robust Systems.” *Systems Engineering*, vol. 10, no. 3, 2007, pp. 203–221., doi:10.1002/sys.20071.

Heaslip, R. J. *Managing Organizational Complexity: How to Optimize the Governance of Programs and Projects to Improve Decision Making*, 2015. *PMI White Papers*.

Herkscher, C., and A. Donnellon. “The Post-Bureaucratic Organization: New Perspectives on Organizational Change.” *Long Range Planning*, vol. 28, no. 5, 1995, p. 124., doi:10.1016/0024-6301(95)90292-9.

Hoopes, David G., and Steven Postrel. “Shared Knowledge, ‘Glitches,’ and Product Development Performance.” *Strategic Management Journal*, vol. 20, no. 9, 1999, pp. 837–865., doi:10.1002/(sici)1097-0266(199909)20:9<837::aid-smj54>3.0.co;2-i.

INCOSE Systems Engineering Handbook. 2015.

Lindemann, U., et al. *Structural Complexity Management – An Approach for the Field of Product Design*, 2008.

Madni, Azad M. *Integrating Humans With and Within Complex Systems: Challenges and Opportunities*, 2011.

Nonaka, Ikujiro. “The Knowledge-Creating Company.” 2007, pp. 162–171.

Sadin, Stanley R., et al. “The Nasa Technology Push Towards Future Space Mission Systems.” *Space and Humanity*, 1988, pp. 73–77., doi:10.1016/b978-0-08-037877-0.50012-0.

Sarsfield, L. *Federal Investments in Small Spacecraft*, Sept. 1996. *Report for OIce of Science and Technology Policy, RAND Critical Technologies Institute, DRU-1494-OSTP*.

Schilling, Melissa A. “Toward a General Modular Systems Theory and Its Application to Interfirm Product Modularity.” *The Academy of Management Review*, vol. 25, no. 2, 2000, pp. 312–334., doi:10.2307/259016.

Sinha, Kaushik. Structural Complexity and Its Implications for Design of Cyber Physical Systems, Feb. 2014. MIT PhD.

Standardization and Program Practices Analysis Final Report. Volume II: Technical Analysis of Program Practices, vol. 2, 30 Nov. 1977.

Star, Susan Leigh, and James R. Griesemer. “Institutional Ecology, `Translations and Boundary Objects: Amateurs and Professionals in Berkeleys Museum of Vertebrate Zoology, 1907-39.” *Social Studies of Science*, vol. 19, no. 3, 1989, pp. 387–420., doi:10.1177/030631289019003001.

Steward, Donald. *A Very Brief Discussion of Information Driven Business Management: A Problem Solving Approach*, 2000.

Sturtevant. *System Design and the Cost of Architectural Complexity*, 2013. *PhD Thesis, MIT*.

Torbert, W. *Pre-Bureaucratic and Post-Bureaucratic Stages of Organization Development*, vol. 5, no. 1, pp. 1–25. *Interpersonal Development*.

Ulrich, K., and S. Eppinger. *Product Design and Development*, 1995. *New York: McGraw-Hill*.

Volberda, Henk W. “Toward the Flexible Form: How to Remain Vital in Hypercompetitive Environments.” *Organization Science*, vol. 7, no. 4, 1996, pp. 359–374., doi:10.1287/orsc.7.4.359.

Zenger, Todd R., and William S. Hesterly. “The Disaggregation of Corporations: Selective Intervention, High-Powered Incentives, and Molecular Units.” *Organization Science*, vol. 8, no. 3, 1997, pp. 209–222., doi:10.1287/orsc.8.3.209.