Spring 2011

Assessing system architectures: the Canonical Decomposition Fuzzy Comparative methodology

Jason Paul Dauby
Missouri University of Science and Technology

Department: Engineering Management and Systems Engineering

ASSESSING SYSTEM ARCHITECTURES: THE CANONICAL DECOMPOSITION FUZZY COMPARATIVE METHODOLOGY

by

JASON P. DAUBY

A DISSERTATION

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

SYSTEMS ENGINEERING

2011

Approved by

Cihan H. Dagli, Advisor
Ivan G. Guardiola
Steven M. Corns
Ann Miller
Jagannathan Sarangapani
Patrick J. Ford

ABSTRACT

The impacts of decisions made during the selection of the system architecture propagate throughout the entire system lifecycle. The challenge for system architects is to perform a realistic assessment of an inherently ambiguous system concept. Subject matter expert interpretations, intuition, and heuristics are performed quickly and guide system development in the right overall direction, but these methods are subjective and unrepeatable. Traditional analytical assessments dismiss complexity in a system by assuming severability between system components and are intolerant of ambiguity. To be defensible, a suitable methodology must be repeatable, analytically rigorous, and yet tolerant of ambiguity. The hypothesis for this research is that an architecture assessment methodology capable of achieving these objectives is possible by drawing on the strengths of existing approaches while addressing their collective weaknesses. The proposed methodology is the Canonical Decomposition Fuzzy Comparative approach. The theoretical foundations of this methodology are developed and tested through the assessment of three physical architectures for a peer-to-peer wireless network. An extensible modeling framework is established to decompose high-level system attributes into technical performance measures suitable for analysis via computational modeling. Canonical design primitives are used to assess antenna performance in the form of a comparative analysis between the baseline free space gain patterns and the installed gain patterns. Finally, a fuzzy inference system is used to interpret the comparative feature set and offer a numerical assessment. The results of this experiment support the hypothesis that the proposed methodology is well suited for exposing integration sensitivity and assessing coupled performance in physical architecture concepts.

ACKNOWLEDGMENTS

I would like to begin by thanking my advisor, Dr. Cihan H. Dagli for his guidance and support throughout my program of study. I am privileged to have received mentorship from a true expert in the field of systems architecture and engineering. Furthermore, as the founder of the Systems Engineering graduate program at Missouri S&T, his unique vision for synchronous net-centric distance education has enabled me to have an on-campus doctoral experience while working from my employer location in Indiana. Thank you to my S&T committee members: Dr. Ivan Guardiola, Dr. Steven Corns, Dr. Ann Miller, and Dr. Jagannathan Sarangapani for their valuable suggestions and contributions to this work. Finally, I want to express my gratitude to Dr. Patrick Ford for his mentorship as my sixth committee member and employer representative. His vast domain expertise helped shape my research into something that contributes to both academic and professional bodies of knowledge.

Thank you to the Airborne Electronic Attack Systems Division at the Crane Division, Naval Surface Warfare Center for sponsoring my doctoral studies. I would like to specifically thank Tom Dalheim for his professional mentorship and encouragement. Thank you to Steve Upholzer for arranging managerial and financial support for my studies. Thanks to Jeremy Barber for his collaboration on various subjects in computational electromagnetics and post analysis and to Justin DeWell for his support in creating the CAD models for this research.

Thank you to my parents, Paul and Janet, for their support and sacrifices throughout my life. You are amazing parents and I appreciate all the opportunities you have given me. Thank you to my sisters, Vanessa and Valerie, for their support and for putting up with me while we were growing up.

Finally and most importantly, thank you to my wife and kids for their love, support, and understanding during the course of my studies. Rebecca, I dedicate this work to you. It would not have been possible without your love and help maintaining the family/work/school balance. Grace and Alden, thank you for your patience while daddy was in class and for “helping” me with my school work. I promise to return the favor.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

NOMENCLATURE

Symbol	Description
A	An arbitrary set
A^c	Complement to arbitrary set A
Ø	Null set
μ_A	Fuzzy membership in arbitrary set A
P_t	Power transmitted
P_r	Power received
G_t	Gain of transmit antenna
G_r	Gain of receive antenna
λ	Wavelength
d	Link distance
h_t	Height of transmit antenna
h_r	Height of receive antenna
J/S	Jammer to signal ratio
A_S	System attribute assessment
δ_i	High-level quality attribute
γ_i	Constituent variable for δ-level elements
β_i	Constituent variable for γ-level elements
α_i	Elemental quantity and constituent variable for β-level elements
θ	Elevation angle
ϕ	Azimuth angle
e_i	Error in predicted gain for a given observation angle
η_p	Polarization efficiency
ξ	Angle between two electric field vectors
x'	Input to a type-2 fuzzy membership function
u_n	Primary membership function grade
$\mu_{\tilde{A}}$	Secondary membership grade in set A

1. INTRODUCTION

Architecture establishes a framework for the subsequent embodiment of a system artifact. It represents the structure, components, connections, and constraints of a system (Rechtin 2000, 5). The impacts of decisions made during the development and selection of the architecture propagate throughout the entire system lifecycle. A disciplined systems engineering effort produces a successful artifact only insofar as the functional architecture successfully captures the needs of the user and the physical architecture represents a stable and achievable physical form.

The art and science of systems architecting is a subject of intense interest for governments, industry, and academia alike. As systems become larger, more complex, and more highly integrated, the cost of failure increases rapidly. This can be observed in failures of electrical power grids, military weapons systems, and space programs. Successful system development begins with the selection of stable system architectures.

In a most fundamental sense, system architecting is a search process that hierarchically reduces the ambiguity in a system description as it seeks to find a solution that represents the most attractive balance between multiple competing objectives. Ambiguity in system architecting is inevitable. The discipline intentionally positions itself between unclear user needs, wants, constraints, and concerns and the clear specificity of a detailed system design to meet those needs. A systems architect must assume a high-level top-down perspective on any potential system solution in order to remain open to the wide range of possibilities available in the search space being explored.

Two basic mechanisms are required for any search process: a means to generate candidate solutions and a means to assess those candidates with respect to some measure of attribute fulfillment. Realistic assessments guide the architecting process toward more inherently robust solutions. The challenge for systems architects is to perform a realistic multi-attribute assessment of an inherently ambiguous solution.

1.1. THE PROBLEM AND MOTIVATION FOR RESEARCH

An assessment can be performed in many different ways and with varying degrees of rigor. Subject matter expert interpretations, intuition, and heuristics are performed quickly and often guide system development in the right overall direction, but these methods are subjective and exhibit poor repeatability. They are generally good at reducing immense search spaces very quickly, but due to their lack of resolution and analytical rigor, these methods rarely provide an objective means to distinguish between complex architecture candidates as the solution space narrows. The intrinsic problem with architecture assessment is that the subject being assessed is inherently ambiguous. Even so, an effective assessment mechanism must be realistic, objective, and repeatable in order to be defended and accepted by the development community.

Analytical assessments, such as timing and constraint relationships, offer a more objective means to gauge system performance. Unfortunately, traditional analytical assessments dismiss part of the complexity in a system by assuming severability between system components. Each system, subsystem, or component is decomposed and assessed in isolation. The total system response is assumed to be a linear combination, or superposition, of the individual responses. It is known that coupling variables exist between system components and between the system and its environment. These coupling variable are believed to be one source of emergence and are lost through improper model decomposition.

Negative impacts resulting from unexpected system performance have prompted researchers to attempt to probe architectures at points where they interact with each other and their environment. The integration impacts of these interactions in a system of systems have been studied and published by Gagliardi, Wood, Klein, and Morley (2009). Others have taken the approach to add synthetic environments in executable models to better understand behavior and reduce system integration risk (Gardiner 2009). The behaviors uncovered in these analyses can be traced to interface attributes in the overall system architecture.

It would be advantageous, therefore, to characterize the architecture and its interfaces, as accurately and early as possible. This characterization begins in the architecture search process. Since a search process hinges on the assessment of a candidate solution, there is motivation to research and develop an assessment methodology that improves fidelity and objectivity while remaining tolerant of ambiguous system descriptions.

1.2. RESEARCH OBJECTIVES AND HYPOTHESIS

In response to the challenges and research motivation identified above, the overall objective of this research was to understand and improve the methods, processes, and tools (MPTs) used in system architecting, specifically in the area of architecture assessment. The goal was to retain the ability to examine coupling variables in a system architecture thereby making the assessment more realistic and providing insight into certain emergent properties. A suitable assessment methodology must be stable, repeatable, consider system coupling, and be analytically rigorous so as to be peer reviewable and defensible within the system development community. Finally, the methodology to be developed must be capable of tolerating ambiguity in the architecture description. The significance of this is that the MPTs must facilitate enough specificity to examine inter- and intra-system coupling variables while being general enough to apply to an entire class of solution types.

The hypothesis for this research was that it is possible to develop an architecture assessment methodology capable of achieving these objectives by drawing upon the strengths of existing assessment techniques while offering unique capabilities to compensate for their collective weaknesses. The resulting methodology seeks to leverage the desirable aspects of several different assessment techniques in a way that offers a novel combination of analytical rigor and tolerance for ambiguity. The process for identifying the strengths and weaknesses of current approaches and synthesizing the proposed methodology is outlined below.

1.3. RESEARCH APPROACH AND ORGANIZATION OF THE DISSERTATION

1.3.1. Understanding Systems Architecting Approaches and Issues. To develop a methodology that addresses a specific opportunity for improvement in architecture assessment, it was necessary to understand existing approaches to generating and assessing architectures. Section 2 summarizes the results of a literature review by describing many aspects and issues related to systems architecting. Included in this is a survey of traditional architecture assessment approaches. Several methods are enumerated and described in terms of their strengths and weaknesses. The result of the literature search was the identification of the research opportunity that was pursued and is presented herein.

1.3.2. Developing a Toolset and Integrating it into a Methodology. As a possible solution to the specific gaps identified in the literature survey, this research resulted in the identification of a number of tools and techniques capable of providing increased fidelity in architecture assessment while remaining tolerant of ambiguity. This toolset has been integrated into an overall assessment methodology. Section 3 describes the individual tools and their integration into the Canonical Decomposition Fuzzy Comparative (CDFC) assessment methodology. The section concludes by identifying how the CDFC approach fulfills the research goal of leveraging the strengths of existing techniques while offering unique attributes to address gaps in architecture assessment.

1.3.3. Testing the Methodology. The underlying principles of the CDFC methodology suggest that the approach is suitable and achieves the objectives identified above. In order to confirm the suitability of the approach, and offer commentary on its relative strengths and weaknesses, the CDFC methodology was tested by applying it to a representative architecture assessment scenario. Section 4 describes the technical attributes of the peer-to-peer wireless network system to be architected. Three physical architecture alternatives are presented and the design details for the customized CDFC elements are provided.

1.3.4. Evaluating the Results and Validating the Methodology. Section 5 provides the results of the architecture assessment described in Section 4. In practice, only the final assessment value would be of interest to the system stakeholders. However, to understand the workings of the CDFC approach, intermediate outputs are provided as well. Section 6 provides commentary on the use of CDFC for the experiment described in Section 4 and executed in Section 5. This commentary includes a discussion on the issue of validation for the integrated CDFC methodology and its constituent elements. Finally, areas of future research opportunities are identified in Section 7.

2. SYSTEMS ARCHITECTING: ARCHITECTURE GENERATION, ASSESSMENT, AND EMERGENCE

Systems architecting presents several unique challenges to the traditional bottom-up engineering approach. Firstly, systems require holistic balance between multiple competing success criteria and therefore transcend any single-factor optimization within an individual technical discipline. This makes it a top-down "big picture" discipline.

Secondly, system architectures are inherently ambiguous. Maier and Rechtin (2002) describe the architect's situation as "ill-structured" and difficult to optimize in quantitative terms due to the inability to accurately measure architecture attributes. Rechtin (1991, 35) suggests that "meaningful measurements may be impossible or impractical" for complex systems that have yet to be designed and built. The ambiguity in a system description reinforces this perspective, but the need to attempt a meaningful measurement exists nonetheless. This need is supported by the architecture evaluation challenges identified by Levis (2005). Because a system in the architecting phase has yet to be designed and built, these are not traditional empirical measurements. Instead, they are *estimates* of a measure and typically form the foundation of architecture evaluations.

Thirdly, system elements are highly interconnected. According to the International Council on Systems Engineering (INCOSE), "A system is a construct or collection of different elements that together produce results not obtainable by the elements alone". A collection of interconnected elements requires a number of interfaces. Maier and Rechtin (2002, 9) observe that "the architect's greatest concerns are still, and should be, with the systems' connections and interfaces". Extending this interconnection of elements further, one can readily observe that the current trend is to interconnect entire systems into a system of systems in order to achieve capability not obtainable from any of the individual systems. With this interconnection comes the manifestation of emergent behavior. White (2007) offers a definition of emergence as "something unexpected in the collective behavior of an entity within its environment, not attributable to any subset of its parts". Unexpected qualities, performance, or behaviors can contribute both positively and negatively the system. Understanding element coupling, interconnection, and emergence is critical to the system architecting process.

As identified in Section 1, the architecture search process must include mechanisms to generate and assess candidate solutions. It is sometimes difficult to separate the generation mechanism from the assessment mechanism. Furthermore, a clear taxonomy of assessment approaches is difficult to achieve due to the hybridization of many approaches. To facilitate a logical progression through several systems architecting issues, existing methods and techniques for generating architectures are discussed first. This discussion is followed by an overview of popular assessment approaches. Finally, the topic of emergence is addressed due to its profound impact on the success or failure of a system.

2.1. GENERATING ARCHITECTURES

To perform an architecture assessment, one must assume than an architecture candidate already exists. In order to understand the relationship between the architecture search process and architecture assessment, it is important to understand how alternatives are generated. There are a number of traditional, model-based, and automated approaches, each incorporating architecture assessment to a different degree.

2.1.1. Traditional Approaches. An overview of the traditional approaches is provided in Maier and Rechtin (2002, 1-3).

- *Normative* – a solution-based approach that assumes an architecture solution is available from authoritative sources. This approach selects an architecture solution from a list of existing designs, codes, or specifications. Assessment does not play a large role in this approach because the design choices are already dictated. Or from another perspective, the assessment was already performed and only suitable candidates are provided.
- *Rational* – a method-based approach emphasizing the application of scientific principles and underlying design theory to arrive at architecture solutions. An element of assessment is assumed to be included in the development of the architecture by virtue of the fact that this is an analytical approach.
- *Participative* – a stakeholder-based approach that produces architectural candidates through collaborative interaction between subject matter experts. Consciously or not, heuristic assessment is performed in real-time as the participating community discusses architecture solutions.

- *Heuristic* – an experience-based approach that applies universal ‘truths’ discovered through past experiences to guide the development of the architecture. In this case, the heuristics themselves are the assessment mechanism and they are applied throughout architecture development.

2.1.2. Model Based Approaches. Model based systems engineering is a new and evolving area of interest. It represents a shift from document-centric systems engineering processes to a model based paradigm (Friedenthal, Moore, and Steiner 2009). Frequently, the model based approach represents an automated and augmented means to perform the traditional architecting and assessment methods. Graphical interfaces, syntax and logic checking, timing and constraint analysis, impact analysis and traceability, and standardized viewpoints, are common features of tools employing modeling languages such as the Systems Modeling Language (SysML) (Object Management Group 2008). SysML is growing in popularity as a visual modeling language that provides semantics and notation to facilitate the description of system behavior, structure, requirements, and constraints (Friedenthal, Moore, and Steiner 2009). Model based approaches can greatly increase the portability and exchange of architecture models between stakeholders. Model based approaches can also automate the architecture generation process and offer tools to perform computational evaluations of the modeled solution. While many SysML tutorials and examples begin after the selection of a system architecture, Weilkiens (2006) illustrates the use of SysML from the very outset of requirements gathering and system architecting.

The Object Management Group (2003) promotes several initiatives emphasizing Model Driven Architecture (MDA) and Model Driven Engineering (MDE). Cloutier (2008) applied MDA concepts to complex systems engineering using SysML to increase the efficiency of the architecting process through the use of patterns. Bowman (2008) used high-level models to generalize implementation-specific details of candidate designs. These approaches accelerate the beginning phases of the architecture search process, and are therefore useful in narrowing the initial search space. However, the use of patterns and generalizations does not support an increased ability to understand coupled system behavior in a way that would illuminate emergent properties and distinguish between a narrowing set of candidate architecture solutions.

An assessment of architecture solutions can be facilitated through model based approaches. In SysML, the parametric diagram allows the architect to establish and monitor constraint equations governing the relationships between system components and between the system and its environment. However, this form of assessment hinges on how accurately the constraint equation models system behavior at different levels of interaction. If the constraint equation is overly simplistic, omits variable dependencies, or ignores coupling, the assessment will be inaccurate and potentially misleading. Peak et.al. (2007a, 2007b) have made progress in demonstrating how simulation-based design through SysML, and custom interfaces to other computational tools, can be used to improve system design through model reuse and provide rapid assessment based on detailed constraint equations.

A model based systems architecting approach also allows the creation of executable models so as to examine behavioral dynamics. This enhances the search and assessment process given that system performance arises from an interaction between its parts, and is not contained in any single system element. Colored petri nets have been used as an extension to SysML to provide dynamic analysis of the interaction between system components (Wang and Dagli 2008a, 2008b).

2.1.3. Computational Intelligence Approaches. The use of computational intelligence is an emerging approach for generating and selecting system architectures. Computational intelligence approaches offer formal mechanisms to solve search, optimization and classification problems that are difficult to evaluate in closed analytical forms. These approaches utilize a variety of tools. Eberhart and Shi (2001) indicate that the field of computational intelligence shares much overlap with "soft computing". In many of his works, Lotfi Zadeh suggests that soft computing is a collection of methodologies including fuzzy logic, neurocomputing, evolutionary computing, and probabilistic computing. He also suggests that in contrast to hard computing, soft computing is able to operate in spite of quantitative imprecision and ambiguity. This makes soft computing, and therefore computational intelligence, an attractive toolset for systems architecting. The prominent tools of computational intelligence are introduced at a high level herein, but can be examined in more detail as referenced below or collectively in computational intelligence texts such as Eberhart and Shi (2007).

2.1.3.1 Neurocomputing. Neurocomputing is concerned with the processing of information through learning structures. Whereas programmed computing executes predetermined logic, neurocomputing incorporates a learning process within a neural network architecture (Ham and Kostanic, 2001). A neural network is a collection of interconnected elements called neurons. Each neuron accepts one or more inputs which are individually weighted. The collection of weighted inputs triggers some level of activation in the neuron depending upon its threshold level and transfer function. The nonlinearity of the transfer function is what allows a collection of interconnected neurons to learn, and subsequently predict, complex patterns.

The complex patterns of interest in systems architecting are widely varied since complex behavior is one of the hallmarks of large-scale systems. Connectivity and interdependence characterize complex adaptive systems as well as artificial neural networks. The number of interactions and strength of coupling make system behavior modeling difficult (Calvano and John 2004). Therefore, a highly coupled learning structure is an attractive approach.

Integration sensitivity functions (Dauby and Dagli 2009b) and complex trading rules for financial markets (LeBaron 2001) have been modeled using neurocomputing techniques. There are also numerous opportunities for the application of artificial neural networks and reinforcement learning in evolutionary architecting (Dagli and Kilicay-Ergin 2009). In this approach, the gaps in a candidate architecture can be determined, implemented, and fed back into the evolutionary process so as to develop a robust architecture (Chen and Han 2001; Chen and Clothier 2003).

2.1.3.2 Evolutionary Computing. Genetic algorithms (GAs) are a stochastic optimization process modeled on biological evolution. Much of the modern work on population-based evolution is founded on Holland (1975). According to Mitchell (1998), there is no singularly agreed-upon definition of a GA, however, most GAs have a number of common elements. The first is a population of individuals. The chromosomes are a representation of an individual in a search space. The second feature is a competitive selection of individuals according to a predefined fitness function. Thirdly, a crossover mechanism is used to generate new individuals from the selection of fittest parents. Finally, a mutation operator is employed to inject minute alterations into random chromosomes in order to prevent the algorithm from converging on a local minimum or maximum within the search space. Subjects such as parent selection, crossover methods, and mutation schemes are areas of active research.

A genetic algorithm approach to system architecting is introduced in Simpson and Dagli (2008a) and Wang and Dagli (2008c). The computationally intelligent approach is expanded to system of systems architecting by Simpson and Dagli (2008b). Several variations on the computational intelligence approach have been explored, including its use at different levels of architecture resolution. The automated approach can be applied iteratively during the architecting process to accommodate the hierarchical reduction in ambiguity that is produced.

The computationally intelligent architecting approach is also referred to as Smart System Architecting (SSA). Current trends in SSA are presented in Dagli, Singh, Dauby, and Wang (2009). In any variation of SSA using genetic algorithms, the architecture structure is encoded in the chromosome representing an individual in the population. The genes in this chromosome become the inputs to an assessment, or fitness, mechanism that scores an individual and provides the selection basis for future generations. In all of the instances cited, the authors used some variation of a fuzzy inference system (FIS) to perform the assessment of the architecture. A fuzzy inference system, or fuzzy associative memory, performs a mapping function between one or more input values and a single output value. The mapping is referred to as a fuzzy inference system because it is implemented using fuzzy logic.

2.1.3.3 Fuzzy Logic. Fuzzy logic allows repeatable computational processes to be applied to imprecise relationships using linguistic variables. Most notably, fuzzy logic extends classical set theory by supporting partial membership in a set. In other words, a quantity can have degrees of membership in one or more classes – including a set A and its complement A^C.

Classical set theory establishes an all-or-nothing proposition. An element is either a member of a set, or a member of its complement, but not both (Ziemer 1997).

$$A \cap A^C = \varnothing \tag{1}$$

The boundary between two classical sets is therefore crisp. Unfortunately, the architecture assessment data to be classified, while crisp, are acknowledged to result from ambiguous component definitions. Even more problematic is that the sets in which one wishes to classify a performance estimator are rather ill-defined. They are usually stated linguistically (e.g. better (B), same (S), worse (W)). When using classical sets to classify an architecture performance estimator, one quickly finds issue assigning membership to *B* rather than *S* when the estimator is only minimally better. The natural linguistic tendency is to classify the architecture as "somewhat better". The mathematical implications of this classification require an architecture assessment element to have membership in both *B* and B^C which is clearly an impossibility.

Fuzzy sets extend classical sets by acknowledging that classifications can be ill-defined, or "fuzzy". The most important characteristic of fuzzy set theory is that it supports partial membership. If the membership function for an element in *A* is valid for the continuous interval [0, 1], then its membership in A^C is simply the complement of its membership in *A*.

$$\mu_{A^C} = 1 - \mu_A \tag{2}$$

In this way, the element can have degrees of membership. Fuzzy logic provides the mathematical operators necessary to perform calculations on fuzzy sets. A treatment of fuzzy sets and fuzzy logic can be found in Zadeh (1996).

A Fuzzy Inference System establishes an association between one or more input variables and an output quantity using fuzzy logic. First, fuzzy membership functions map input variables into fuzzy sets. Fuzzy rules establish the logical mapping between input and output conditions. These rules take the form of IF-THEN statements and can include logical connectives such as AND, OR, and NOT. Fuzzy operators determine the mathematical operation associated with each connective. An implication method specifies each fuzzy rule's impact on fuzzy output membership functions. A FIS usually contains several fuzzy rules in its rule set. The aggregation process combines the implications of all the fuzzy rules. Finally, the aggregated results are defuzzified using a prescribed method. More information on the operation of Fuzzy Inference Systems can be found in MathWorks (2009). The output of the FIS is a crisp number, which in the context of architecture assessment, represents the final assessment of the approach.

2.2. ASSESSING ARCHITECTURES

No matter how architectures are generated, there must be some means by which they are evaluated. The evaluation may be a clearly distinguishable fitness assessment, as in the evolutionary architecting approach, or it may be implicitly contained within the approach such as through the use of heuristics. There are many ways an architecture can be characterized, and a number of terms are frequently used with respect to architecture evaluation. To ensure consistency and precision in the discussion of architecture assessment, the author suggests a working definition for several of these terms.

- *Evaluation* – a general term used to describe any process by which the properties of an architecture are determined and compared. This includes verification, validation, and assessment.
- *Verification* – the distinguishing characteristic of verification is a traceability analysis to determine whether all the allocated functionality and constraints have been honored in the architecture solution.
- *Validation* – a process to determine whether a proposed architecture meets the high level needs of the user through the definition and inclusion of all required attributes.
- *Assessment* – a process to distinguish the amount or degree of satisfaction with regard to a key architecture attribute so as to differentiate between alternatives.

Verification and validation offer a pass / fail judgment of an architecture, but do little to differentiate between them on the basis of how well they achieve a set of objectives. This research is specifically concerned with understanding current practices in performing architecture assessments.

2.2.1. Multiple Criteria Decision Making. As was previously mentioned, the selection of a system architecture requires the satisfaction of many competing success criteria. The architecting process can be described as a multi-attribute optimization problem. Multiple Criteria Decision Making (MCDM) and Multi-Attribute Utility Theory (MAUT) are tools specifically suited for selecting among choices based on multiple criteria (Dyer et. a. 1992). Xu and Yang (2001) suggest that there are two distinct types of MCDM implementation. The first is for selection among a finite number of choices utilizing a decision matrix ($m \times n$) comprised of m alternatives with n attributes. Employing MCDM therefore assumes that values for the attributes in the decision matrix are already known. The second implementation is related to design where individual attributes may take on an infinite number of distinct values. This scenario may also be referred to as multiple objective optimization.

To be successfully applied to systems architecting, any assessment approach should either be inherently multi-attribute or capable of being made multi-attribute through a factor combination scheme. The subsequent discussion on assessment is focused on the approach for estimating attribute qualities and quantities. Some approaches can be made inherently multi-attribute based on their embodiment. Others may simply offer an input quantity to be included in an ($m \times n$) MCDM decision matrix.

2.2.2. Heuristic Assessment. Heuristics are rules of thumb, or general principles, that can be applied very quickly to initiate architecture search and explore vast trade spaces. According to (Maier and Rechtin 2002, 26), a heuristic is an abstraction of experience. It represents lessons learned, trends identified, and statements that are readily accepted as general truths. These are expressed in verbose descriptions utilizing natural human language. The following is an example of a heuristic on the use of heuristics:

Heuristics work best when applied early to reduce the solution space (Cureton 1991).

The use of heuristics in systems architecting and systems engineering is consistent with an experiential discipline characterized by both individual and collective lessons-learned. There is abundant literature attempting to collect, organize, interpret, and present heuristics related to systems engineering. These collections have been grouped and are easily reference by system type and phase of the systems lifecycle. Maier and Rechtin (2002) offer heuristics for builder-architected, manufacturing, social, software, and collaborative systems. Rechtin (2000) offers additional heuristic guidance for organizations as complex social architectures.

While heuristics remain useful to guide architecture development, assessments based on heuristics or subject matter expertise remain open to challenge. One practitioner's judgment may be inconsistent with, or even in opposition to, another's. It can be hard to defend the basis on which an assessment is made due to the lack of objective data or proven relationships. Part of this limitation lies in the fact that there is generally no way to separate correlative from causative relationships in a heuristic. This is especially true as system complexity increases and the fundamental interactions between system components are poorly understood.

A positive aspect of heuristic assessments is that they are very tolerant of uncertainty and ambiguity. This makes many heuristics applicable to a wide range of scenarios. Usually expressed in succinct natural human language, heuristics are compact and easy to apply. Traditional computational processes have found it difficult to offer such broadly useful techniques in a compact and ambiguity-tolerant form. However, recent advances in Computing with Words may help facilitate a computational embodiment of systems architecting heuristics.

2.2.3. Computing with Words. Computing with words (CW), or linguistic computation, is a concept being proposed by researchers in computational intelligence. CW draws its inspiration from the ability of humans to perform a wide variety of mental tasks without measurements or numerical computation (Zadeh 1999). Instead, humans rely on perceptions and words. Zadeh describes four principle rationales for using CW. These include instances where numerical values are not known, not needed, cannot be solved for, or cannot be defined. In these instances, humans naturally rely on linguistic quality attribute descriptions. In the early stages of system architecting, quantitative system descriptions are similarly unknown, not needed, cannot be solved for, and cannot be defined. Singh and Dagli (2010a) propose a framework to evaluate potential architectures using a linguistic representation model. This framework focuses on solution independent architecture, explored at the conceptual level, in the form of Measures of Effectiveness. The process involves the assembly of a panel of experts who define a number of linguistic preferences related to key evaluation attributes. The mechanics of the proposed approach allows for the aggregation of linguistic consequent variables and associated degrees of belief.

It would appear that the principle contribution of CW is the ability to computationally model the human decision making process. The impact of this contribution is that a CW assessment model can be incorporated into Smart System Architecting approaches such as those utilizing evolutionary computation. These architecting approaches can generate thousands of architectures very quickly. It would be impossible for a human, or team of humans, to evaluate the architecture alternatives at the rate they are generated. Thus, the rate of human assessment becomes the limiting factor in the search process. By modeling the human decision making process through the use of CW, assessment can be accomplished by the evolutionary search process at the rate of architecture generation. In summary, CW represents an algorithmic embodiment of architecting heuristics.

2.2.4. Fuzzy Assessment. Fuzzy assessment is one method of adding mathematical rigor to the processing of inherently ambiguous information. The use of fuzzy methods in architecture assessment is based on the design of a fuzzy inference system (Singh and Dagli 2009). Fuzzy inference systems provide a mathematically rigorous and repeatable means of performing architecture assessment. Depending upon their implementation and the level of architecting being conducted, there can still be significant subjectivity in this form of assessment. Without prior analysis, the membership function shapes must be created heuristically. Simpson and Dagli (2008a) suggest that "best professional judgment and expert opinion" are suitable for use in determining the fuzzy membership functions in the fuzzy inference system. This perspective is echoed in Singh and Dagli (2009).

As mentioned, the concern with relying upon input from subject matter experts is that an element of unrepeatable subjectivity is inserted in the assessment. This weakens the defensibility of the approach. Some degree of subjectivity will always remain since the membership functions ultimately encode a stakeholder's perspective. The goal is to minimize the uncertainty and variability in the creation of membership functions by focusing on elemental relationships. These elemental relationships are cognitively easier to grasp than multi-attribute relationships and provide the underlying basis for a high-level multi-attribute assessor comprised of many low-level fuzzy inference systems. As the elemental assessors are developed and integrated into system assessors, the overall architecture assessment process is able to simultaneously evaluate the multiple competing criteria by which an architecture is judged.

2.2.5. Probabilistic Assessment. Probabilistic approaches recognize the uncertainty associated with assessing ambiguous quantities. The Evidential Reasoning (ER) form of MCDM is another quantitative technique to address the need for multi-attribute assessment. The ER variant of MCDM is formed by incorporating degrees of belief into the MCDM decision matrix (Xu and Yang 2001; Purewal, Yang, and Grigg 2009). The degrees of belief are subjective probabilities associated with the assessment grades. These probabilities represent a confidence level in the nominal assessment. To facilitate the determination of assessment grades and degrees of belief, architecture qualities being assessed using the multiple criteria approach can be decomposed into underlying contributing factors. These simpler sub-factors may be quantitatively estimated so as to support a more accurate high level qualitative assessment (Purewal, Yang, and Grigg 2009).

Bayesian Belief Networks represent another tool for probabilistically addressing the uncertainty in system interactions by incorporating conditional probabilities between system states. These networks are effective at modeling scenarios where some information is known, but new incoming data is uncertain (Charles River Analytics 2004). By combining Bayesian probability with graph theory, sensitivities in system designs can be assessed (Doguc and Kardes 2009). Since Bayesian networks utilize joint probabilities, causal relationships can be explicitly considered.

In the same way that fuzzy assessors and heuristics rely on professional judgment, one must question the basis for the probabilities assigned to the decision nodes of the system representations produced by these methods. If these probabilities are developed too subjectively, the assessment again suffers from lack of defensibility even though the mechanics of the assessment algorithm are repeatable and mathematically rigorous.

2.2.6. Analytical Assessment. Analytical relationships form the basis of the *Rational* approach to traditional architecture generation. Once formulated, they are inherently objective and repeatable. Analytical relationships in the form of constraint equations are the underlying element in most SysML parametric diagram types. Their concise closed form provides useful indicators of performance, constraint relationships, and resource consumption. Cause and effect relationships are readily captured in analytical form. The principle weaknesses of most analytical relationships are that they are frequently too simplistic and generally intolerant of ambiguity.

Some relationships are exact by virtue of their definition. This is true of geometric relationships such as determining the area of a polygon. The truncation of irrational quantities such as π and *e*, produce little impact on an assessment. The analytical assessments of concern are those that attempt to describe macro-level phenomena too simplistically. This can occur in at least two ways:

1. Underlying principles consisting of complex quantities and positional, angular, temporal, or other such dependencies are simplified to time-invariant scalar form.
2. Coupling variables between system components and between the system and its environment are omitted or lost due to improper model decomposition.

While these analytical relationships remain useful as first order approximations, they may also conceal design sensitivities due to the exclusion or over-simplification of important real world coupling effects. It can even be suggested that many emergent behaviors are attributable to a lack of understanding or consideration in the complexity of the relationships between system elements. System and system of systems (SoS) architects are motivated to understand and control these emergent behaviors insofar as they contribute constructively or destructively to the intended system performance.

Integration impacts on system performance and the analytical relationships used to characterize them are presented in Dauby and Dagli (2009a). The example presented is that of antenna gain patterns produced by radio frequency (RF) transmit and receive systems. System or SoS architects use analytical relationships to estimate wireless link distance performance indicators. Commonly used relationships include the free space and two-ray path loss equations provide in 3 and 4 respectively.

$$P_r = \frac{G_t G_r P_t \lambda^2}{(4\pi d)^2} \tag{3}$$

$$P_r = \frac{G_t G_r P_t h_t^2 h_r^2}{d^4} \tag{4}$$

It should be noted that G_t and G_r are the transmit and receive antenna gains respectively. Equations 3 and 4 imply that these values are represented by a single value. In reality, there is a strong angular dependence to the far field gain value. Integration impacts, as shown in Figure 2.1, produce unexpected angular dependencies in otherwise predicable antenna patterns. The variation by spatial angle should be identified using an index on the angularly dependent variables, but this is typically not done. The issue just described is an example of the first type of simplification identified above.

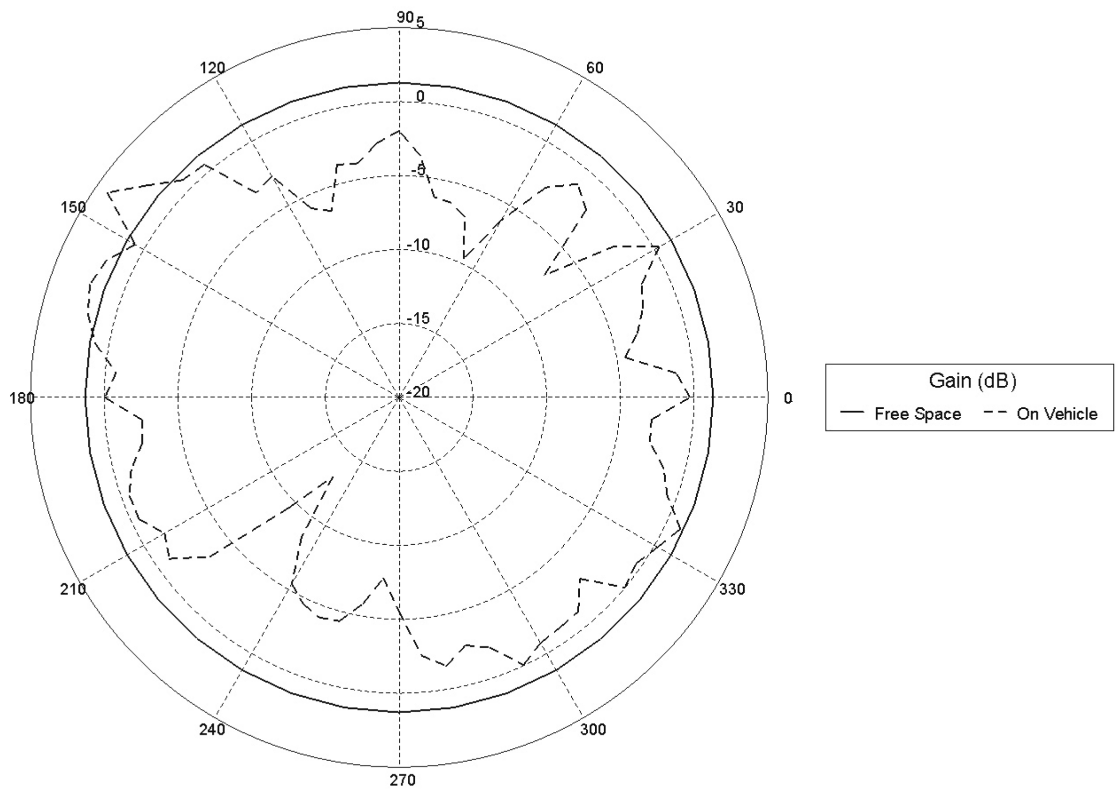


Figure 2.1. Comparison of Free Space to Installed Antenna Gain
(Dauby and Dagli 2009a)

It is important to consider whether the gain values being used for the assessment are those of free space performance or installed performance. The gain pattern of an antenna can change by many decibels when integrated with other system components. The magnitude of the impact depends upon the installation environment, operating frequency, and antenna type. This omission of coupling variables exemplifies the second type of simplification identified above and is also illustrated in the comparison between free space and on vehicle antenna gain shown in Figure 2.1.

2.2.7. Multi-Attribute Tradespace Exploration. Multi-Attribute Tradespace Exploration (MATE) concludes this section by bringing the discussion back to the multiple criteria aspect of systems architecting and highlighting the difficulty separating architecture generation and assessment. MATE with concurrent design (MATE-CON) is a conceptual architecture generation and assessment methodology that combines decision theory and model based design (Ross, Hastings, and Warmkessel 2004). MATE combines stakeholder preferences using elements of Multi-Attribute Utility Theory. Architectures are generated and parameterized in terms of underlying design variables. These parameters are varied and the resulting architecture design vector is assessed through analysis (Ross 2006). This assessment provides an estimation of performance which is then related to both utility and cost for the concept.

MATE approaches visually present a multitude of alternatives in an architecture search space in terms of stakeholder utility versus lifecycle cost. From a plot of these solution points, a Pareto Front is formed by the set of architecture solutions whose objective functions cannot be improved without reducing at least one other objective function (Ross 2006). This solution space allows stakeholders to consider the cost of utility.

As the name indicates, MATE provides an approach for combining multiple criteria in an architecture assessment. The approach also establishes a relationship between stakeholder preferences and underlying technical design variables to form an architecture design vector. This allows the employment of analytical techniques to determine the performance of an architecture and relate that performance to high-level utility. In this way, MATE establishes technical decomposition, but is susceptible to the same analytical fragility discussed in the previous section. The realism and completeness of the parametric equations ultimately determine the accuracy of the performance predictions that are generated. If the MATE approach is viewed as a framework for architecture generation and assessment, then it would seem feasible that new assessment methodologies would serve to augment the approach by increasing the ability to study coupled system performance.

2.3. EMERGENCE IN SYSTEMS ARCHITECTING

Philosophers have long debated the meaning and significance of emergence, but the subject has seen renewed interest given its implications in systems engineering. High level discussions about emergence lament the fact that systems engineers have yet to find a way to rigorously describe and predict the phenomenon. Holland (1998, p.3) states that "despite its ubiquity and importance, emergence is an enigmatic, recondite topic, more wondered at than analyzed". This problem is compounded by a lack of agreement on what it is that actually "emerges". Once the emergent entity has been identified, one must be curious about the source of such emergence.

White (2007) offers a definition of emergence as "something unexpected in the collective behavior of an entity within its environment, not attributable to any subset of its parts". White also provides several definitions and perspectives offered by other authors. A common theme is the notion that emergent behavior is either very difficult, or impossible, to predict prior to observation. According to Hitchins (2003, p.24), emergence is the "phenomenon of properties, capabilities, and behaviors evident in the whole system that are not exclusively ascribable to any of its parts". Instead, it is suggested that "emergence is brought about by interactions between the parts of a system" (p.25). For systems analysis employing both practical reductionist and holistic thinking, this definition seems appropriate. It accepts the existence of emergence while acknowledging a practical source for those properties.

Bar-Yam (2004) presents four types of emergent behavior. Type 0 emergence represents the assumption of severability that many system architects make when assessing integrated performance factors. Properties of the whole system are inferred from the properties of the individual parts in isolation. Also called weak emergence, Type 1 represents a practical limitation of understanding overwhelmingly large datasets arising from high resolution analysis. Collective behaviors are described in the data, but are difficult to identify without the aid of statistical techniques. A form of strong emergence, Type 2 emergence arises from the set of all possible system states, not just a single state. It should be noted that the set of all possible system states is not equal to the sum of all possible component states. Because of system level constraints and influences, not all combinations of component states are allowed within the system. For this reason, only coupled components will exhibit Type 2 emergence. Type 3 emergence results from the relationship between a system and its operational environment. Properties of a system or component may not be obvious until the complementary properties are identified in the environment. Coupling is therefore an important consideration when attempting to accurately describe system attributes.

2.4. CONCLUSIONS AND RECOMMENDATIONS

Based on this literature search, it is suggested that traditional architecture assessment and decision methods are useful and well intentioned, but may result in subjective, oversimplified, or indefensible estimations of system performance. This view of analytical methods, in particular, is supported by (Maier and Rechtin 2002, 185) who suggests that “rational and analytical methods produce a gloss of certainty, but often hide highly subjective choices”. Regardless of which architecture generation or assessment mechanism is used, some level of uncertainty will remain. The source of uncertainty stems from ambiguity in the architecture description and the fact that assessments are based on estimated values. Sometimes the architecture models themselves introduce new sources of uncertainty (Bartholomaus and Dagli 2008). The ambiguity in the architecture definition precludes a statistical description of the uncertainty in terms of confidence intervals $\pm \delta x$, but one can intuitively understand the concept of being ‘more certain’ as a result of having more reliable indicators.

A successful architecture plays a principle role in the system integration and operational phases. Thus, there is a need for a realistic, objective, repeatable, and defensible assessment mechanism that remains tolerant of design ambiguity. Many existing architecture generation and assessment approaches have been surveyed and present herein.

A gap appears in the ability of existing architecture assessment techniques to provide realistic estimation of coupled system performance in a manner that is tolerant of ambiguity. Ambiguity is an inherent characteristic of systems architecting and must therefore be accommodated. Since emergence is tied to the collective behavior of parts and appears to only manifest itself in the context of the system and its environment, it is necessary to perform an assessment of coupled system elements rather than individual elements. An assessment technique that severs the coupling between interacting elements potentially precludes the identification and characterization of the sources of emergence. Thus an additional goal of the research is to enable the characterization of coupled architecture performance.

An improved architecture assessment methodology will likely draw upon the strengths of existing techniques while compensating for the identified gaps. The focus of this research was to identify a set of tools to support this need for new assessment MPTs and to integrate those tools into a cohesive methodology for achieving the stated goals. The result is the Canonical Decomposition Fuzzy Comparative assessment methodology. The theory and underlying framework of this approach are presented in the next section.

3. DEVELOPING THE METHODOLOGY: THE CANONICAL DECOMPOSITION FUZZY COMPARATIVE APPROACH

Through successive iterations, the architecting process reduces the ambiguity in a system description while maintaining a balance between competing measures of system success. The process begins by eliciting a full description of user needs, system scope, and socio-technical constraints. The first substantial iteration results in a functional architecture that captures the stated needs and goals within the operating boundaries of the system definition. Subsequent iterations add more clarity to the design details, and in doing so, reduce the architecture search space until a number of candidate physical architectures are produced. This process is illustrated in Figure 3.1 below. The physical architecture may identify technology genres to incorporate, but remains ambiguous as to the final form of the production artifact. The need for MPTs to improve assessment fidelity and objectivity while remaining tolerant of design ambiguity exists at all levels of the architecting process. The research presented herein focused specifically on assessment at the physical architecture level. This section introduces the foundational elements of a new architecture assessment methodology resulting from this research.

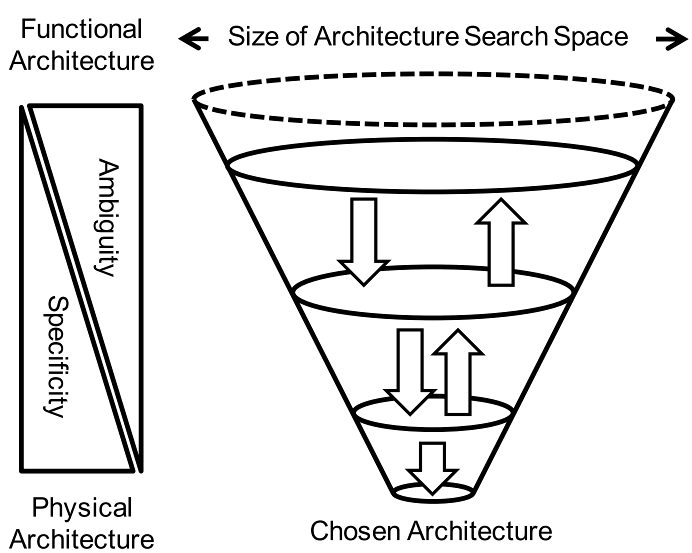


Figure 3.1. Ambiguity Reductions in the Architecture Search Process

The search for new MPTs for system architecting began by studying and understanding the breadth of tools available to the specialty engineering domains. There are numerous tools, techniques, algorithms, and analysis approaches available in these communities. Leveraging elements from these existing tools has many advantages. The existing tools are accepted and rigorously validated within their specific disciplines. Usage of common tools reduces the need for validation of the tools themselves and establishes an initial level of buy-in from stakeholders. Common tools also facilitate communication between the systems engineering community and the specialty engineering communities. It was discovered that existing tools had the potential to address many of the assessment needs presented in Section 2 if one altered the way they were used.

The proposed methodology, the Canonical Decomposition Fuzzy Comparative approach, represents the fusion of four underlying elements: extensible modeling, canonical design primitives, comparative analysis, and fuzzy inference as shown in Figure 3.2. These elements are used in the context of a technical domain to offer a performance assessment of architectural sub-attributes. This is accomplished by applying a system architecting mentality, most notably an appreciation for ambiguity, to the usage of specialty domain tools. The architecture sub-attributes can be evaluated and compared individually or combined into overall architecture assessments.

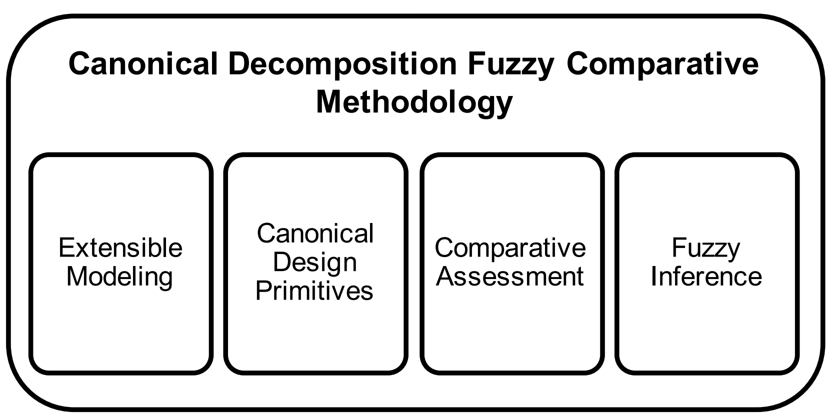


Figure 3.2. Elements of the Canonical Decomposition Fuzzy Comparative Methodology

3.1. EXTENSIBLE MODELING

Multiresolution modeling is the ability to model system attributes and behaviors at different levels of detail depending upon the amount of available information and the needs of the analysis. Multiresolution modeling can be accomplished through a family of models and is strongly promoted by the Committee on Modeling and Simulation for Defense Transformation (National Research Council 2006). Within a family of models there exists a hierarchy of detail resolution levels as shown in Figure 3.3. A representation of the relationship between models implies the opportunity for an interface through which to exchange model data. Extensible modeling is the means by which the exchange of data between model resolution levels is facilitated.

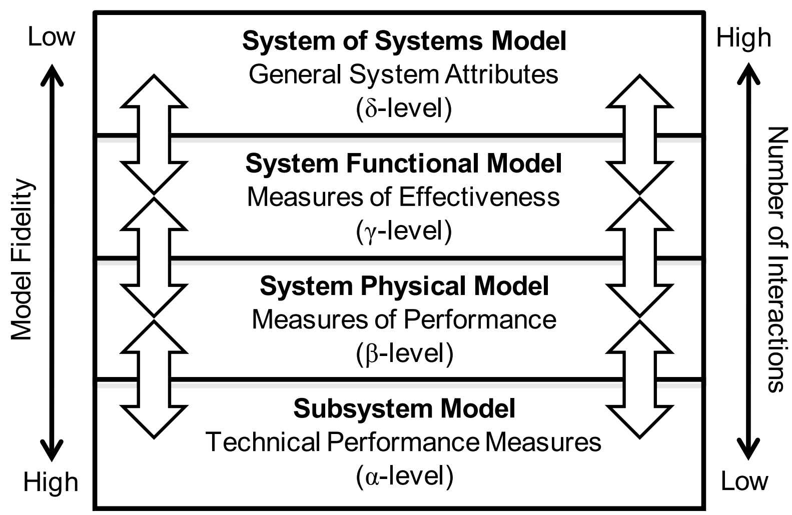


Figure 3.3. Extensible Modeling – Resolution Hierarchy and Data Exchange

If one applies the same decomposition and integration relationships found in the systems engineering process to the model resolution levels, one can establish traceable linkages between them. The systems engineering guidance suggests that measures of effectiveness (MOEs) are decomposed into system measures of performance (MOPs) which are subsequently supported by technical performance measures (TPMs) (U.S.

Department of Defense 2001). This is only possible in architecture assessment if one understands the equations and algorithms used at each level of system modeling. By identifying input parameters that represent an aggregation of many underlying conditions, one can establish an opportunity for decomposition to a lower level model. The lower level model is generally narrower in scope, but is able to more accurately estimate the quantity in question. Once calculated, the output can be integrated into the higher level model as a value approximation of increased fidelity and objectivity.

Consider an example relating the decomposition and integration of parameter values for a campaign model analyzing the effects of a radar jamming system (Adamy 2001; Stimson 1998). The campaign model assumptions might assign effectiveness in the form of a percentage or probability with respect to the jammer's ability to defeat a specific hostile radar. Given the complexity of system interactions, unpredictable environmental variables, and large number of actors, campaign modeling is usually statistical in nature. Data farming is a modeling technique where thousands of model runs are executed using the same tool, but the input parameters are varied from run to run (National Research Council 2006). The resulting composite data set contains emergent trends resulting from the complex model interactions. Realistically bounding the intervals and defining probability distributions for the input parameters can allow the campaign model to produce more meaningful data in fewer iterations.

To support a macro-level attribute such as percent effectiveness, one should decompose the jammer-to-radar engagement scenario into a functional model that assesses their respective measures of effectiveness. One will find that jamming effectiveness is directly related to the jamming-to-signal (J/S) power ratio that is achieved at the radar receiver. Based on the J/S ratio, and a number of other system variables including system mode and countermeasure protections, one can estimate the probability or percent effectiveness achieved in a specific engagement. To substantiate the estimation of a scenario-specific J/S ratio, one can decompose the functional system model into a physical system model to assess measures of performance such as effective isotropic radiated power and minimum receive signal level. These quantities can be linked via RF path loss and transmission equations to determine jammer and radar signal power levels, the ratio of which is the J/S number in question. Finally, by decomposing the system model into subsystem models, one is able to assess technical performance measures such as antenna gain, which contribute directly to the effective isotropic radiated power.

A different level of resolution, and likely a different model, is needed to estimate each of the aforementioned values. The concept of extensibility connects each resolution level by defining the decomposition and integration of assumptions and dependencies needed to support multiresolution modeling. By defining the relationship between data elements in this way, it minimizes the need to recalculate those quantities at each resolution level. Further treatment of extensible modeling can be found in Dauby and Dagli (2009a).

To facilitate a more precise explanation of model extensibility, the following mathematical definition is proposed. First, reconsider the modeling hierarchy shown in Figure 3.3. The implication is that models can be grouped according to the level of resolution involved in the analysis. Expression of model grouping through hierarchy implies an interface or dependency between the levels. In general, extensibility suggests that a model at resolution level *n* can produce more accurate results by incorporating data from a model at resolution level *n*-1.

DeLaurentis and Callaway (2004) propose a lexicon to describe hierarchy in a system of systems. The emphasis of the lexicon is on levels of integration and can be similarly helpful in understanding the relationship between resolution levels in extensible modeling. The most basic elements in the hierarchy are the alpha (α) elements. For a given scope of analysis, these represent fundamental quantities that are not decomposed further. Beta (β) elements represent an aggregation relationship between α-level variables. Similarly, Gamma (γ) elements are a collection of β-level quantities and Delta (δ) elements are comprised of γ-level relationships. For the extensible modeling framework in the CDFC approach, the resolution levels represent more than collections of constituent level variables. It is proposed that they are functions of those variables as shown in 5-8, where A_S is a system attribute assessment.

$$A_S = f(\delta_1, \delta_2, \ldots \delta_n) \tag{5}$$

$$\delta_i = f(\gamma_1, \gamma_2, \ldots \gamma_n) \tag{6}$$

$$\gamma_i = f(\beta_1, \beta_2, ... \beta_n) \tag{7}$$

$$\beta_i = f(\alpha_1, \alpha_2, ... \alpha_n) \tag{8}$$

3.2. CANONICAL DESIGN PRIMITIVES

By establishing an extensible exchange of model data between different resolution levels, one can see how detailed modeling tools used at the subsystem design level can contribute to the goal of more accurately modeling overall system, or system of system, behaviors. Finite element analysis, method of moments, finite difference time domain, and finite integration techniques provide a powerful means of accurately predicting salient technical characteristics in subsystem components. Numerous commercial, government, and academic analysis codes are available and have been scrutinized by large user communities to ensure the techniques are validated and stable. Many of these tools are widely accepted and there exists a common understanding of their functionality, limitations, and outputs. By leveraging these available techniques, the systems architecting community receives the immediate benefits of a diversified analysis toolset without the need to recreate or validate it.

The primary obstacle to widespread use of these tools and techniques is the reality that a system architecture description is ambiguous whereas these tools require a specific physical description of the artifact to be analyzed. Traditionally, these tools are used in the final design and optimization of strictly specified physical artifacts during the design and build phase of system development. However, there is a way to merge the ambiguity of physical architecture descriptions with the artifact specificity needed to use these tools.

When a physical architecture is generated, one may not have detailed designs for the final artifact, but a list of candidate technologies is typically identified. Here, the word *technology* refers to a generalized class, or genre, of physical artifact. As an example, candidate technologies for providing emergency electrical power might include solar panels, batteries, or fossil fueled generators. Technologies for vehicular movement might include wheels, tracks, or walkers. Technology descriptions at this level are still vague enough to accommodate a level of ambiguity, but one will find that each of the identified technologies has several basic design equations that govern its inherent performance attributes. These basic design equations define a canonical form, or primitive, of the candidate technology. Canonical design primitives typically form the basis for final design solutions. In this way, the canonical form of a technology captures its representative behavior while remaining tolerant of ambiguity by not over-specifying the design details.

Figure 3.4 illustrates five canonical design primitives that may arise out of a physical architecture definition. The notch antenna shown in 3.4a is a structure suitable for electromagnetic modeling using method of moment or finite difference time domain. The ram air turbine in 3.4b, the airfoil in 3.4d, and the pod structure in 3.4e are representative structures appropriate for computational fluid dynamic analysis. The convective heat sink in 3.4c can be meshed and solved in a multi-domain analysis involving both heat transfer and fluid flow analysis. The canonical structures are especially representative when the physical architecture includes integration constraints such as weight and volume. The use of canonical structures and low level computational tools allows natural coupling effects to be identified between system components. Estimations of integration and design sensitivities can be incorporated in the assessment of the overall architecture description.

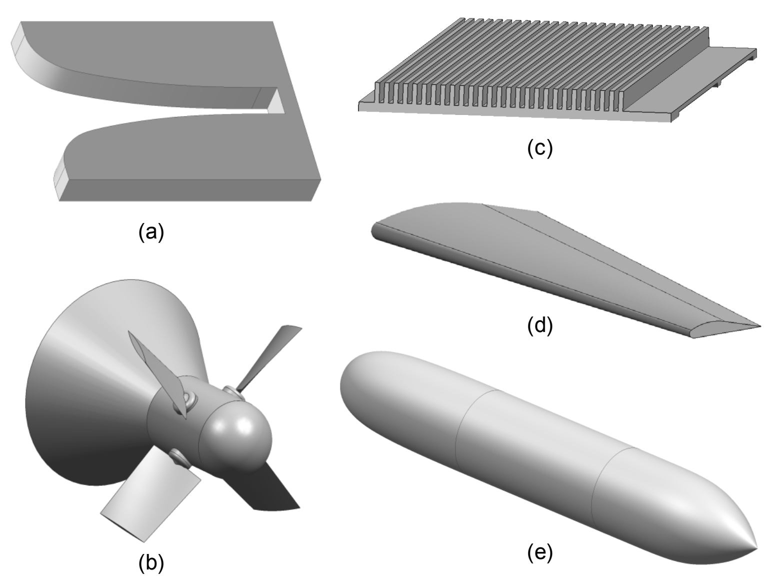


Figure 3.4. Canonical Design Primitives: (a) notch antenna, (b) ram air turbine, (c) convective heat dissipater, (d) airfoil, (e) airborne pod

It should be cautioned that the numerical techniques suggested herein produce assessment results that are representative of the specific structure under analysis. The performance of the final physical artifacts that embody the architecture being assessed will differ from the canonical structures insofar as the final hardware design deviates from them. This means that the modeled technical performance measures may not accurately represent the final system performance, but they are useful in comparing integration sensitivity between alternatives. It is suggested that the value of the model data from canonical design primitives is extracted via a comparative analysis approach.

3.3. COMPARATIVE ANALYSIS

Several quantitative sensitivity analysis methods have been developed and represent powerful ways to predict the impact of trading one variable for another. The Sensitivity Design Structure Matrix presented in Kalligeros, de Weck, Neufville and Luckins (2006) provides one solution to tracking the impact of design variable changes between variants of an architecture. However, the authors assume that the architectural concepts have already been derived and thus mathematical relationships describing the sensitivity functions may be more readily known. It is the interest of this research to investigate a way to estimate similar sensitivity functions in the early stages of the architecting process.

The comparative analysis technique examines one quantity or configuration relative to a baseline or alternative configuration. The process is similar to differential analysis used to evaluate business case alternatives and comparative statics analysis used in economics to analyze market conditions (Keat and Young 2003). In a most basic sense, the controlled change of one or more variables is used to distinguish between alternatives. In empirical settings, the design of experiments approach prescribes a number of statistical techniques that can be used to estimate the size of a response and generate confidence intervals based on a sampling of data centered on a nominal state (Montgomery 2005). In the systems architecting context, the ambiguity in system descriptions and the inability to directly observe a physical phenomenon often prohibit the accurate estimation of mean and standard deviation indicators. However, the controlled variation of input parameters can expose design sensitivities and trends in output performance. By comparing one configuration to a baseline, and observing the differences, the modeled performance data is normalized to that baseline. This approach simultaneously acknowledges the ambiguity in the predictive date while facilitating an ordinal ranking of architecture configurations based on performance.

As was mentioned in Section 3.2, the model data generated through the analysis of canonical design primitives is inextricably tied to that specific physical structure. However, each canonical primitive serves as a probe structure to facilitate a comparative analysis between configurations. One of the most obvious comparative baselines is the default assumption made in traditional architecture assessments – a severed subsystem component considered in isolation. For example, the radiation pattern of an installed antenna can be compared to the isolated free space performance of the same antenna. The power generated from a ram air turbine in a ducted airstream can be compared to the same turbine in normally incident free stream air.

Other comparative baselines can be used, especially when assessing architectures that describe permutations of the same set of technologies. An example would be comparing antenna performance between various installation locations on a vehicle. Because the same canonical structure is used in each configuration, the predicted performance of each location is normalized to the performance of the canonical design primitive itself. When used in the way, the canonical structure becomes a probe that allows for creation of integration sensitivity functions (Dauby and Dagli 2009b) and *n*-dimensional response surfaces (Dauby and Guardiola 2010). These response surfaces expose inter- and intra- system coupling between structures as a result of a particular configuration. The comparison, or trend, that emerges is typically valid in spite of the fact that the calculated magnitude of the response is a result of the primitive, not the final system hardware.

Comparisons between system architectures employing different technologies require additional consideration. Because the design primitive changes between architectures, it is not possible to normalize one response to another. It is possible however to normalize each installed configuration to the isolated performance of its respective probe structure. The comparison made in this way is that of integration sensitivity in the choice of technology. It may not be possible to say that one alternative performs *better* than another, only that one is more sensitive to integration than another.

To achieve a direct comparison between alternatives using different probe structures, additional constraint information is required. For instance, a model may indicate that a heat sink relying on free convection can dissipate x watts of thermal power whereas a phase-change spray-cool technology can dissipate y watts. This is not to say that free convection heat sink technology is incapable of dissipating y watts. Instead, it suggests that based on volumetric constraints and predicted ambient temperatures, the estimated dissipation is limited to x watts. Comparative analysis in this manner involves multivariate changes in the system model. This requires additional constraint information which implies somewhat less ambiguity in the architectural description.

Ambiguity was originally accommodated in the use of canonical design primitives instead of specific hardware definitions. Continued appreciation for ambiguity is provided through the use of comparative analysis versus a direct analysis of data points. To provide a concise crisp feedback to the architecture search process, a tool is needed to assess the multitude of comparisons that exist in the dataset while again remaining tolerant of ambiguities in the data.

3.4. FUZZY INFERENCE

Architecture ambiguity makes it extremely difficult to generate fixed quantitative estimations of system performance. Any estimation of performance is only an indicator of the expected performance. Unlike statistical sampling estimators, it is not possible to rigorously generate confidence intervals. It is also not possible to calculate an uncertainty interval from a known list of contributing error sources. Given the challenges in generating the performance estimation itself, one must expect it to be difficult to classify the results.

The CDFC methodology makes use of a fuzzy inference system to assess an architecture candidate by considering features in the comparative data set. The number and type of input variables is customizable and depends upon the architecture assessment being made. Comparative datasets in the form of n-dimensional response surfaces can contain prohibitively many data points to assess directly. Instead, some form of preconditioning may be necessary. Response surfaces features can be analyzed and extracted for FIS processing. Such features might include maximum deviation magnitude and direction, average response level, localized surface gradients, or points exceeding predefined thresholds. Figure 3.5, from Dauby and Dagli (2010) provides one example of a Mamdani type FIS suitable for use in assessing the comparative datasets generated by the CDFC methodology.

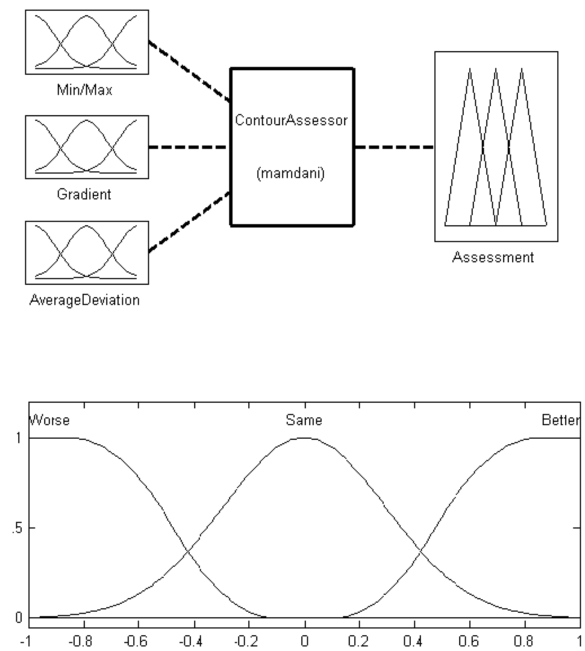


Figure 3.5. Fuzzy Inference System and Membership Functions (Dauby and Dagli 2010)

For each quantity, response levels must be mapped into fuzzy sets via the fuzzy membership functions. There is an element of subjectivity in the creation and tuning of the membership functions, and this will be addressed in Section 3.5.2. Perhaps the most challenging aspect of FIS design is the creation of a suitable, stable, rule set. These rules provide the mapping between input and output values. Output mapping is highly sensitive to membership function shape and interactions in the rule set. Much diligence is needed in creating and testing the FIS prior to use as an interpretation tool for the comparative data.

The output of the FIS represents the final output of the CDFC approach. Once a stable FIS has been created, it can be used to provide the crisp assessment value needed for feedback to the architecture search process. This is of special interest to the Smart Systems Architecting method, described in Section 2, which relies on a ranking of architecture fitness values to determine future design evolutions.

3.5. CONCLUSION

3.5.1. Summary of the CDFC Methodology. The four constituent elements of the CDFC approach have been introduced and explained from the perspective of assessing system architectures. Figure 3.6 provides a graphical summary of the CDFC assessment methodology.

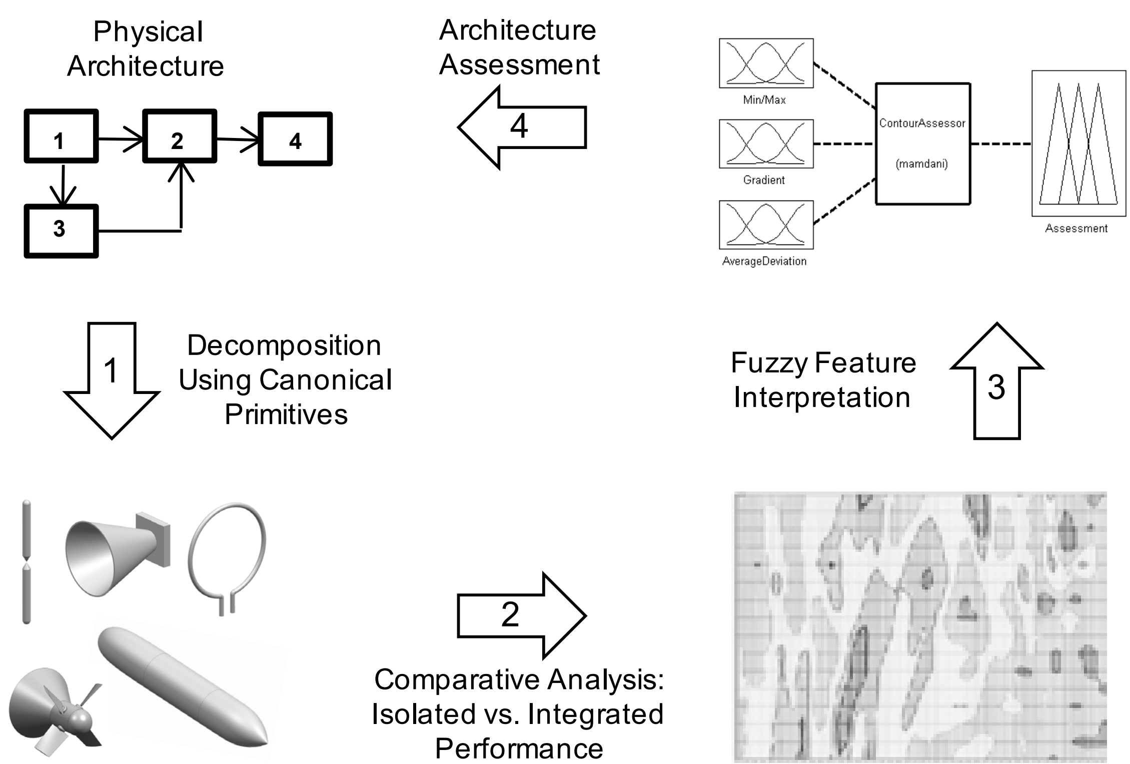


Figure 3.6. Graphical Summary of the CDFC Assessment Methodology

To begin the CDFC process, a physical architecture in the form of a block diagram is examined for regions that warrant further analysis. Candidate regions are those that require close integration between components or with the environment. Specific components within the architecture are modeled in the form of canonical design primitives using computer aided design (CAD) modeling software. These structures contain the coupling variables that yield increased fidelity in the assessment. Computational predictions of canonical structure performance are generated using detailed design techniques such as finite element analysis or method of moments. This performance is modeled for each canonical design primitive in isolation and as it is integrated with other components. The two data sets are compared by normalizing the integrated configuration data to the isolated configuration as a baseline. Normalization illuminates the integration sensitivity. Features in the *n*-dimensional comparative data are extracted and fed into a fuzzy inference system to characterize the integration sensitivity. The output of the CDFC approach is a FIS interpretation of the comparative dataset features.

3.5.2. Fulfillment of Research Goals. It is proposed that the CDFC methodology addresses the gap in existing architecture assessment techniques and successfully meets the stated research goals. The extensible modeling paradigm supports the iterative levels of refinement in systems architecting and offers a mechanism to supply detailed assessment values as inputs to other decision frameworks such as MCDM. Canonical design primitives contain coupling variables consistent with the technology genre they represent. Coupled performance modeling supports an analysis of some sources of emergence. The comparative analysis technique tolerates ambiguity by illuminating differences in estimated quantities rather than absolutes. Finally, the fuzzy inference system provides an ambiguity tolerant mechanism for considering a number of comparative data features when determining a single assessment value.

The intent of the CDFC methodology is to provide higher fidelity, more realistic, objective, repeatable and defensible assessments. It is proposed that the CDFC approach makes significant progress toward this goal, even though some elements of subjectivity remain. The creation and shaping of the membership functions is perhaps the most subjective element and is highly influenced by stakeholder inputs. While this research has focused on reducing the subjectivity in architecture assessment, at some point it becomes necessary for the stakeholders to articulate preferences in this way. By decomposing architecture attributes to individual sub-elements, it is believed that subjective decisions are more reliable and more easily substantiated. For instance, going back to the example used in Section 3.1, asking a subject matter expert to estimate percent effectiveness of an entire architecture is almost certainly more disputable than asking the expert to differentiate between antenna gain values. In fact, if discrepancies occur between how the stakeholder or subject matter experts assess an overall architecture and how they assess an underlying technical attribute, it actually serves to highlight an area of misunderstanding in the system dynamics. At the very least, CDFC enhances the stakeholders' understanding of the interplay between system elements.

In summary, the CDFC approach draws strengths from the existing architecting and assessment techniques while offering a novel contribution to increase fidelity and objectivity. The selection of architecture elements for decomposition and the overall CDFC flow is determined *heuristically*. Computational modeling brings to bear modern *analytical assessment* approaches. The fuzzy inference system leverages principles of *computational intelligence*. Finally, the extensible modeling concept allows assessments of technical performance measures to be used as inputs to higher level integrated *multiple criteria decision making* environments.

4. TESTING THE METHODOLOGY: ARCHITECTURES FOR A RADIO FREQUENCY WIRELESS NETWORK SYSTEM

This section describes the setup of an experiment to determine the efficacy of the CDFC methodology in assessing physical architecture alternatives. A general system description is provided to establish the context of the experiment. Implementation details are then provided for the specific CDFC assessment approach that was used.

4.1. SYSTEM DESCRIPTION AND KEY ATTRIBUTES

The subject of analysis was a peer-to-peer wireless network for both air and ground nodes. The system concept was intended to meet the high-level user need for increased situational awareness and tactical effectiveness through data exchange. The air and ground vehicles themselves represented the immediate integration environment for a supplemental system intended to accept, format, transmit, receive, and present sensor data via a secure wireless exchange between participating nodes. The Operational View of the system under study is shown in Figure 4.1.

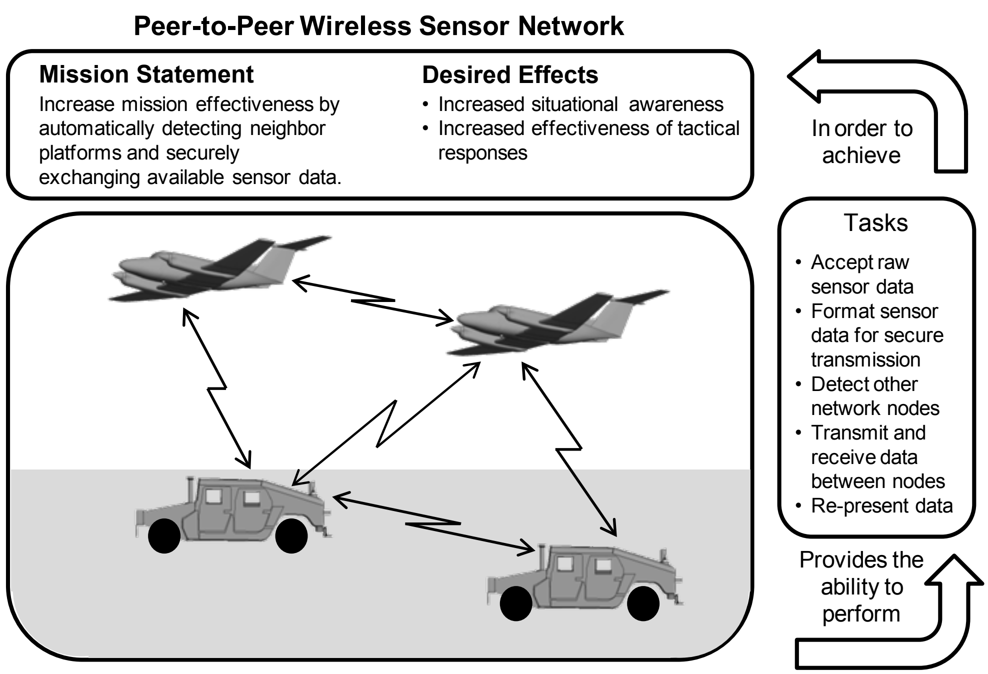


Figure 4.1. Peer-to-Peer Wireless Sensor Network Operational View (OV-1)

The SysML model of the physical architecture for a single wireless node is shown in Figure 4.2. There are six principle subsystems in the architecture definition.

1. Rx/Tx Processor – This subsystem is responsible for interfacing with data sources and consumers on the vehicle. When transmitting, this unit performs the proper formatting, encryption, and modulation of user data into RF waveforms. When receiving, the unit demodulates, decrypts, formats and presents data for user consumption.
2. Amplifiers – This subsystem is responsible for conditioning and amplifying both the transmitted and received RF waveforms.
3. Antenna – This subsystem exists at the system boundary, radiates the transmitted RF waveform, and converts received electromagnetic energy into transmission line waveforms.
4. Power – This subsystem is responsible for either producing or accepting prime electrical power and conditioning it for use in the system.
5. Cooling – This subsystem is responsible for cooling the active devices in the system.
6. Structure – This subsystem provides the structural support and mechanical environmental conditioning necessary for proper operation.

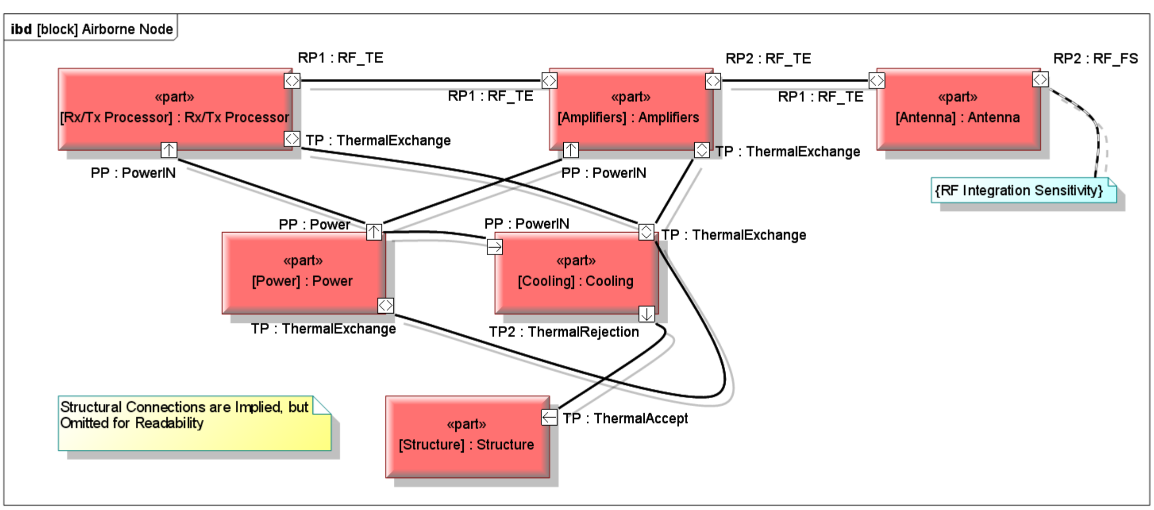


Figure 4.2. System Architecture for the Wireless Network Node

The wireless system just described is complex and involves the exchange of many α-level physical quantities. From some perspectives, the description provided is that of a system of systems. It was necessary to limit the scope of this experiment in order to determine the merits of the CDFC approach. The scope was established by accepting the following limitations.

- Nodes – This experiment considered three different physical architectures for integration onto a common airborne platform. No ground node configurations were considered.
- Attributes – A single system attribute was evaluated. Range, as defined by RF link distance, was assessed based on the contributing technical performance measure of antenna gain.

It is suggested that the limitations described do not diminish the generality of the CDFC approach. The methodology is equally suitable for the assessment of other system attributes as described in Section 3 and as proposed in Section 7.

4.2. ASSESSMENT OVERVIEW

4.2.1. Assessment Criteria – The Extensible Decomposition. Given the aforementioned wireless network system description, the system engineer may evaluate design feasibility in the architectural trade space for a δ-level attribute like *Effectiveness*. *Effectiveness* may be composed of γ-level variables such as *Connection-Capacity*, *Transfer-Rate*, and *Reachability*. It is suggested that *Link Distance* (d) is a β-level indicator of the range over which communication can occur and is thus a contributor to the measure of *Reachability*. At this point, the architect can solicit stakeholder utility to formulate fuzzy membership functions for the link distance measure.

Triangular and trapezoidal are two of the most commonly used fuzzy membership function shapes (Singh and Dagli 2010b). They are easily constructed via linear interpolation between an n-tuple definition of their slope transitions. Trapezoidal functions are defined by a quadruple, $f(x;a,b,c,d)$. Triangular functions are a special form of the trapezoidal function where b=c and therefore collapse into a single point. The triangular function is defined by a triple, $f(x;a,b,c)$.

In this analysis, the link distances were normalized to the desired nominal system range. With stakeholder input, membership functions representing High (H), Medium

(M), and Low (L) suitability were defined over an interval of 0.2-2.0 times the nominal threshold range. These were defined as L=*f*(x; 0.2,0.2,0.4,1.0), M=*f*(x; 0.5,0.8,1.2,1.5), and H=*f*(x; 1.0,2.0,2.0) as shown in Figure 4.3.

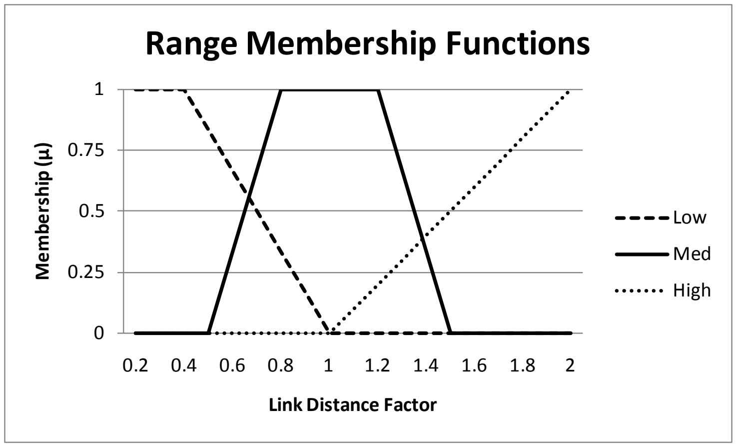


Figure 4.3. Range Membership Functions

An equation was needed to relate link distance, *d*, to technical system parameters such as transmit power, minimum receive signal level, operating frequency, and antenna gain. For the air-to-air and air-to-ground RF propagation paths, the free space path loss equation is appropriate. Equation 1 was solved for link distance, *d*, and is shown in 9.

$$d = \sqrt{\frac{G_t G_r P_t \lambda^2}{P_r (4\pi)^2}} \tag{9}$$

From this representation, one can observe that link distance is impacted by both the transmit (G_t) and receive (G_r) antenna gains. These represent technical performance measures impacted by coupling to the integration environment and are therefore good candidates for assessment via the CDFC methodology. In this example, it was assumed that transmit and receive system paths used the same antenna structure and that the two communicating nodes were identical. For the purposes of path loss analysis, the implication of these assumptions was that $G_t = G_r = G$. Equation 9 can be rearranged to solve for G as shown.

$$G = \sqrt{\frac{P_r(4\pi d)^2}{P_t\lambda^2}} \tag{10}$$

To establish a relationship between gain and distance, transmit power, minimum receive signal level, and operating frequency were held constant. The relationship between gain and distance in 10 is that of direct proportionality as shown in 11.

$$G \propto d \tag{11}$$

This relationship was used to reinterpret the fuzzy membership functions created above. The membership function shapes remain, but the set values were converted from distance to gain in decibels as shown in Figure 4.4. In this way, the membership functions for the α-level variable were analytically validated via traceability to stakeholder defined preferences for a β-level variable.

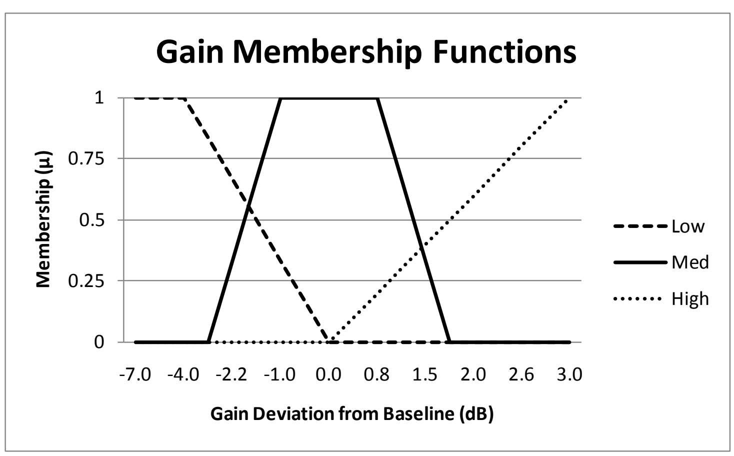


Figure 4.4. Conversion of Range Membership to Gain Membership Functions

To complete the specification of assessment criteria, one must define the angular region over which network communication must occur. This is also referred to as the field of view (FOV) for the system. For an aircraft in straight and level flight, the FOV was defined as ranging from 20° above the horizon, to 45° below the horizon, and 360° around the aircraft. This will be clarified once a standard coordinate system is defined in 4.2.3.

4.2.2. Canonical Model Set. A geometric model set was required to perform computational electromagnetic analysis and determine the integration sensitivity of the antenna performance. CAD models were produced in NX commercial modeling software (Siemens PLM). The integration environment for the analysis was a C-12 Huron, which is an all-metal military variant of the commercial King Air (US Navy – Fact File). The all-metal property meant that the entire vehicle could be modeled as Perfect Electric Conductor (PEC) which simplified the model computation. The composite propeller blades were removed for electromagnetic analysis. Figure 4.5 illustrates the C-12 geometry used as the integration environment.

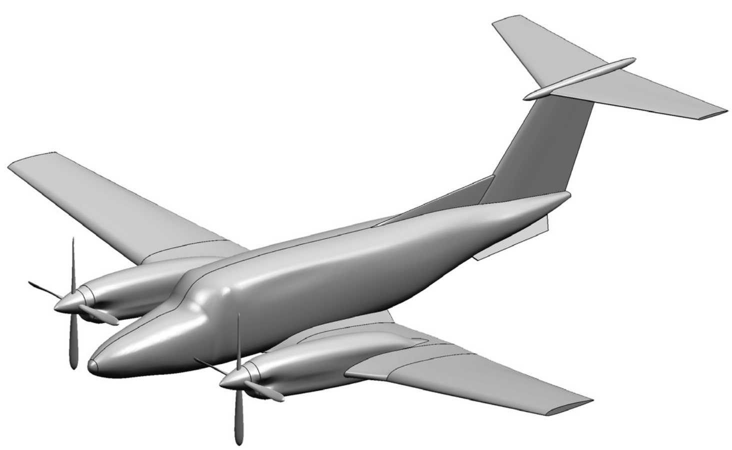


Figure 4.5. C-12 Integration Environment

A blade antenna is a common Very High Frequency (VHF) / Ultra High Frequency (UHF) communications antenna (Lo and Lee 1988). While production airborne blades contain many design features for frequency selection, lightning protection, and aerodynamic stability, the blade is closely related to the canonical monopole (Johnson 1993). This experiment was conducted at a single UHF frequency. A resonant quarter-wave monopole with electrically small ground plane was created for the 450MHz center frequency. The circular ground plane was a quarter wavelength in diameter. Figure 4.6 illustrates the canonical monopole used for this analysis.

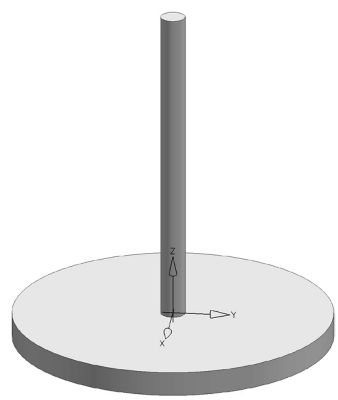


Figure 4.6. Canonical Monopole Antenna

4.2.3. Comparative Baseline. The first step to establishing a comparative baseline was to define a coordinate system that was used throughout the assessment process. All angle specifications were with respect to the standard spherical coordinate system described in Johnson (1993) and shown in Figure 4.7. It should be noted that a point consists of a radius (r), elevation (θ), and azimuth (ϕ) coordinate in the form (r, θ, ϕ). All far field antenna radiation patterns in the form of either directivity or gain are reduced from a point to a direction and are thus represented in the form (θ, ϕ). The radial component is eliminated because gain and directivity are ratios of power density in a particular direction to the average power density at the same distance from the antenna (Stutzman and Thiele 1998).

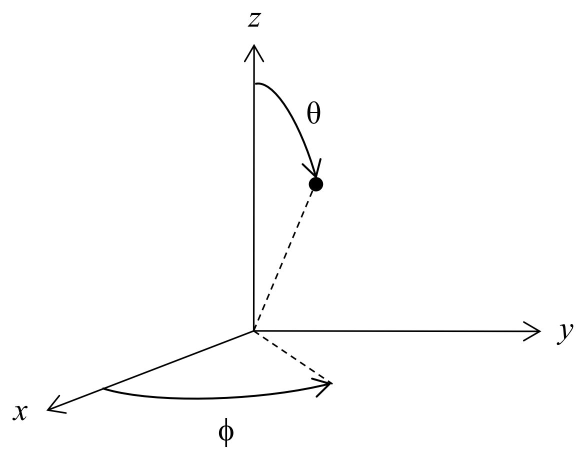


Figure 4.7. Standard Spherical Coordinate System for Global Angles

The orientation of the aircraft with respect to the global origin is shown in Figure 4.8. In section 4.2.1, the system FOV was defined as 20° above / 45° below the horizon and 360° around the aircraft. Using Figure 4.8, this translated into 70°-135° in elevation (θ) and 0°-360° in azimuth (ϕ).

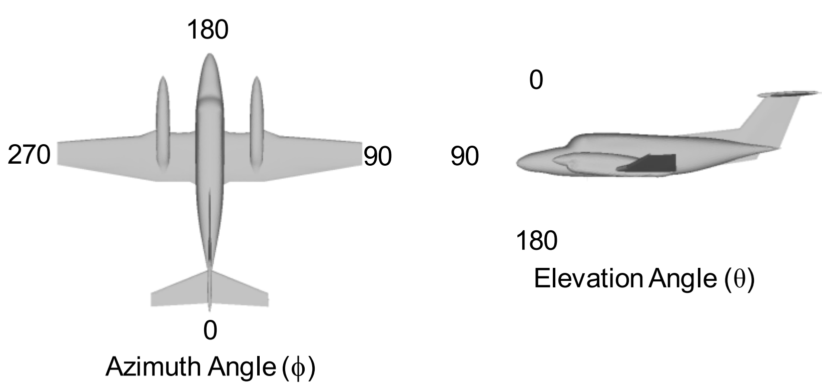


Figure 4.8. Alignment of Air Vehicle with Global Coordinate System

Computational Electromagnetic Modeling (CEM) was performed using FEKO commercial software (EM Software & Systems). The isolated monopole with electrically small ground plane was used as the comparative baseline. The baseline electric field polarization for this wireless system was vertical and the mechanical orientation was 0° theta as shown in Figure 4.6 earlier. Because the canonical monopole used an electrically small ground plane, the peak effective isotropic gain and pattern shape appeared more like those of a dipole antenna when analyzed in isolation. However, the structure contained the monopole coupling variables due to way it was integrated in its operational environment. The isolated antenna elevation gain pattern is shown in Figure 4.9.

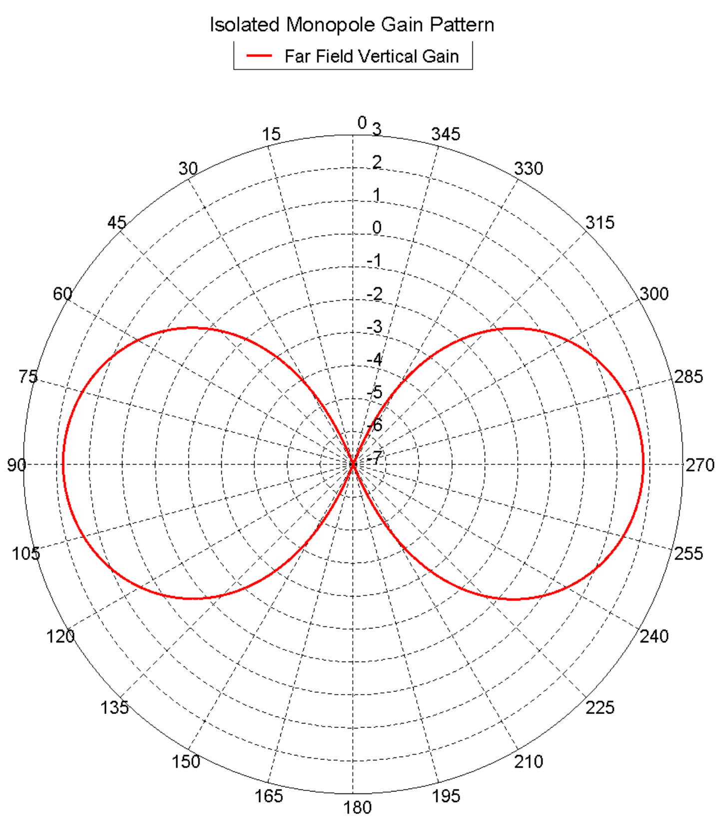


Figure 4.9. Canonical Monopole Elevation Gain Pattern

Antenna gain can be evaluated anywhere over the spherical surface defined by (θ, ϕ). This information is easily stored as a 2-dimensional matrix, but is somewhat difficult to visualize. One approach is a 2-dimensional contour plot. The contour plot is especially convenient for visualizing changes in a comparative analysis.

The contour plot was formed via a projection of the surface of a sphere onto an enclosing cylinder. The cylindrical surface was subsequently "unrolled" to show all the (θ, ϕ) on a planar surface. This process is illustrated pictorially in Figure 4.10. More information on specific projections can be found in Pearson (1990).

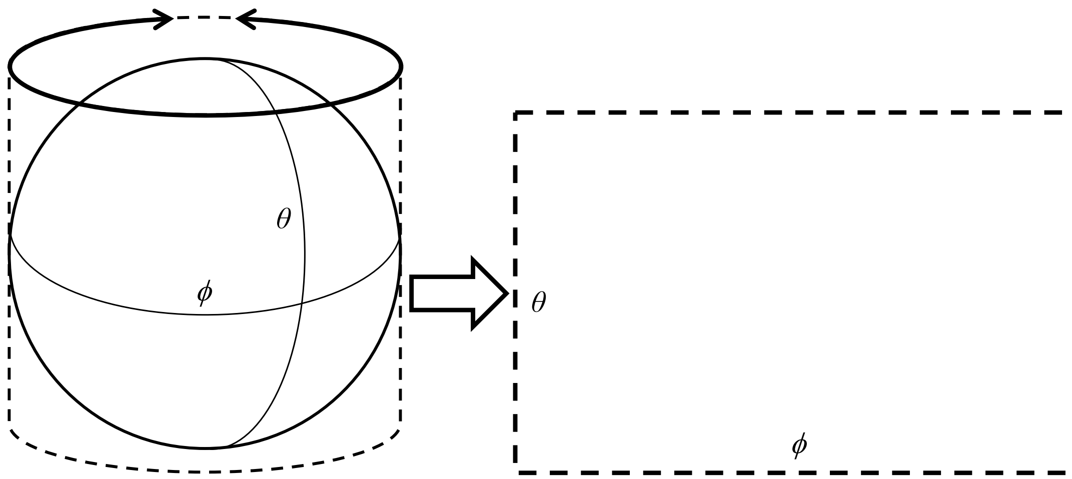


Figure 4.10. Formation of the Contour Plot from a Spherical Surface

All subsequent presentations of antenna gain data are in the form of contour plot data. Figure 4.11 presents the baseline monopole gain pattern data in contour plot form as reference. Gain values were scaled to the universe of discourse for the gain fuzzy set determined earlier.

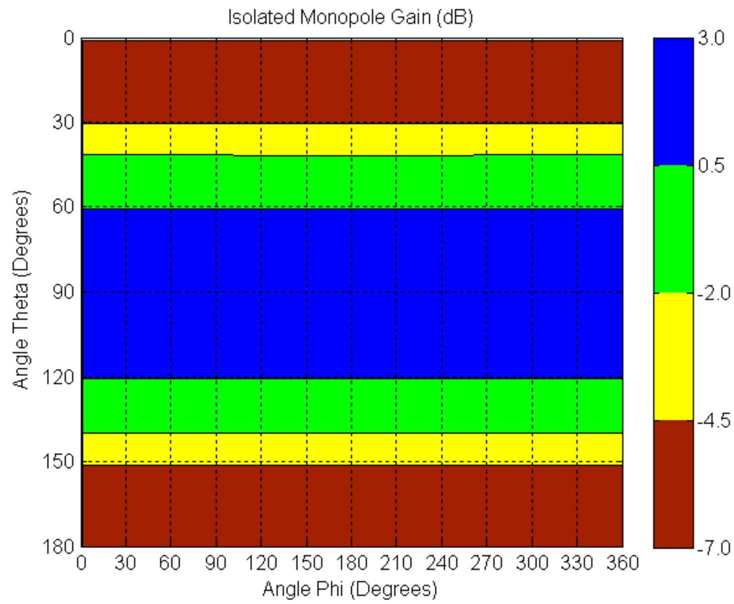


Figure 4.11. Contour Plot – Baseline Isolated Monopole Gain Pattern

4.2.4. Fuzzy Inference System. Section 4.2.1 described the decomposition of link distances to antenna gain membership sets. However, as shown in 4.2.3, there are a multitude of angles over which to assess gain. The fuzzy assessment required further interpretation and refinement in order to assess an entire FOV as specified above.

4.2.4.1 Input Variables. Antenna gain is passive which means that power is not added to the system, it is only redistributed spatially. The concept is similar to that of a lens focusing the energy that flows through it. For power density to increase at one angle, it must be reduced at another. For a system like the airborne wireless communication system with a large FOV, there will be regions of increased and decreased gain with respect to the baseline. The system architect must communicate with the stakeholder and determine which additional attributes further contribute to the assessment of performance. The number and type of input variables will be specific to the program employing the CDFC assessment approach. This research used the following input variables for the antenna FIS.

- Thresholds – The thresholds variable captured the percentage of angles in the FOV where comparative gain was less than -4dB. Upon converting range membership functions into gain deviation membership functions, one finds that a value of ≤ -4dB has full membership in the Low set. Gain deviations of this amount resulted in less than half the target link distance and were determined to provide no value to the stakeholder. The universe of discourse for this variable was [0, 1] since it represented a percentage of the FOV.

- Mean Absolute Error – Mean Absolute Error (MAE) is an indicator of the integration sensitivity in that it determines how well the baseline antenna pattern predicts the installed antenna pattern on a point-by-point basis where all points in the FOV are weighted equally. MAE was calculated using 12 where e_i is the difference between the installed and isolated predicted gain.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|e_i\right| \tag{12}$$

Large relative increases or decreases in gain (e_i) reveal an integration scenario characterized by high sensitivity. The importance of this metric is twofold. Firstly, architecture definitions are ambiguous. Installed performance predictions are imperfect, but the MAE indicates a level of volatility to be associated with either the component technology being used as the probe structure or the integration environment itself. In programmatic terms, this translates into technical performance risk. The second way MAE is helpful is in interpreting the deviation about the mean. In statistical analysis, one typically calculates a sample mean and standard deviation. This approach was not used in this research because all computational electromagnetic, fluid dynamic, or thermal models of this nature are deterministic. This results in zero spread of the data. The multitude of data points in the set correspond to gain values at different (θ, ϕ) observation angles. One observation angle cannot be used to estimate another and thus should not be considered a sample of the same quantity. In practice, one finds that this simplification is frequently employed as described next. The universe of discourse for this variable was [0, 7.5] representing the MAE in dB.

- Mean(dB) – Mean(dB) is named as such because the arithmetic mean was performed on the comparative gain values in decibels. This made it a geometric mean of the raw data. Since gain represents a multiplicative rather than an additive operation, the geometric mean was determined to be most appropriate. The interpretation of this metric is the average amount by which the baseline estimation of gain at any angle must be adjusted to account for integration effects. This metric represents a reduction in the dimensionality of the data since the mean is calculated over the entire FOV data set. The resulting scalar has no observation angle dependency and does not provide an indication of the gain pattern volatility. The author has found this sort of simplification to be common in professional practice especially as the system model works its way up the extensible hierarchy. Thus it was important to include a metric such as MAE to provide some indicator of integration sensitivity. The universe of discourse for this variable was [-7.5, 3] representing the mean gain deviation in dB.

The input conditions for the FIS were described by a three element vector consisting of the three aforementioned variables. While the mechanics of fuzzy set membership, fuzzy logic, implication, aggregation, and defuzzification are straightforward, there are some significant challenges to building a suitable FIS. Two of these include the completeness and complexity of the fuzzy rule set and nature of the aggregation method. The following strategy was employed to address these challenges.

4.2.4.2 The Fuzzy Rule Set. Like the input variables, the fuzzy rule set is specific to the program employing the CDFC approach. The decision surface created by the rule set increases in dimension with the inclusion of each input variable. Compound rules create dependencies between the variables which add features to the contour of the decision surface. Unintended features frequently result in regions of errant FIS operation. The magnitude of the FIS validation effort is directly related to the number of input conditions and the complexity of the rule set. In response to this challenge, the number of input variables was limited and the rules tended toward atomic rather than compound.

4.2.4.3 Aggregation Method. One of the most common aggregation methods is the *max* operation, but a problem can arise when the rule set consists of many atomic relationships. Each rule activates an output membership function. When the same output function is activated more than once, the maximum activation is retained rather than the combined contribution. If the rule set is linearly independent, it can be advantageous to use a *sum* aggregation method as was done for this work. This can be interpreted as counting weighted votes towards an output membership. Issues with this approach arise when there are correlations between the input variables, thus emphasizing the need for careful consideration and limitation of the input variables.

4.2.4.4 Interpreting FIS Outputs. The FIS developed for this research was created using the Fuzzy Logic Toolbox in MATLAB (Mathworks). The full syntax describing the creation of the FIS is provided in Appendix B. The output of the FIS represented the numerical assessment of the architecture candidate and thus the output of the CDFC methodology for this experiment. In order to better understand the significance of an architecture assessment value from the FIS described, the minimum, baseline, and maximum assessment values were determined. Since the input data set was comparative, an input vector of [0 0 0] represented no difference between installed gain and isolated gain. Thus the baseline score for a candidate architecture was the FIS output value for this input vector. It should be noted that the maximum FIS score is only theoretical given the necessary condition of 0 dB MAE and yet 3dB mean gain improvement. This score does not appear to be achievable in reality. Table 4.1 summarizes the input conditions and the results of the minimum, maximum, and baseline FIS scores.

Table 4.1. FIS Output Characterization

Scenario	**Input Vector**	**FIS Output**
Worst Possible Score	[1 7.5 -7.5]	0.0467
Best Possible Score	[0 0 3]	0.9530
Baseline Architecture Score	[0 0 0]	0.7010

4.3. PHYSICAL ARCHITECTURE 1

The first architecture alternative was an external pod structure mounted on the centerline of the aircraft. The pod concept supplied its own power through the use of a ram air turbine at the forward end of the pod. Cooling was provided via ducting that sent ram air across cooling fins along the inner structure of the upper mounting assembly. The antenna was positioned vertically under the platform within the pod envelop. As a self-contained concept, only data and mechanical interfaces were required between the pod and the host platform. This concept was highly portable between platforms and allowed the aircraft to be easily configured with or without the communications system. Architecture 1 is depicted in Figure 4.12.

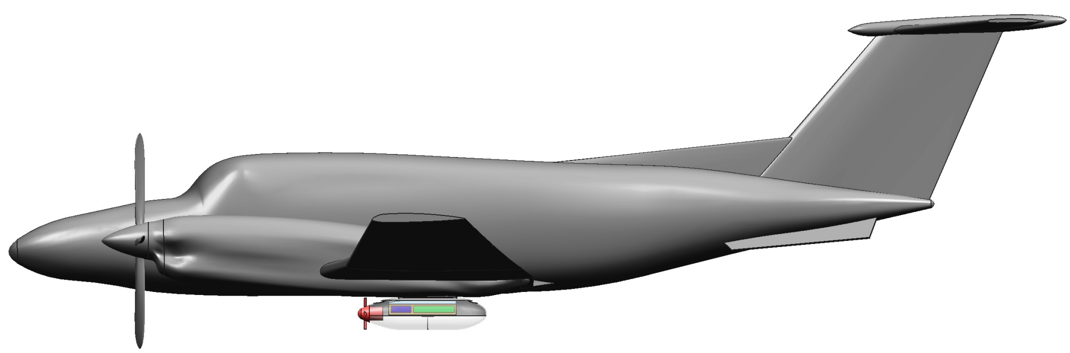


Figure 4.12. Architecture 1 – Pod-based Concept

A heuristic-based assessment of this architecture would suggest that the positioning of the antenna will provide good performance in the portion of the FOV below the horizon, but airframe blockage may limit its performance in the upper portions of the FOV. It was noted that the RAT blades in front of the pod may impact antenna gain along the (θ=90°, ϕ=180°) direction. The RAT and pod structures also increased the total aerodynamic drag of the air vehicle. This will increase fuel consumption and reduce flying qualities.

4.4. PHYSICAL ARCHITECTURE 2

The second architecture utilized an equipment bay in the rear of the aircraft to mount the electronic subassemblies. In addition to data and mechanical interfaces, this system concept relied on aircraft supplied power. The equipment bay was not climate controlled which increased the technical risk associated with cooling the electronic assemblies, but a small air duct did route a limited volume of air through the compartment. The antenna was mounted normal to the aircraft skin just outside the equipment bay. Architecture 2 is depicted in Figure 4.13.

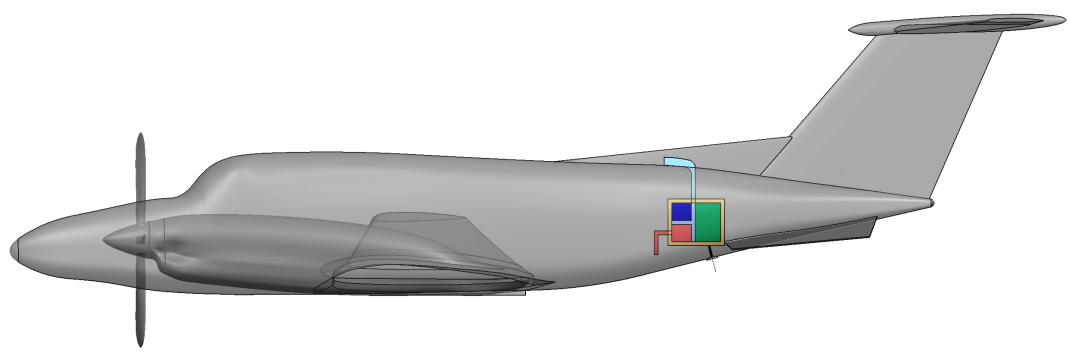


Figure 4.13. Architecture 2 – Rear Equipment Bay

Heuristically, one will notice that the orientation of the antenna is not vertical. Given that the polarization specification for the system is vertical, one would expect a polarization mismatch loss resulting in lower vertical gain for this concept. Polarization efficiency, η_p, can be computed using 13 as suggested in (Lo and Lee 1988) or utilizing the Poincaré sphere as described in (Johnson 1993), where ξ is the angle between the electric field vectors.

$$\eta_p = \cos^2 \xi \tag{13}$$

Reliance on the availability of aircraft power may be a limitation for this system concept. System reliability may also be affected by the challenging thermal integration environment. Only the antenna itself violated the outer mold line of the aircraft and therefore this alternative was likely to impact drag and flight qualities very little.

4.5. PHYSICAL ARCHITECTURE 3

The third architecture placed the electronic subsystems in an equipment rack within the aircraft cabin. The antenna was mounted normal to the aircraft skin on the dorsal surface above the other subsystem components. Cooling was provided by the climate controlled environment within the cabin. In addition to data and mechanical interfaces, this system concept relied on aircraft supplied power. Figure 4.14 shows Architecture 3.

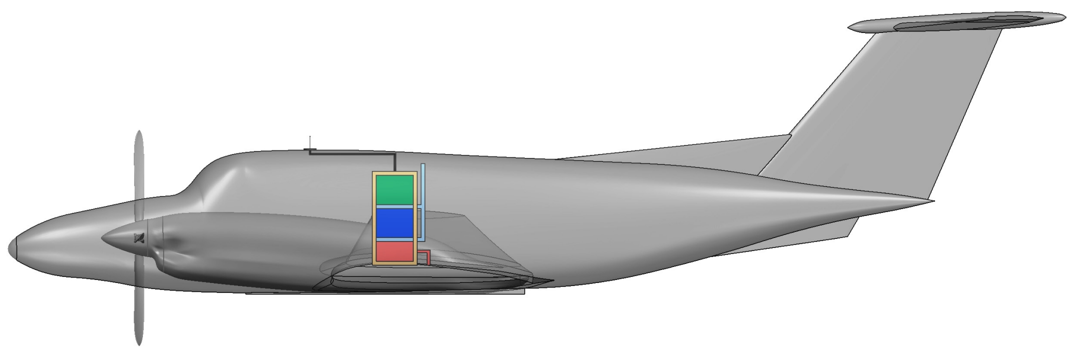


Figure 4.14. Architecture 3 – Crew Cabin Integration

Based on heuristic observations, one might suspect that a dorsal antenna will perform well for elevation angles in the upper part of the FOV, but suffer degradation in the lower portion. As with Architecture 2, the reliance on available aircraft power may be a limitation. The climate controlled cabin environment will aid the cooling of active components, but the rejection of heat into the crew compartment may offer a challenge with respect to human integration factors. Again, only the antenna altered the outer mold line of the aircraft. Drag and flight qualities will likely experience minimal impact.

5. RESULTS: ARCHITECTURE ASSESSMENTS USING THE CANONICAL DECOMPOSITION FUZZY COMPARATIVE APPROACH

5.1. PHYSICAL ARCHITECTURE 1

As described in Section 4, Architecture 1 was the external pod-based solution. One important feature of Architecture 1 was the presence of the ram air turbine on the front. For an RF technical domain expert, the ram air turbine structure was important because the turbine diameter was approximately equal to that of the pod. This meant that a single blade was nearly the same length as the monopole antenna. Electromagnetic resonance is influenced by the electrical size of a structure and thus is it was possible that the blade would reradiate incident energy. As the turbine rotated, the blades changed orientation with respect to the vertical monopole antenna. If reradiation did occur, the reflection levels would change based on blade orientation. The turbine represented an additional integration sensitivity variable and thus two sub-configurations were analyzed. The sub-configurations are illustrated in Figure 5.1 below.

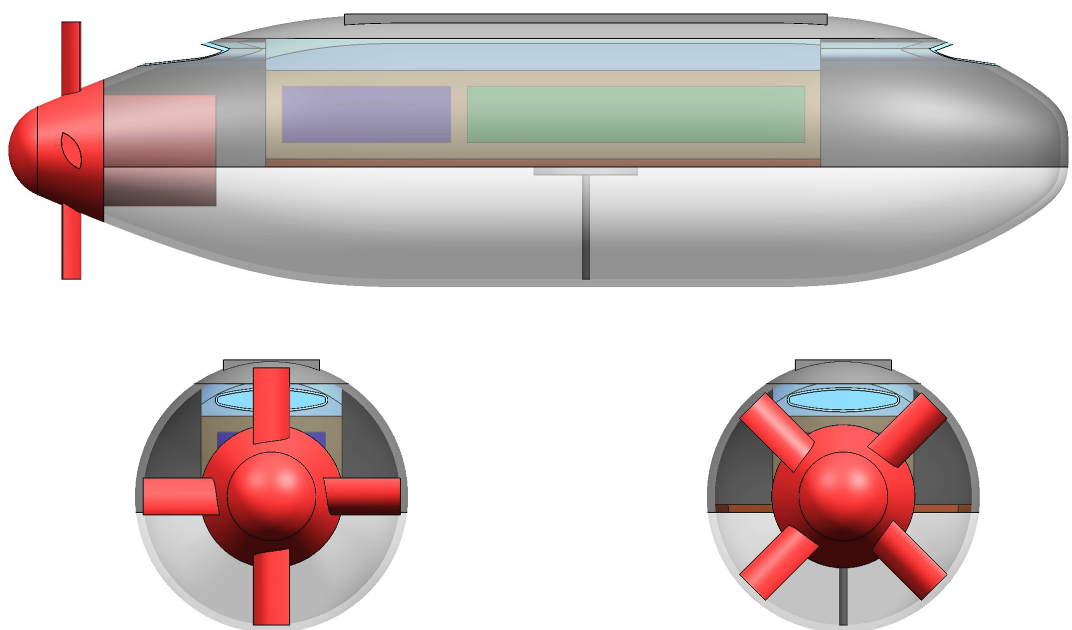


Figure 5.1. Pod Sub-Configurations: Architecture 1a (left) and Architecture 1b (right)

Architecture 1a. The contour plot illustrating the comparison between installed gain performance and the isolated free space antenna gain is shown in Figure 5.2.

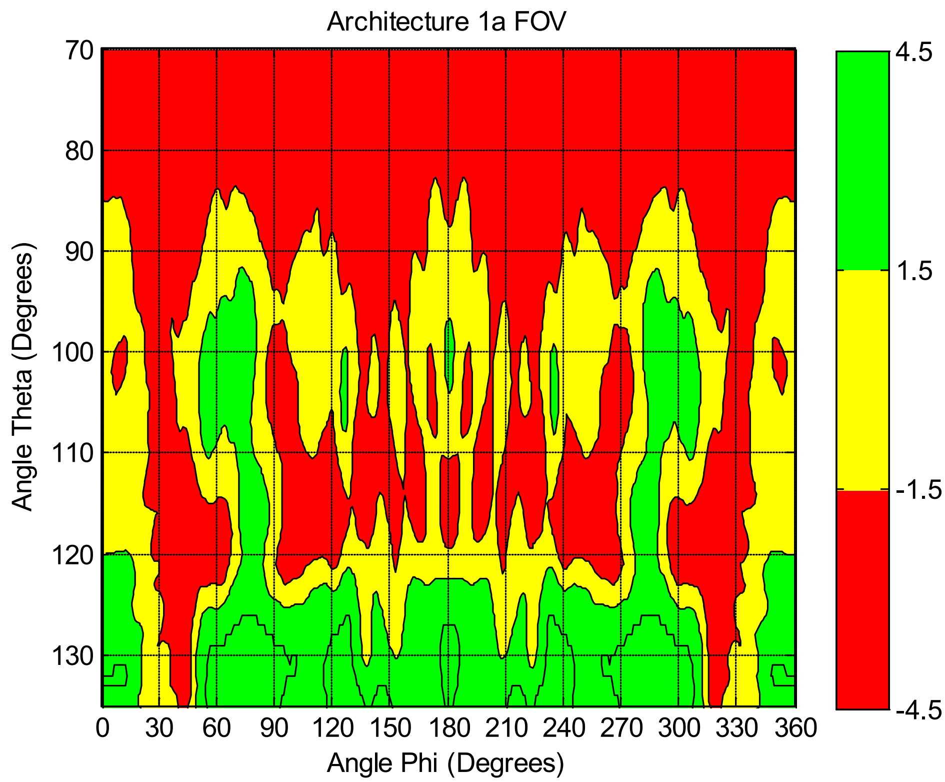


Figure 5.2. Architecture 1a Comparative Gain over System FOV

Table 5.1. Architecture 1a FIS Input Values and Resulting Assessment

	Value
Thresholds	0.2517
Mean Absolute Error	3.3164
Mean(dB)	-1.6473
Assessment	**0.3780**

Architecture 1b. The contour plot illustrating the comparison between installed gain performance and the isolated free space antenna gain is shown in Figure 5.3.

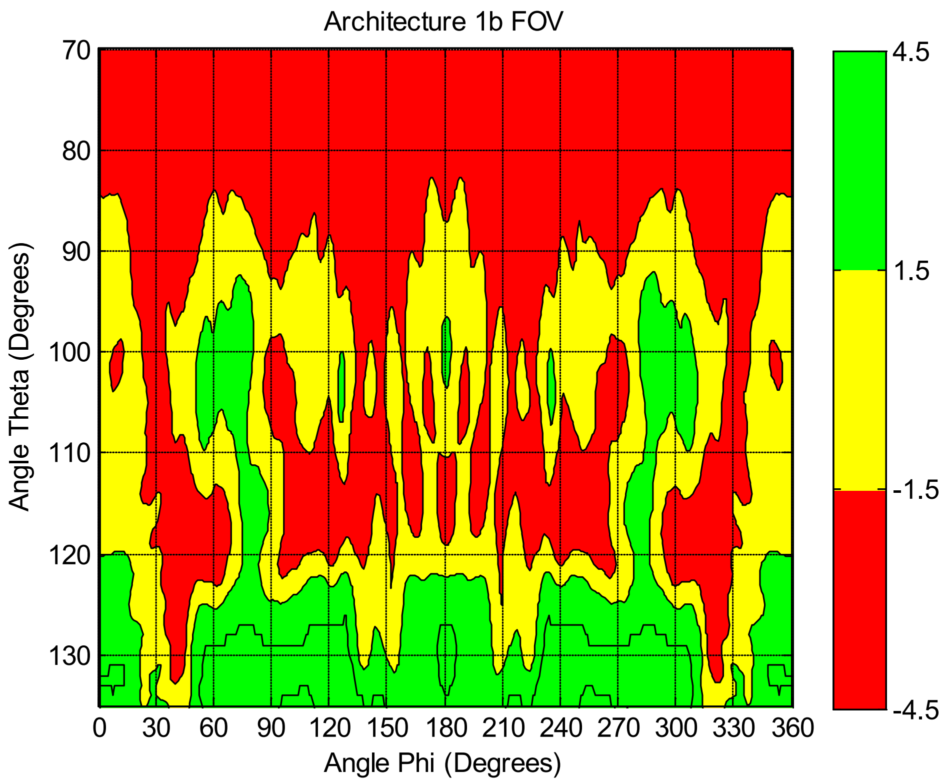


Figure 5.3. Architecture 1b Comparative Gain over System FOV

Table 5.2. Architecture 1b FIS Input Values and Resulting Assessment

	Value
Thresholds	0.2414
Mean Absolute Error	3.2530
Mean(dB)	-1.6080
Assessment	**0.3881**

Based on the results provided in Table 5.1 and Table 5.2, it was concluded that the orientation of the RAT blades was not a significant factor in the integration sensitivity of Architecture 1. In both sub-configurations, roughly 25% of the FOV was rendered useless to the stakeholder. The effects of integration resulted in an average adjustment of -1.6 dB across the FOV from the baseline assumptions about antenna gain. As an indicator of integration sensitivity, the MAE score for both RAT orientations was approximately 3.3 dB.

5.2. PHYSICAL ARCHITECTURE 2

Architecture 2 was the aft equipment bay installation with the externally mounted antenna normal to the aircraft surface. No additional sub-configuration variables were identified. The contour plot illustrating the comparison between installed gain performance and the isolated free space antenna gain is shown in Figure 5.4.

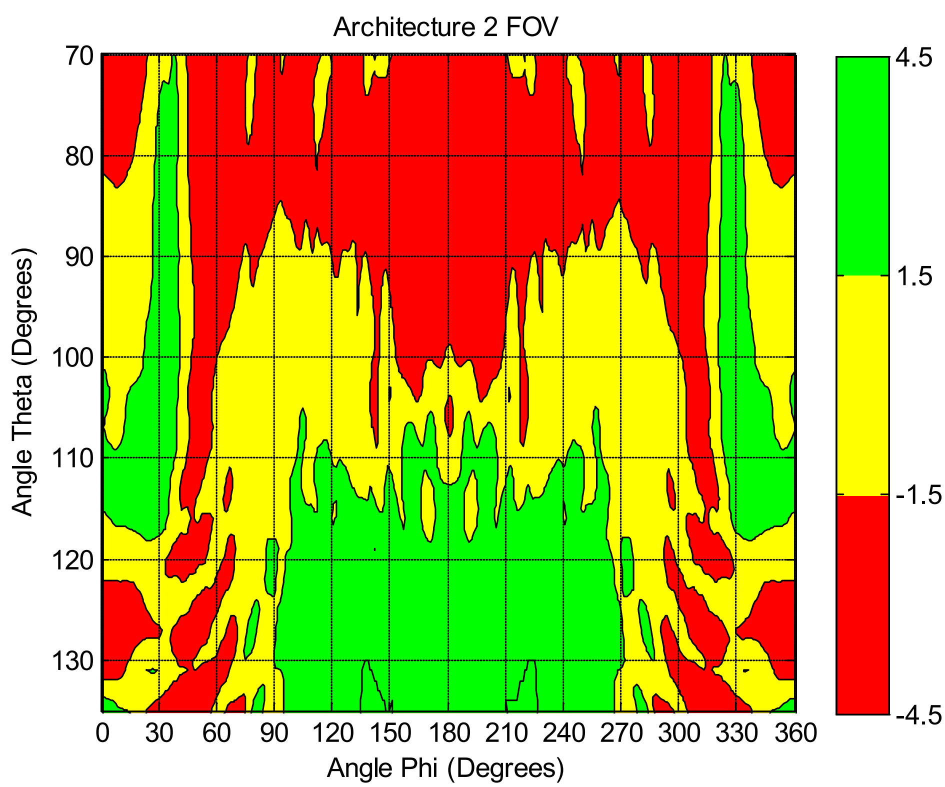


Figure 5.4. Architecture 2 Comparative Gain over System FOV

Table 5.3. Architecture 2 FIS Input Values and Resulting Assessment

	Value
Thresholds	0.1065
Mean Absolute Error	2.5080
Mean(dB)	-0.8628
Assessment	**0.4785**

It was mentioned in the description of Architecture 2 that a vertical polarization mismatch was expected as a result of the monopole being mounted normal to the angled tail surface. The angle of the tail section was measured to be approximately 15°. Using 13, it was determined that this only resulted in a reduction of vertical gain by 0.3 dB. The assessment results presented in Table 5.3 indicate that all input parameter values were improved in comparison to Architecture 1. The resulting assessment value is thus higher as expected.

5.3. PHYSICAL ARCHITECTURE 3

Architecture 3 was the crew cabin integration with externally mounted antenna normal to the aircraft surface on the top side. No additional sub-configuration variables were identified for this architecture. The contour plot illustrating the comparison between installed gain performance and the isolated free space antenna gain is shown in Figure 5.5.

From Table 5.4 one can observe that over 44% of the user-defined FOV is rendered useless as a result of the antenna location in Architecture 3. Both MAE and Mean(dB) are worse than either Architecture 1 or 2. Subsequently, the assessment for Architecture 3 is low. In fact, it should be noted that an assessment of 0.0467 is the minimum achievable output from the FIS.

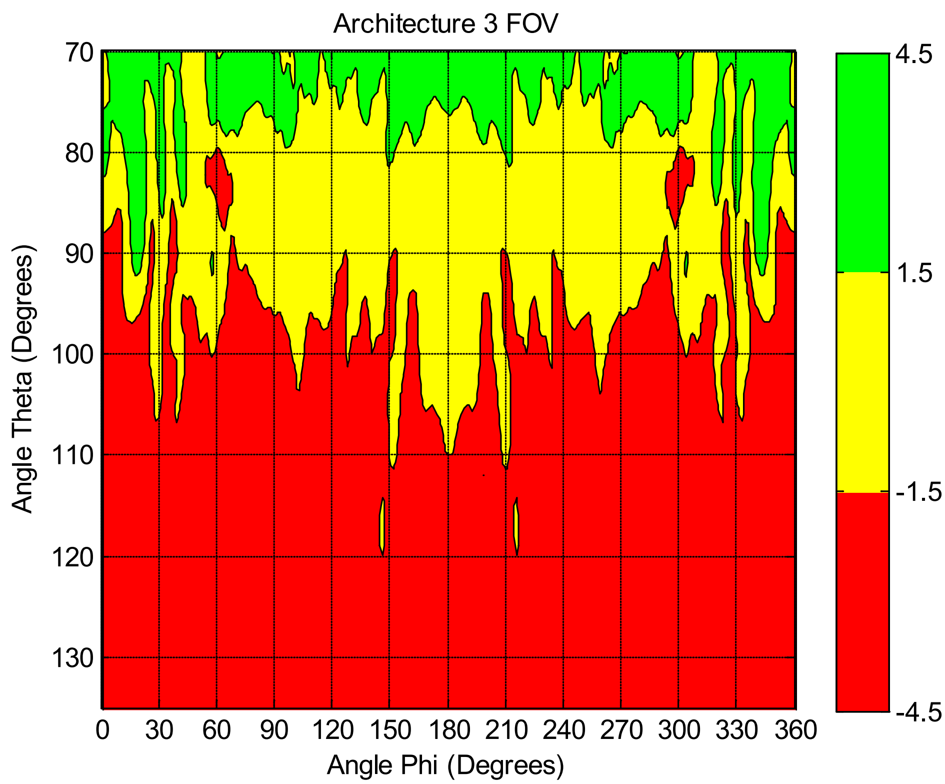


Figure 5.5. Architecture 3 Comparative Gain over System FOV

Table 5.4. Architecture 3 FIS Input Values and Resulting Assessment

	Value
Thresholds	0.4434
Mean Absolute Error	5.4933
Mean(dB)	-4.8398
Assessment	**0.0467**

5.4. SUMMARY

A summary of the raw FIS assessment scores is provided in Table 5.5 along with an interpretation of that score as normalized to the baseline score of 0.7010. Normalizing to the design baseline allows for assessments from different technical sub-domains to be combined in a manner where magnitude is consistent and units are eliminated.

Table 5.5. A Summary Comparison of Architecture Assessments

	Raw Assessment	Norm. to Baseline
Architecture 1a	0.3780	0.5392
Architecture 1b	0.3881	0.5536
Architecture 2	0.4785	0.6826
Architecture 3	0.0467	0.0666

Selecting the correct system architecture requires finding a balance between a multitude of competing quality attributes. The logical construction of this can be expressed as follows:

$$\text{Overall\_Assessment} = x_1 \text{ AND } x_2 \text{ AND } x_3 \ldots \text{ AND } x_n$$

where x_i is the assessment of an individual performance measure. Given this interpretation, a maximum score for one x_i does not guarantee a maximum overall assessment. For this reason, the assessments summarized in Table 5.5 should not be interpreted to mean that Architecture 2 is the best alternative. This experiment has identified Architecture 2 as having the most desirable installed antenna performance with respect to integration sensitivity. Other technical domains using the CDFC methodology must assess relevant attributes such as aerodynamic drag, prime power production, thermal management, and vibration sensitivity. An overall architecture assessment must consider these performance attributes as well. This idea is discussed in more detail in Section 7.

However, since the proposed logical construct for architecture assessment is a conjunction, a very poor assessment for one x_i may effectively eliminate the architecture candidate regardless of how well it scores on other assessments. It is suggested that the CDFC approach provides a defensible rationale for eliminating a system architecture from further consideration by virtue of its inability to adequately fulfill a contributing technical performance measure such as installed antenna gain.

6. COMMENTS AND CONCLUSIONS

The technical rigor of coupled computational modeling adds to the fidelity and objectivity of the CDFC approach, but also makes it resource intensive. The complexity and resource requirements of the CDFC approach therefore necessitate a limitation to the search space over which it is applied. It is suggested that this approach is most applicable at the physical architecture level.

It is important to consider that the CDFC is not an all-or-nothing methodology. Extensible modeling is a general concept that facilitates an exchange of data between different modeling resolution levels. This concept can be employed with or without the canonical decomposition and fuzzy comparative elements. For programs with technical peer-review teams, the comparative datasets may provide provocative sources of conversation and serve as a useful output without fuzzy interpretation. Finally, in addition to the composite assessor for the comparative dataset, constituent assessors can provide indicators of individual technical attributes adding to an overall awareness of the inherent characteristics of the architecture.

It can be observed from the CDFC contour plot results in Section 5 that the installed antenna performance was consistent with the heuristic evaluations offered in the architecture description in Section 4. The advantage of a CDFC assessment is that it offers a more explicit, repeatable, objective, and defensible measure. The agreement between heuristic evaluations from subject matter experts and the CDFC assessment values is critically important because this represents the fundamental mechanism for validating the assessment outputs of the methodology.

6.1. VALIDATING THE CDFC METHODOLOGY

To begin a discussion on validation, consider the following definition.

Validation is "the process of determining the degree to which a model is an accurate representation of the real-world from the perspective of the intended use of the model" (U.S. Department of Defense 2007).

When discussing validation rigor, one must keep in mind that the proposed methodology contributes to the systems architecting process. Therefore, architecture search is the intended use of any models or methods in question. Validation can and should occur at many levels within the CDFC approach, but the rigor of the validation is directly related to the specificity in the architecture to be assessed. Design ambiguity precludes the determination of model uncertainty in the form of $\pm\delta x$. To do so would require the complete decomposition of the architecture into design specific detail. In addition to being very time consuming, this is not the goal of the architecture search process. It is suggested that validation should be performed only insofar as it enhances the defensibility of the assessment. To accomplish this in the most expeditious manner, it is further suggested that individually validated sub-elements be used. This research does not specify the set of tools to use for an analysis. It is therefore not the goal of this research to validate any of the underlying computational tools, but it is the responsibility of the analyst performing the assessment to ensure that defensible techniques are selected.

6.1.1. Validating the Extensible Modeling Framework. The validation of the decomposition of system attributes into technical performance measures can be done via analysis and comparison to published information. The decomposition of a wireless network MOE, such as coverage area, into a MOP, such as link distance, has to be validated against stakeholder input. In other words, only the stakeholder can confirm the meaning of coverage area. Representations of RF path loss models can be validated against widely published literature. Deviating from path loss models published in popular wireless communication texts would required an independent validation process. Underlying technical performance measures are directly identified from the MOP equations. The selection of an underlying parameter can be validated by inspection.

6.1.2. Validating the Canonical Design Primitives. The ability of the canonical design primitive to predict installed coupling effects is directly related to the validity of its construction. Since the design primitives are selected from canonical forms, the proper underlying design equations are frequently available in domain specific texts and handbooks. For example, a multitude of primitives can be created using a reference such as the Antenna Engineering Handbook (Johnson 1993). As an authoritative source, the design equations can be considered normatively validated. Outputs from the computational predictions of canonical performance can be validated for correctness by comparing salient characteristics (e.g. gain, polarization, bandwidth) of the canonical model with the handbook predictions.

6.1.3. Validating the Integration Environment. Validating the geometric representation and material composition of the integration environment can be a challenge. For vehicles, it may be possible to access three dimensional CAD models of the structure. These models may or may not contain material property information. If such models are available, initial validation can be based on the pedigree of the source. Models supplied by original equipment manufacturers can be considered authoritative, while reverse engineered models may be somewhat less reliable. The level of rigor attributed to the determination of model validity is very problem specific.

6.1.4. Validating the Computational Tools. The experiment conducted for this research utilized two principle computational tools subject to validation scrutiny. FEKO computational electromagnetic software was used to predict performance of canonical antenna structures in isolation and in the integrated configurations. FEKO is mature software used by government, academia, and industry. As a frequency domain solver, it utilizes method of moments (MoM), multi-level fast multipole method (MLFMM), finite element method, physical optics, and geometric theory of diffraction. Each of these techniques has been subject to rigorous validation, both theoretically and empirically. Common validation techniques include comparison with analytical solutions, approximate solutions, measurement, or other computational codes (Davidson 2005). Each technique has limitations and susceptibility to error when employed beyond its intended range of operation. MoM was used to compute the isolated performance of the canonical design primitives. MLFMM was used for the vehicle level analysis in this research. This is the approach recommended by published literature and FEKO technical guidance (EM Software & Systems 2009).

6.1.5. Validating the Fuzzy Inference System. The MATLAB Fuzzy Logic Toolbox was used to create the fuzzy inference system. As a commercial product, the proper embodiment of fuzzy logic theory and computational accuracy was assumed. However, the validation of the fuzzy inference system in terms of its ability to codify stakeholder preferences is a legitimate concern. It is suggested that validation can occur in the following ways.

6.1.5.1 Membership Functions. TPM membership functions can be validated as an embodiment of stakeholder preference by analysis and traceability. If the membership functions for the high level system attributes are validated by the stakeholders, there can be no disagreement by the subject matter experts on the interpretation of decomposed membership functions.

6.1.5.2 Fuzzy Rules. The fuzzy rule set must be validated by the subject matter expert community and the stakeholders. For rule weighting, only the stakeholder can determine the relative importance of the input variables and their combinations. The rules can be validated by inspection of the decision surface and logical mapping of input to output relationships from test case scenarios.

6.1.5.3 FIS Operation. One might approach the validation of the FIS by generating a set of representative comparative data and asking the stakeholder or subject matter experts to personally assess them in terms of the factors included in the rule set. Subject matter expert assessments typically do not result in a numerical value of several significant digits in the same way that the FIS produces an output. The numerical FIS output is completely arbitrary unless it is intentionally mapped to a defined rating scale. The intended use of the FIS is to compare alternatives and thus any rating scale is useful if it is used consistently. To validate the output of the FIS, the ordinal ranking of several test data sets can be compared to the ordinal rankings determined by a community of subject matter experts. It may also be possible to validate the FIS in terms of how much better it ranks one alternative than another. Subject matter experts might, or might not, concur that alternative one is roughly twice as good as alternative two, for example. It should be noted that a discrepancy could indicate either a flaw in the FIS or an inconsistency in the stakeholders' views. Either way, the discrepancy should be resolved for the approach to be defensible.

6.1.6. Validating the Assessment. A validation of the assessment in terms of its prediction of installed performance can be rigorously performed by comparing predicted and measured results. However, measured results can only be obtained once physical artifacts are produced. The production of physical artifacts takes place after the selection of an architecture and system development has occurred. Since at that point the architecture choice has been made, this approach does little to guide the selection of the architecture. However, a posteriori validation can serve to refine the CDFC methodology for future programs. It should be noted that validation using actual system hardware must be cognizant of the differences between the canonical design primitive and final system hardware.

A validation of the architecture selection is even harder. To effectively confirm the ranking of architecture alternatives using empirical data, one must design and build each alternative. This is costly and time consuming, and furthermore, it defeats the purpose of selecting a single alternative to begin with.

6.2. SUGGESTIONS FOR USING THE CDFC APPROACH

It is acknowledged that the CDFC methodology is time-consuming and resource intensive, but for large scale complex system development, it can potentially improve the architecture search process. The INCOSE Systems Engineering Handbook recommends that large systems may warrant the development of special simulations so as to establish parameter values for system requirements (Haskins ed. 2007). The CDFC approach may not be suitable for smaller programs or those for which little time can be afforded for architecture assessment. However, the approach is suitable for large-scale defense, space, or infrastructure development programs.

Given the influence of the architecture selection on the eventual success or failure of the development process, it would be desirable to use the CDFC approach as early as possible in the architecture search process. However, there is a point at which it is not appropriate to use this approach. Guidance for use of the approach can be offered from both a theoretical and practical perspective.

6.2.1. Theoretical Guidance. Once an extensible framework has been established between high level system attributes and technical performance measures, the first critical step is the identification and selection of canonical design primitives. One cannot use a design primitive prior to the selection of a technology. These structures are produced using the basic design equations of that technology. This level of detail allows the design primitive to serve as a probe structure, illuminating integration sensitivity even without full design knowledge. Improperly selecting a canonical design primitive will yield results that are not representative of the coupling variables affecting the architecture. At best, the predicted results will be irrelevant. At worst, they will be misleading. The CDFC approach can be used from the point of technology selection onward. CDFC fidelity increases as architecture specificity increases.

6.2.2. Practical Guidance. The CDFC approach is computationally expensive and can be time consuming depending upon type of analysis being performed. Programs will dictate a schedule and budget for performing architecture search. It may be impractical to employ CDFC assessment when too many architecture alternatives are involved. Subject matter expert interpretations, intuition, and heuristics can be performed quickly and often guide system development in the right overall direction. They are generally good at reducing immense search spaces very quickly, but rarely provide an objective means to distinguish between complex architecture candidates as the solution space narrows. It is suggested that the CDFC approach be used at the point where the solution space narrows and greater fidelity is required.

The CDFC methodology is intended to expose inter- and intra-system coupling variables. The focus of research to date has been on coupling between system components and the operational environment. This coupling exists through interfaces along the system boundary. Environmental interfaces are critical because the architect typically only has control over one side of that interface. In other words, it is not often possible to change the characteristics of a particular environment. One can either change the system or pick a different environment.

It may not be practical or necessary to decompose every aspect of the system architecture. Aspects of the architecture that contain many degrees of design freedom can likely be adapted as the design evolves. A prudent approach would focus on the critical aspects of the design, such as environmental coupling, where the designers' ability to compensate for emergent properties is limited.

6.3. FINAL COMMENTS

The assessments performed via CDFC use canonical design primitives which frequently represent the starting point for actual hardware design and optimization. Also, the α-level tools used in the proposed approach are the same as those used in the detailed design disciplines. These commonalities facilitate communication and understanding between the architecting and detailed design communities.

Susceptibility to inter- and intra-system coupling and integration issues detected in the canonical models serve as early indicators of technical risk for the final design. It is proposed that the information gathered from the canonical models will lead to the development of better systems in at least three ways.

1. Identification of strong architecture candidates.
2. Identification of technical risk in an architecture. This risk can be considered when producing derived requirements for component performance. An example might be the inclusion of a performance safety factor in the specification.
3. Identification of integration issues so that development teams can anticipate challenges and address them proactively in the design.

7. FUTURE WORK

7.1. MULTI-DOMAIN ANALYSIS AND HIERARCHICAL ASSESSMENT

A system is a complex set of components, functions, interfaces and constraints. The challenge of system architecting is to balance the multitude of competing cost, schedule, performance and risk factors in order to achieve a design that is operationally suitable and effective. The CDFC assessment methodology was developed as a tool to help system designers more effectively assess architectural choices. The efficacy of the CDFC approach was established in this research and demonstrated through the work presented herein. While the work presented focused on the use of CDFC for assessing electromagnetic integration issues, it is proposed that the approach is equally suited for use in other technical domains. Examples include fluid dynamic, thermal, and mechanical assessments. Blattert (2008) demonstrates the use of computational fluid dynamics to estimate drag parameters for an airborne pod with internal ducting. This scenario is similar to Architecture 1 presented earlier and is shown in Figure 7.1.

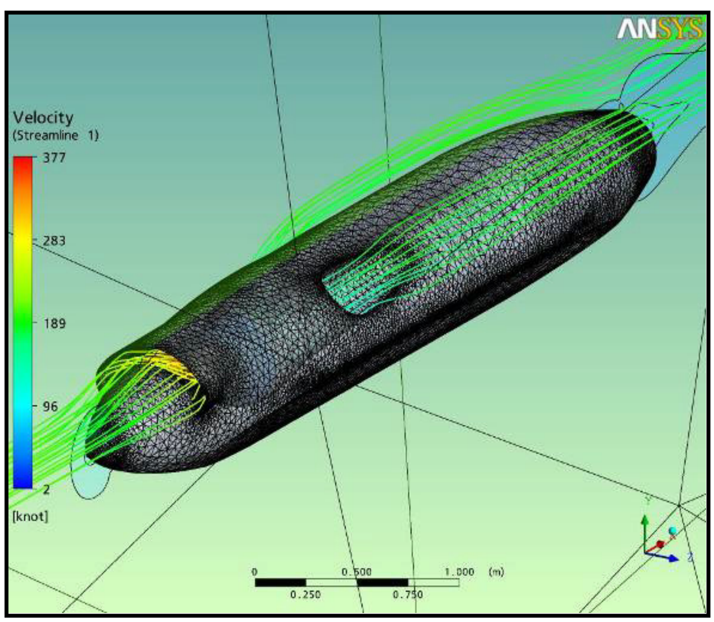


Figure 7.1. Aero Analysis for a Notional Ducted Airborne Pod (Blattert 2008)

Aero analysis would form the basis of an α-level assessment of a γ-level attribute such as vehicle range. Similarly, thermal analysis can be used to calculate the thermal profile of system components under various operating conditions. An α-level assessment of heat flow and steady-state temperature forms the basis for determining failure rate acceleration factors. These in turn contribute to a δ-level understanding of system reliability. A multi-domain example is presented in Figure 7.2. This example should not be considered a comprehensive breakdown, but serves to illustrate the multi-domain extensible concept.

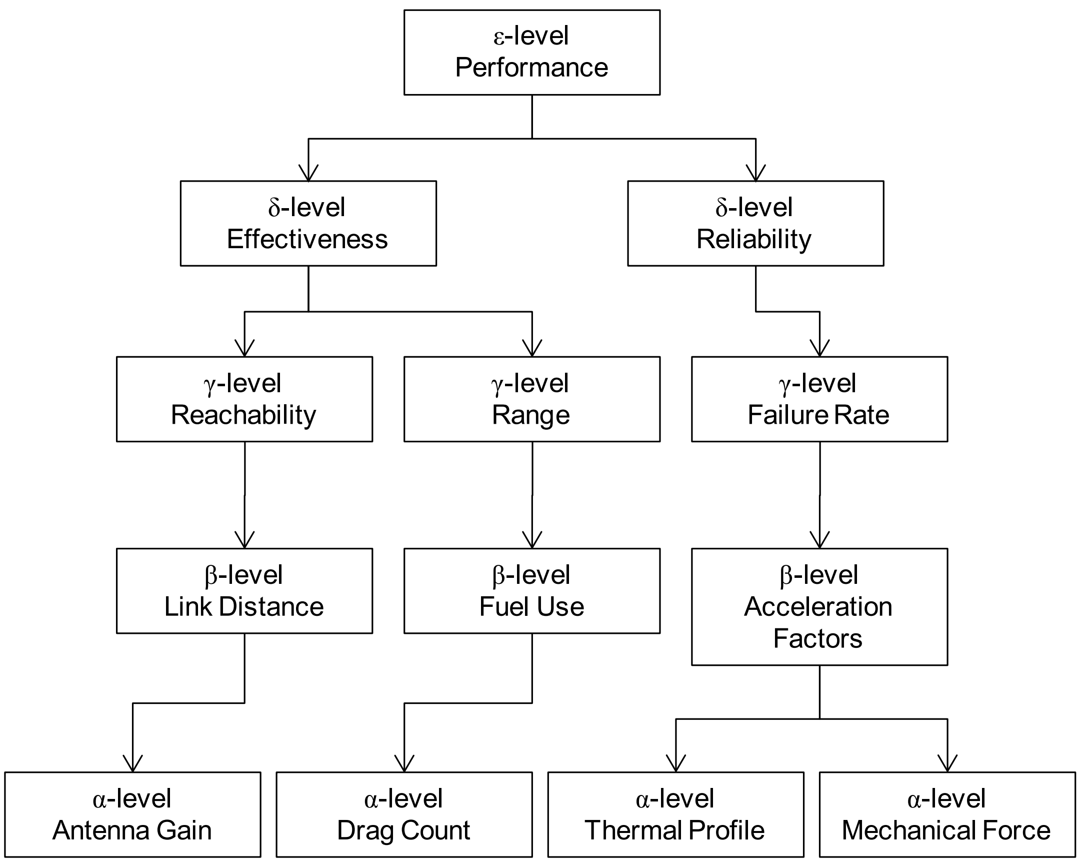


Figure 7.2. Decomposition for Multi-domain CDFC Assessment

Future work includes the design of a hierarchical assessment mechanism to combine the individual assessments from a multitude of technical domains. One approach is to use a hierarchical fuzzy inference system where α-level FIS outputs provide the β-level FIS inputs, etc. In this way, an architecture assessor could be built to operate at different levels of the extensible framework.

7.2. TYPE 2 FUZZY ASSESSORS

Those familiar with fuzzy sets will recognize that the fuzzy inference system employed in this research used type-1 fuzzy sets. The principle limitation of type-1 fuzzy sets is that although they are called “fuzzy” they do not address uncertainty (Mendel 2003). Type-1 fuzzy sets effectively embody the critical idea of partial membership, but for each element in the universe of discourse, the membership, μ, is crisp.

There are two kinds of high-level uncertainty of particular importance: random and linguistic (Mendel 2003). Random uncertainties pertain to the ability to estimate the value of the input variable along the universe of discourse. Random uncertainty in the FIS is analogous to the combined effects of measurement uncertainty and noise on the observation variable. Linguistic uncertainty pertains to the meaning of words, and specifically, the definition of the membership functions. This uncertainty is comprised of fuzziness (imprecise boundaries of the fuzzy set), non-specificity (the size of the fuzzy set), and strife (conflicts between sets) (Klir and Wierman 1998). Mendel (2003) suggests that all these sources of uncertainty impact fuzzy rule based systems in at least three ways.

1. The words used to identify the sets in the antecedent/consequent relationships mean different things to different individuals.
2. Different subject matter experts can assign different consequent sets to the same fuzzy rule.
3. The training or calibration data for the FIS is noisy.

An alternative to the type-1 set is available. Zadeh (1975) originally proposed a type-2 fuzzy set more than thirty years ago. Type-2 fuzzy sets account principally for the uncertainty in the meaning of words. Mendel (2001) demonstrated how uncertainty intervals in word meaning can be determined by polling a sample population. Interpreted from a systems perspective, this represents subject matter expert or stakeholder disagreement in set membership which directly impacts the shaping of the FIS membership functions. For example, if three stakeholders were asked to define a triangular fuzzy membership function rating a particular system measure, it is unlikely that they would generate the same triple f(x; a,b,c) describing that shape. In the simplest scenario, assume that the triangular shapes cover the same interval (c-a), and are simply shifted along the universe of discourse. This is shown in Figure 7.3 below.

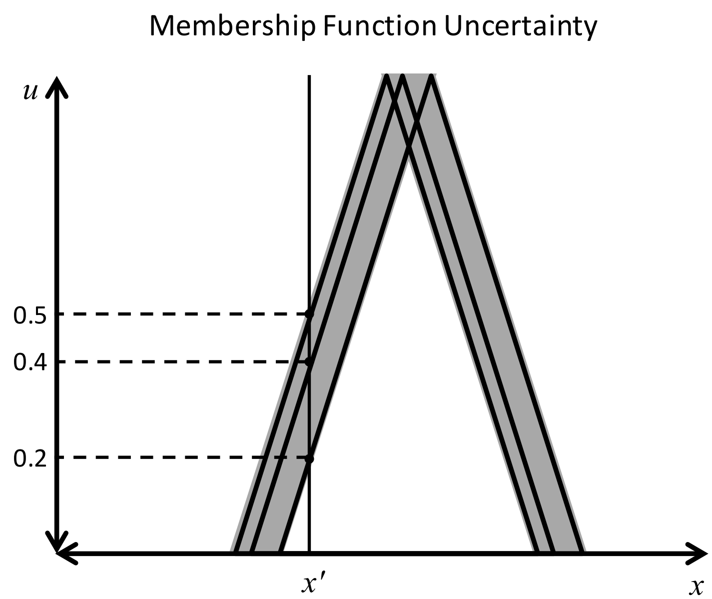


Figure 7.3. Type-2 Membership Function and Footprint of Uncertainty

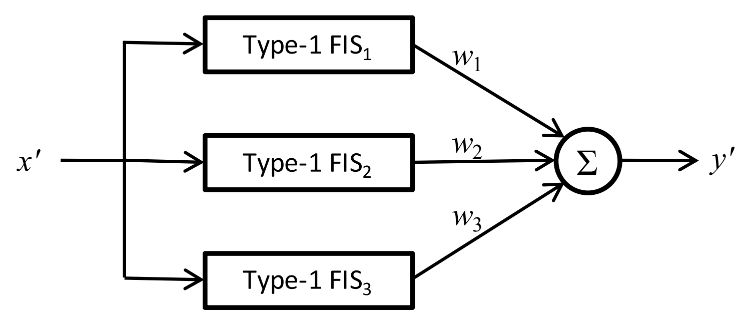
The upper and lower bounds of these three membership functions form the perimeter of a type-2 fuzzy membership function whose footprint of uncertainty (FOU) is shown as the shaded grey region. For each input value x', there are several primary membership grades, u. Each primary membership grade has a secondary membership grade, or weight [0,1]. The set of secondary grades for a single x' is called the secondary membership function. The weight distribution of the secondary grades can take many forms and may be interpreted as a probability distribution for the associated primary membership grades. The chosen probability distribution effectively characterizes the nature of the uncertainty associated with membership of the primary set.

Uncertainty in the meaning of words might also be handled with type-1 hedges. Concentration, dilation, and artificial hedges serve to expand and contract the fuzzy membership function based on degrees of certainty in assigning membership to an input quantity. The act of assigning membership as a composite of individual memberships of various hedges begins to resemble a type-2 treatment of fuzzy membership. Mendel (2001) suggests that the very notion of traditional hedges may need to be re-examined in the context of a type-2 set.

7.2.1. Uncertainty in the Observation Variable. Architecture definitions are regularly described as being ambiguous. Ambiguity exists because design details are incompletely specified. This is different than numerical uncertainty which might be described by an interval or a mean and standard deviation. The estimation of technical performance measures via computational analysis of canonical design primitives is deterministic. This means there is zero noise on the observed quantity. However there is an unknown error between the performance prediction of the design primitive and the final physical artifact. Error is the difference between a predicted and known value. Since the artifact whose performance is being estimated does not exist, the known value does not exist. Error is therefore indeterminate. The CDFC approach attempts to address this issue by not assessing the technical performance measure directly. Instead, the comparison between isolated and installed performance is assessed. It is believed that this accomplishes two objectives: exposing integration sensitivity directly and normalizing the installed response to the natural response of the probe structure.

7.2.2. Uncertainty in the Meaning of Words. There are two ways to view the need for type-2 assessment. The first perspective is that any uncertainty associated with the shaping of type-1 membership functions represents a disagreement among stakeholders. If this is the case, the system architect is not ready to assess alternatives because the desired system attributes have not yet been refined. Instead, the disagreement indicates that the architect needs to spend more time defining the customer needs. The second perspective is that type-2 sets add value to the approach in that multiple stakeholder interpretations can be encoded in the assessment. In this way, the assessor objectively considers the differences in stakeholder perspective. It is expected that the proper perspective will be dependent upon the circumstances within the program employing the approach.

7.2.3. A Pseudo-type-2 Assessor. One approach for achieving the intent of a type-2 fuzzy system is to combine the results of several type-1 fuzzy inference systems. In this approach, each type-1 FIS captures the fuzzy membership assignments of a single stakeholder. The output of each FIS is a crisp number and is multiplied by a weight, w_{sn}. The weight is analogous to the secondary membership grade and the set of weights $\{w_{s1}, w_{s2}, \ldots w_{sn}\}$ represents the secondary membership function. The secondary grades can be used to weight the individual assessments based on how influential or knowledgeable the stakeholder is. The summation of weighted individual FIS assessments, normalized by the sum of the weights, represents the output of the pseudo-type-2 assessor.



$$w_n = \mu_{\tilde{A}}(x', u_n)$$

Figure 7.4. A Pseudo-type-2 Fuzzy System Approach.

While it is acknowledged that this approach does not adhere to the formal mechanics of the type-2 fuzzy system, it is suggested that it accomplishes a similar objective. This proposed approach simplifies the type-2 computation while accounting for the differences among stakeholders.

According to Mendel (2003), there is no current theory that guarantees a type-2 fuzzy set will outperform a type-1 based system. Changing from type-1 to type-2 fuzzy sets for the CDFC FIS does not change the philosophy of the overall approach. Each element of the methodology can be developed, refined, or substituted to customize it to the specific needs of the program.

7.3. EMERGENCE

One of the critical features of the CDFC approach is the ability to examine coupled system performance. For this reason, it is believed that the CDFC methodology may offer a significant contribution to the effort of characterizing and understanding sources of Type 3 emergence. Many opportunities for research remain on the subject of emergence. Through an appreciation for the sources of emergence and their expected impacts, a systems architect can embrace the phenomenon rather than fear it.

APPENDIX A.

MATLAB CODE

```
% Jason Dauby
% Smart Engineering Systems Laboratory
% Missouri University of Science and Technology
%
% This script automates the assessment of antenna pattern data in support
% of CDFC architecture assessment.  The input file structure must be that
% of the .ffe as defined by FEKO computational electromagnetics software
% (www.feko.info)
clear;
clc;
clf;
clear all;
%*************************************************************************
%These values are the max and min values for data plotted
data_max = 4.5;
data_min = -4.5;
color_step = 3;
% Defines information about the .ffe file
Num_freq = 1;
Num_theta = 181;
Num_phi = 361;
theta_min = 0;
theta_max = 180;
phi_min = 0;
phi_max = 360;
% System Field of View (FOV)
el_win1 = 71;
el_win2 = 136;
az_win1 = 1;
az_win2 = 361;
%End user defined variables
%*************************************************************************
%Import Architecture Data*************************************************
[filename,pathname] = uigetfile('*.ffe', 'Select Architecture File');
fid = fopen([pathname,filename]);
%The asterisk denotes fields in the .ffe file that are not imported. In
%this case only theta directivity is imported.
%Data are imported into a cell array where columns are elevation cuts
%*************************************************************************
i=1;
while 1
    data_temp = textscan(fid, '%f%f%*f%*f%*f%*f%f%*f%*f', Num_theta,
'delimiter', '(), \t\n','multipledelimsasone', 1, 'collectoutput', 1);
    data_temp = data_temp{:};
    if isempty(data_temp)
        break
    else
        data_in{i} = data_temp;
        i = i+1;
    end
end
fclose(fid);
%Import Baseline data*****************************************************
[filename,pathname] = uigetfile('*.ffe', 'Select Baseline File');
fid = fopen([pathname,filename]);
%The asterisk denotes fields in the .ffe file that are not imported. In
%this case only theta directivity is imported.
%Data are imported into a cell array where columns are elevation cuts
%*************************************************************************
i=1;
while 1
    data_baseline_temp = textscan(fid, '%f%f%*f%*f%*f%*f%f%*f%*f', Num_theta,
'delimiter', '(), \t\n','multipledelimsasone', 1, 'collectoutput', 1);
```

```
    data_baseline_temp = data_baseline_temp{:};
    if isempty(data_baseline_temp)
        break
    else
        data_baseline{i} = data_baseline_temp;
        i = i +1;
    end
end
fclose(fid) ;
m = 1;
n = 1;
for k = 1:Num_phi*Num_freq
    %Assemble raw comparative data
    dataraw{k} = cat(1,data_in{1,k}(:,3)-data_baseline{1,k}(:,3));
    %Clip min/max data to the defined min/max range for plotting
    data{k} = cat(1,data_in{1,k}(:,3)-data_baseline{1,k}(:,3));
    while n <= Num_theta
        if data{1,k}(n,1) > data_max
            data{1,k}(n,1) = data_max;
        elseif data{1,k}(n,1) < data_min
            data{1,k}(n,1) = data_min;
        end
        n = n+1;
    end
    n = 1;
end
feature_data = cell2mat(dataraw);
%***************************************************************************
% Plotting the Full Comparative Data Set
%***************************************************************************
plot_data{m} = data{1,1+(m-1)*Num_phi};
for n = 2:Num_phi
    plot_data{1,m} = cat(2,plot_data{1,m},data{1,(m-1)*Num_phi+n});
end
%set color scale of graphs
z_limits = [-4.5 -1.5 1.5 4.5];
cmap = [1 0 0;1 1 0;0 1 0];
if data_in{1,1}(1,2) > 0
    data_zeros = zeros(data_in{1,1}(Num_theta,1),data_in{1,1}(1,2)-1);
    plot_data{1,m} = cat(2,data_zeros,plot_data{1,m});
else
    plot_data{1,m} = cat(2,plot_data{1,m});
end
graph_title = input('please enter the title of the graph: ', 's');
figure(1);
contourf(plot_data{1,1}, z_limits)
xlim([phi_min phi_max]);
ylim([theta_min theta_max]);
axis ij
axis([phi_min phi_max theta_min theta_max])
title(graph_title);
xlabel('Angle Phi (Degrees)');
ylabel('Angle Theta (Degrees)');
caxis([data_min data_max]);
colormap(cmap);
colorbar('CLim', [data_min data_max], 'YTick', [-4.5 -1.5 1.5
4.5],'YTickLabel', {'-4.5','-1.5', '1.5', '4.5'});
grid on;
set(gca,'ytick',[0 30 60 90 120 150 180]);
set(gca,'xtick',[0 30 60 90 120 150 180 210 240 270 300 330 360]);
fig{1} = gcf;
%***************************************************************************
% Plot Angular Window of Interest
```

```
%*************************************************************************
figure(2);
contourf(plot_data{1,1}, z_limits)
xlim([phi_min phi_max]);
ylim([el_win1 el_win2]);
axis ij
axis([phi_min phi_max el_win1-1 el_win2-1])
title(cat(2, graph_title, ' FOV'));
xlabel('Angle Phi (Degrees)');
ylabel('Angle Theta (Degrees)');
caxis([data_min data_max]);
colormap(cmap);
colorbar('CLim', [data_min data_max], 'YTick', [-4.5 -1.5 1.5
4.5],'YTickLabel', {'-4.5','-1.5', '1.5', '4.5'});
grid on;
set(gca,'xtick',[0 30 60 90 120 150 180 210 240 270 300 330 360]);
%*************************************************************************
% Interpret Features in Contour Plot
%*************************************************************************
az = 1;
el = 1;
L = 0;
ML = 0;
M = 0;
MH = 0;
H = 0;
numpoints = 0;
%*************************************************************************
% Identifies the percentage of angular points that have full membership in
% the set "Low"
%*************************************************************************
for el = el_win1:el_win2
    for az = az_win1:az_win2
        if feature_data(el,az)<-4
            L = L + 1;
        end
        numpoints = numpoints + 1;
    end
end
% Thresholds
L = L/numpoints
%*************************************************************************
% Average gain and sensitivity features
%*************************************************************************
FOV = feature_data(el_win1:el_win2,az_win1:az_win2);
%MAE is an indicator of integration sensitivity
MAE = mean(mean(abs(FOV)))
%Geometric mean is used since gain is a multiplicative factor
MeandB = mean(mean(FOV))
%*************************************************************************
% Assess Features
%*************************************************************************
Assessor = readfis('OneWindowAssessorR4');
Assessment = evalfis([MeandB MAE L], Assessor)
```

APPENDIX B.

FUZZY INFERENCE SYSTEM DESIGN

```
[System]
Name='OneWindowAssessorR4'
Type='mamdani'
Version=2.0
NumInputs=3
NumOutputs=1
NumRules=12
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='sum'
DefuzzMethod='centroid'

[Input1]
Name='Mean(dB)'
Range=[-7.5 3]
NumMFs=3
MF1='Low':'trapmf',[-8 -8 -4 0]
MF2='Med':'trapmf',[-3 -1 0.8 1.5]
MF3='High':'trimf',[0 3 4]

[Input2]
Name='MAE'
Range=[0 7.5]
NumMFs=3
MF1='Low':'trapmf',[0 0 1 2]
MF2='Med':'trimf',[1 2 3]
MF3='High':'trapmf',[2 3 8 8]

[Input3]
Name='Thresholds'
Range=[0 1]
NumMFs=3
MF1='Low':'trimf',[-3 0 0.2]
MF2='Med':'trimf',[0.1 0.2 0.3]
MF3='High':'trapmf',[0.2 0.4 1.5 1.5]

[Output1]
Name='output1'
Range=[0 1]
NumMFs=5
MF1='L':'trimf',[-0.5 0 0.15]
MF2='ML':'trimf',[0 0.2 0.4]
MF3='M':'trapmf',[0.25 0.45 0.55 0.75]
MF4='MH':'trimf',[0.6 0.8 1]
MF5='H':'trimf',[0.85 1 1.5]

[Rules]
1 0 0, 1 (1) : 1
2 0 0, 3 (1) : 1
3 0 0, 5 (1) : 1
0 1 0, 5 (1) : 1
0 2 0, 3 (1) : 1
0 3 0, 1 (1) : 1
0 0 1, 5 (1) : 1
0 0 2, 3 (1) : 1
0 0 3, 1 (1) : 1
1 1 0, 2 (1) : 1
2 1 0, 4 (1) : 1
3 1 0, 5 (1) : 1
```

BIBLIOGRAPHY

Adamy, D. 2001. *EW 101 A First Course in Electronic Warfare*. Norwood: Artech House.

Bar-Yam, Y. 2004. A Mathematical Theory of Strong Emergence Using Multiscale Variety. *Complexity*. 9(6): 15-24.

Blattert, S. 2008. NSWC Crane Modeling and Simulation: Fluid Dynamics and Thermal Analysis for Airborne Electronic Attack. Internal whitepaper.

Boman, R. M. 2008. Model-Driven Architecture for Platform and Platform Store Design. In *Proceedings of the Sixth Annual Conference on Systems Engineering Research* (Los Angeles, CA).

Calvano, C. N., and P. John. 2004. Systems Engineering in an Age of Complexity. *Systems Engineering Journal*. 7(1):25-34.

Charles River Analytics. 2004. "About Bayesian Belief Networks". Cambridge: Charles River Analytics.

Chen, P. and J. Han. 2001. Facilitating System-of-systems Evolution with Architecture Support. In *Proceedings of the International Workshop Principles Software Evolution*. (Vienna, Austria).

Chen, P. and J. Clothier. 2003. Advancing Systems Engineering for Systems-of-systems Challenges. *Systems Engineering*. 6(3):170-183.

Cloutier, R. 2008. Model Driven Architecture for Systems Engineering. In *Proceedings of the Sixth Annual Conference on Systems Engineering Research* (Los Angeles, CA).

Cureton, K. L. 1991. "Metaheuristics," USC Graduate Report, Dec. 9.

Dagli, C. H., and N. Kilicay-Ergin. 2009. "System of Systems Architecting," In *System of Systems Engineering Innovations for the 21st Century* ed. Mo Jamshidi, 77-100. Hoboken: Wiley.

Dagli, C. H., A. Singh, J. P. Dauby, R. Wang. 2009. Smart System Architecting: Computational Intelligence Applied to Trade Space Exploration and System Design. *Systems Research Forum* 3(2): 101-119.

Dauby, J. P., and C. H. Dagli. 2009a. Using Extensible Modeling in Systems Engineering and Architectural Search. In *Proceedings of the Third Annual IEEE International Systems Conference* (Vancouver, Canada).

Dauby, J. P. and C. H. Dagli. 2009b. “A Neural Network Approach to Modeling System Integration Sensitivity for Architectural Assessment,” In *Intelligent Engineering Systems Through Artificial Neural Networks Volume 19*, eds. Cihan H. Dagli et. al., 3-10. New York: ASME Press.

Dauby, J. P. and I. G. Guardiola. 2010. Computational Techniques for Probing Integration Sensitivity Functions and Response Surfaces. In *Proceedings of the Eighth Annual Conference on Systems Engineering Research*. (Hoboken, NJ).

Dauby, J. P. and C. H. Dagli. 2010. Complex Systems Architecting Using Design Primitives, Comparative Analysis and Fuzzy Analytical Feedback. In *Proceedings of the Fourth Annual IEEE International Systems Conference* (San Diego, CA).

Davidson, D. B. 2005. *Computational Electromagnetics for RF and Microwave Engineering*. Cambridge: Cambridge University Press.

DeLaurentis, D. and R. K. Callaway. 2004. A System-of-systems Perspective for Public Policy Decisions. *Review of Policy Research* 21(6): 829-837.

Doguc, O. and O. Kardes. 2009. Sensitivity Analysis Method for System Operational Effectiveness in Complex Systems. In *Proceedings of the Seventh Annual Conference on Systems Engineering Research* (Loughborough, United Kingdom).

Dyer, J. S., P. C. Fishburn, R. E. Steuer, J. Wallenius, and S. Zionts. 1992. Multiple Criteria Decision Making, Multiattribute Utility Theory: The Next Ten Years. *Management Science*. 38(5):645-654.

Eberhart, R. C. and Y. Shi. 2001. *Swarm Intelligence*. San Diego. Academic Press.

Eberhart, R. C. and Y. Shi. 2007. *Computational Intelligence: Concepts to Implementation*. Burlington: Elsevier.

EM Software & Systems. Comprehensive Electromagnetic Solutions – FEKO. EM Software & Systems. http://www.feko.info.

EM Software & Systems. 2009. “FEKO User’s Manual Suite 5.5”. Stellenbosch: EMSS.

Friedenthal, S., A. Moore, and R. Steiner. 2009. *A Practical Guide to SysML: The Systems Modeling Language*. Burlington: Elsevier.

Gagliardi, M., W. G. Wood, J. Klein, and J. Morley. 2009. “A Uniform Approach for System of Systems Architecture Evaluation,” *The Journal of Defense Software Engineering.* Mar/Apr: 12-15.

Gardiner, J. 2009. Linking Architecture Models and Synthetic Environments – An Integrated Project Support Environment. In *Proceedings of the Seventh Annual Conference on Systems Engineering Research*. (Loughborough, United Kingdom).

Ham, F. M. and I. Kostanic. 2001. *Principles of Neurocomputing for Science & Engineering*. New York: The McGraw-Hill Companies.

Haskins, C., ed. 2007. Systems Engineering Handbook. Version 3.1. Seattle: INCOSE.

Hitchins, D. 2003. *Advanced Systems Thinking, Engineering, and Management*. Norwood: Artech House.

Holland, J. 1975. *Adaptation in Natural and Artificial Systems.* Ann Arbor: University of Michigan Press.

Holland, J. 1998. *Emergence*. New York: Plenum.

INCOSE. A Consensus of the INCOSE Fellows. http://www.incose.org/practice/fellowsconsensus.aspx

Johnson, R. C, editor. 1993. *Antenna Engineering Handbook*. 3rd ed. McGraw-Hill.

Kalligeros, K., O. de Weck, R. de Neufville, and A. Luckins. 2006. Platform Identification Using Design Structure Matrices. In *Proceedings of the Sixteenth Annual International Symposium of the International Council on Systems Engineering* (Orlando, FL).

Keat, P. G. and P. K. Y. Young. 2003. *Managerial Economics – Economic Tools for Today's Decision Makers*. 4th ed. Upper Saddle River: Pearson Education.

Kennedy, J. and R. Eberhart. 1995. Particle Swarm Optimization. In *Proceedings of the IEEE International Conference on Neural Networks*. Vol. 4. pp.1942-1948.

Klir, G. J. and Wierman, M. J. 1998. *Uncertainty-Based Information*. Heidelberg: Physica-Verlag.

LeBaron, B. 2001. Evolution and Time Horizons in Agent-Based Stock Market. *Macroeconomic Dynamics*. 5:225-254.

Levis, A. H., 2005. Research Challenges in Design and Evaluation of System Architectures. Presented at the Third Annual Conference on Systems Engineering Research (Hoboken, NJ).

Lo, Y. T. and S. W. Lee, editors. 1988. *Antenna Handbook Theory, Applications, and Deign*. New York: Van Nostrand Reinhold.

Maier, M. W. and E. Rechtin. 2002. *The Art of Systems Architecting*. 2nd ed. Boca Raton: CRC Press.

MathWorks. Fuzzy Logic Toolbox – MATLAB. The MathWorks Inc. http://www.mathworks.com/products/fuzzylogic/.

MathWorks. 2009. "Fuzzy Logic Toolbox™ User's Guide". Natick: MathWorks.

Mendel, J. M. 2001. *Uncertain Rule-Based Fuzzy Logic Systems Introduction and New Directions*. Upper Saddle River: Prentice Hall.

Mendel, J. M. and R. I. B. John. 2002. Type-2 Fuzzy Sets Made Simple. *IEEE Transactions on Fuzzy Systems*. 10(2): 117-127.

Mendel, J. M. 2003. Type-2 Fuzzy Sets: Some Questions and Answers. *IEEE Connections*. 1:10-13.

Mitchell, M. 1998. *An Introduction to Genetic Algorithms*. Cambridge: First MIT Press.

Montgomery, D. C. 2005. *Design and Analysis of Experiments.* Hoboken: John Wiley & Sons.

National Research Council, The Committee on Modeling and Simulation for Defense Transformation. 2006. *Defense Modeling, Simulation, and Analysis – Meeting the Challenges*. Washington D.C.: Academies Press.

Object Management Group. 2003. *MDA Guide*. Version 1.0.1.

Object Management Group. 2008. *OMG Systems Modeling Language*. Version 1.1.

Peak, R. S., R. M. Burkhart, S. A. Friedenthal, M. W. Wilson, M. Bajaj, and I. Kim. 2007a. Simulation-Based Design Using SysML Part 1: A Parametrics Primer. In *Proceedings of the Seventeenth Annual International Symposium of the International Council on Systems Engineering* (San Diego, CA).

Peak, R. S., R. M. Burkhart, S. A. Friedenthal, M. W. Wilson, M. Bajaj, and I. Kim. 2007b. Simulation-Based Design Using SysML Part 2: Celebrating Diversity by Example. In *Proceedings of the Seventeenth Annual International Symposium of the International Council on Systems Engineering* (San Diego, CA).

Pearson, F. 1990. *Map Projections: Theory and Applications*. Boca Raton: CRC Press.

Purewal, K., L. Yang, and A. Grigg. 2009. Quantitative Assessment of Quality Attributes in Systems Architecture using Evidential Reasoning. In *Proceedings of the Seventh Annual Conference on Systems Engineering Research* (Loughborough, United Kingdom).

Rechtin, E. 1991. *Systems Architecting – Creating & Building Complex Systems.* Upper Saddle River: Prentice Hall.

Rechtin, E. 2000. *Systems Architecting of Organizations – Why Eagles Can't Swim.* Boca Raton: CRC Press.

Ross, A. M., D. E. Hastings, and J. M. Warmkessel. 2004. Multi-Attribute Tradespace Exploration as Front End for Effective Space System Design. *Journal of Spacecraft and Rockets*. 41(1): 20-28.

Ross, A. M. 2006. Managing Unarticulated Value: Changeability in Multi-Attribute Tradespace Exploration. PhD diss., Massachusetts Institute of Technology.

Siemens PLM. NX: PLM – Product Lifecycle Management. Siemens Product Lifecycle Management Software. http://www.plm.automation.siemens.com/en_us/products/nx/index.shtml.

Simpson, J. J. and C. H. Dagli. 2008a. Evolutionary Computation Applied to System Architecture Development. In *Proceedings of the Sixth Annual Conference on Systems Engineering Research* (Los Angeles, CA).

Simpson, J. J. and C. H. Dagli. 2008b. System of Systems Architecture Generation and Evaluation using Evolutionary Algorithms. In *Proceedings of the Second Annual IEEE International Systems Conference* (Montreal, Canada).

Singh, A. and C. H. Dagli. 2009. Multi-Objective Stochastic Heuristic Methodology for Tradespace Exploration of a Network Centric System of Systems. In *Proceedings of the Third Annual IEEE International Systems Conference*. (Vancouver, Canada).

Singh, A. and C. H. Dagli. 2010a. 'Computing with Words' to Support Multi-Criteria Decision Making During Conceptual Design. In *Proceedings of the Eighth Conference on Systems Engineering Research*. (Hoboken, NJ).

Singh, A. and C. H. Dagli. 2010b. Using Quality Attributes and Computational Intelligence to Generate and Evaluate System Architecture Alternatives. In *Proceedings of the Fourth Annual IEEE International Systems Conference* (San Diego, CA).

Stimson, G. W. 1998. *Introduction to Airborne Radar*. 2nd ed. Raleigh: SciTech.

Stutzman, W. L. and G. A. Thiele. 1998. *Antenna Theory and Design* 2nd ed. Hoboken: Wiley.

U.S. Department of Defense. 2001. *System Engineering Fundamentals*. Fort Belvoir: Defense Acquisition University Press.

U. S. Department of Defense. 2007. *DoD Modeling and Simulation (M&S) Management 5000.59*. Washington, DC: Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics.

U. S. Navy – Fact File. 2009. C-12 Huron logistics aircraft. United States Navy. http://www.navy.mil/navydata/fact_display.asp?cid=1100&tid=300&ct=1

Wang, R. and C. H. Dagli. 2008a. An Executable System Architecture Approach to Discrete Events System Modeling Using SysML in Conjunction with Colored Petri Nets. In *Proceedings of the Second Annual IEEE International Systems Conference* (Montreal, Canada).

Wang, R. and C. H. Dagli. 2008b. Executable System Architecture Using SysML in Conjunction with Colored Petri Nets. In *Proceedings of the Sixth Annual Conference on Systems Engineering Research* (Los Angeles, CA).

Wang, R. and C. H. Dagli. 2008c. "Optimum Architecture Development Using Evolutionary Programming," in *Intelligent Engineering Systems Through Artificial Neural Networks* Volume 18, ed. C. H. Dagli et. al., New York: ASME Press.

White, B. E. 2007. On Interpreting Scale (Or View) and Emergence in Complex Systems Engineering. In *Proceedings of the First Annual IEEE International Systems Conference* (Honolulu, HI).

Weilkiens, T. 2006. *Systems Engineering with SysML/UML. Modeing, Analysis, and Design*. Burlington: Kaufmann.

Xu, L. and J. Yang. 2001. "Introduction go Multi-Criteria Decision Making and the Evidential Reasoning Approach". Working Paper No. 0106. Manchester School of Management, Manchester, UK.

Zadeh, L. A. 1975. The Concept of a Linguistic Variable and its Application to Approximate Reasoning. *Information Science*. 8: 199-249.

Zadeh, L. 1996. *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A. Zadeh*. Ed. G. J. Klir and B. Yuan. Singapore: World Scientific Press.

Zadeh, L. A. 1999. From Computing with Numbers to Computing with Words – From Manipulation of Measurements to Manipulation of Perceptions. *IEEE Transactions on Circuits and Systems – I: Fundamental Theory and Applications*. 45(1): 105-119.

Ziemer, R. E. 1997. *Elements of Engineering Probability and Statistics*. Upper Saddle River: Prentice Hall.

VITA

Jason Paul Dauby was born in Indiana, USA on September 13, 1978. He received the Bachelor of Science degree in Electrical Engineering from the University of Evansville in May 2001 and the Master of Science degree in Engineering Management from the Rose-Hulman Institute of Technology in May 2005. He began his studies at the Missouri University of Science and Technology in January 2007 and earned the Ph.D. degree in Systems Engineering in May, 2011.

While pursuing his degree, he was also employed as an RF systems engineer at the Crane Division, Naval Surface Warfare Center in Indiana. His professional assignments included technical analysis and design for various airborne, ground, and maritime electronic warfare programs. This experience included the use of model-based analysis and design throughout all phases of the program life-cycle.

Jason Paul Dauby has been a member of the International Council on Systems Engineering (INCOSE) since 2010 and the Association of Old Crows (AOC) since 2002. His research on the Canonical Decomposition Fuzzy Comparative methodology earned the INCOSE Foundation/Stevens Institute Doctoral Award for Promising Research in Systems Engineering and Integration for 2010. Additional research interests include architecture assessment, generating system architectures using computational intelligence, and architecting for emergence.