A FUZZY FRAMEWORK FOR ROBUST ARCHITECTURE IDENTIFICATION IN CONCEPT SELECTION

A Thesis
Presented to
The Academic Faculty

by

Frank H. Patterson

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
December 2015

A FUZZY FRAMEWORK FOR ROBUST ARCHITECTURE IDENTIFICATION IN CONCEPT SELECTION

Approved by:

Professor Daniel Schrage, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Professor Dimitri Mavris
School of Aerospace Engineering
Georgia Institute of Technology

Professor Brian German
School of Aerospace Engineering
Georgia Institute of Technology

Professor George Vachtsevanos
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Sylvester Ashok
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: November 9, 2015

To my growing family,

Love does multiply.

ACKNOWLEDGEMENTS

First and foremost I would like to thank Dr. Daniel Schrage for giving me the opportunity to return to school for my graduate degrees. He gave me a chance when no one else would, and provided an opportunity that changed not just my career, but my entire life. Thank you for your continued support and encouragement. And many thanks to Bill McCandless for vouching for me yet again.

I would also like to thank the rest of my committee for their time, help and guidance throughout this process.

I owe so much to my amazing family for all they have provided me, and for their love, support and faith throughout my entire career (academic and otherwise). My deepest gratitude to my amazing wife, Martha, and my mother, without whose selfless sacrifice I would have never been able to even begin a PhD. And thank you to my son, Everett, who lights up my every day, no matter how long.

In my seemingly short time in graduate school at Tech, I have been privileged to work with some of the best colleagues I've ever had. Thanks to the graduate students and research engineers (Rich, Nate, Sylvester, Mike M., Mike R., Kyle, and many, many more), who provided indispensable advice and insight on making it through this. I would also like to thank all of my labmates in my time with the IPLE, who have made my time there so enjoyable.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Acronyms

AHP Analytic Hierarchy Process.

ANN Artificial Neural Networks.

BAA Broad Agency Announcement.

DARPA Defense Advanced Research Project Agency.

DFES Discrete Fuzzy Expert System.

DSM Design Structure Matrix.

FM Figure of Merit.

FRBS Fuzzy Rule Based System.

GA Genetic Algorithm.

IPPD Integrated Product and Process Development.

IRMA Interactive Reconfigurable Matrix of Alternatives.

JPDM Joint Probabilistic Decision Making.

MADM Multi-Attribute Decision Making.

MCDM Multi-Criteria Decision Making.

MCS Monte Carlo Simulation.

MISO Multi-Input Single Output.

MODM Multi-Objective Decision Making.

NFS Neuro-Fuzzy System.

NSGA Nondominated Sorting Genetic Algorithm.

OEC Overall Evaluation Criterion.

PDF Probability Density Function.

QFD Quality Function Deployment.

RDS Robust Design Simulation.

RSE Response Surface Equation.

SISO Single Input Single Output.

TOPSIS Technique for Order of Preference by Similarity to Ideal Solution.

TRL Technology Readiness Level.

VTOL Vertical Takeoff and Landing.

SUMMARY

The primary objective of this research is to address gaps in concept selection made critical by trending development of a new generation of systems, with new capabilities, capable of accomplishing missions that conventional vehicles can't be empirically redesigned to perform. Modern conceptual design methods often utilize physics-based, multi-disciplinary, and probability based techniques to extend conceptual design and more completely understand the design space. These approaches can forecast robust and successful designs, but are often computationally expensive and time consuming to develop and execute for a single given concept. This investment in conceptual design amplifies the need to identify a manageable number of capable and robust concept alternatives early in the design process.

A review of current concept selection methods illustrates significant gaps in concept down-selection, with a lack of approaches for exploring dissimilar architectures while accounting for uncertainty. Some of the most often utilized methods to identify concepts, such as filtered morphological analysis, can produce millions or billions of feasible alternatives. However, common approaches to compare and make selections from among disparate alternatives require designers to identify a relatively small subset of alternatives for analysis. These simple approaches use methods regularly related to decision matrices (Pugh, TOPSIS, AHP, etc.), and rarely account for the uncertainty inherent in analysis during these early design phases. More complex approaches can account for uncertainty, but are based largely on conceptual design techniques requiring time to setup and run more complicated modeling and analysis tools. This makes these approaches reasonable for only a discrete few core architectures, and

certainly not the full design space.

This research seeks to develop an approach to consider the uncertainty inherent in predicting system capability during concept selection, while exploring large design spaces for promising alternatives. A proposed flexible framework leverages fuzzy systems in conjunction with morphological analysis to assess large design spaces of potential architecture alternatives while accounting for the uncertainty and ambiguity in the assessment of these early concepts. After an analysis of fuzzy set theory as a means for capturing uncertainty in concept selection, experiments explore several approaches to using fuzzy systems for alternative evaluation. During data elicitation, identification and analysis of design parameters driven by architectural choices provide links to drive fuzzy systems from a morphological matrix. Systems of varying complexity are then shown to work as models for both expert judgement and physics-based data. These models can provide first-order estimates of system performance with relation to various criteria of interest while inherently capturing the uncertainty in the evaluation.

In the final phase of research, these fuzzy system models are combined into a single framework for multi-disciplinary analysis of alternatives from the morphological matrix. By wrapping the framework in a multi-criteria optimizer, the entire design space is explored for a Pareto set of the most promising architecture alternatives. Experiments demonstrate that a multi-criteria genetic algorithm, specifically the Non-Sorting Genetic Algorithm II can be used to identify a diverse population of non-dominated alternatives in the design space. With a large set of promising architectures identified, several means for analysis and decision making are explored to identify architecture families or individual alternatives to move forward with into conceptual design.

The proposed approach allows for exploration of the entire design space of compatible alternatives within the morphological matrix while accounting for the uncertainty

inherent during concept selection. This research contributes a method for decision makers to analytically identify conceptual architectures with potential that might not otherwise be considered in traditional concept identify methods. Moreover, it presents a flexible and capable methodology for understanding complex system problems in the earliest stages of design, even in architecturally disparate design spaces.

CHAPTER I

INTRODUCTION

Engineering design of complex aerospace systems is often divided into three phases: conceptual design, preliminary design, and detailed design. In conceptual design, designers will consider a wide spectrum of system configuration concepts and ultimately develop the single most promising design concept to pass on to preliminary design [88]. Traditional conceptual design was often a data-driven point design process, where proven concepts were resized and refitted to a new set of requirements. More recently, early design phases have undergone a transformation to meet an emerging focus in government and industry on system life-cycle affordability and overall system capability. Here affordability is defined as not simply an initiative to reduce the cost of a given system, but as the initiative to maximize the system's effectiveness with respect to the cost of obtaining that effectiveness. One reason for the shift towards capability based design is the increasing desire to design a generation of vehicle concepts with no historical basis to meet radically new requirements and perform new missions.

To meet these challenges, advanced design methods have been incorporated, including elements such as physics-based modeling, multidisciplinary analysis and optimization methods, and the use of probability theory. Many of these advanced design methods were predicated on the idea of developing more knowledge of the design space earlier in the design process in order to understand the impact of early design decisions on capability [74]. This ability to know more about the design space, and maintain design freedom by carrying more design alternatives further into the design process has become critical to the success of modern acquisition programs. Figure

1 illustrates the desired effects of advanced design methods on design knowledge, freedom, and cost commitment.

In order to utilize the physics based models and multidisciplinary analysis tools that the advanced conceptual design methods are based on, one must first define the concept space. That is, some number of promising concept configurations must be generated and evaluated, and the best one (or few ones) must be selected through some "concept selection" method. In general, the more fidelity conceptual design tools possess, the more limited they are in the scope of what configuration elements can be modeled, and the more costly the models become computationally. This means that many concept configuration trades are made before advanced design methods can be utilized and thus much of the design freedom lies in, and is lost through, the concept selection process.

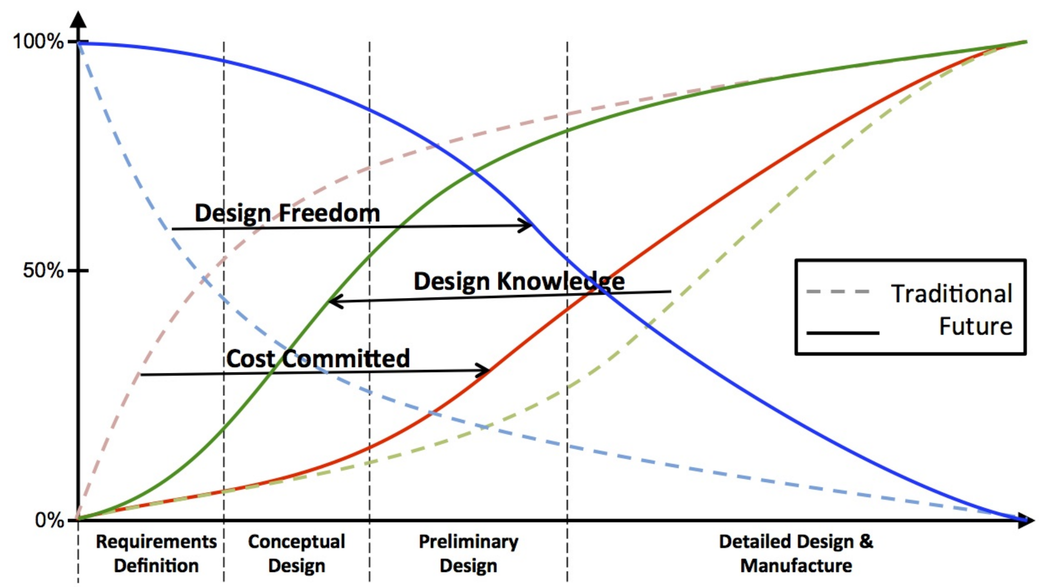


Figure 1: Phases of Design [74]

The core of many of the aforementioned advanced design methods were born through the pioneering of the concurrent systems engineering approach dubbed Integrated Product and Process Development (IPPD) [97]. IPPD is a *methodology* *that incorporates a systematic approach to the early integration and concurrent application of all the disciplines that play a part throughout a system's life cycle* [80]. A generic, iterative IPPD framework developed by Georgia Institute of Technology is illustrated in Figure 2. Conceived as a framework to enhance affordability and capability in advanced systems, IPPD led to the development of Robust Design Simulation (RDS), illustrated in Figure 3, which was the vehicle for developing many of the physics-based, multi-disciplinary advanced design methods. Robust design is defined by Dieter as"the systematic approach to finding optimum values of design factors which lead to economical designs with low variability" [34]. In robust design the designer attempts to limit variability by more fully understanding the uncertainty in the design product and design process, and then making appropriate decisions. RDS was conceived as a framework to develop a thorough and integrated understanding of a design's cost, schedule, and performance to identify affordable and robust designs early in concept development.

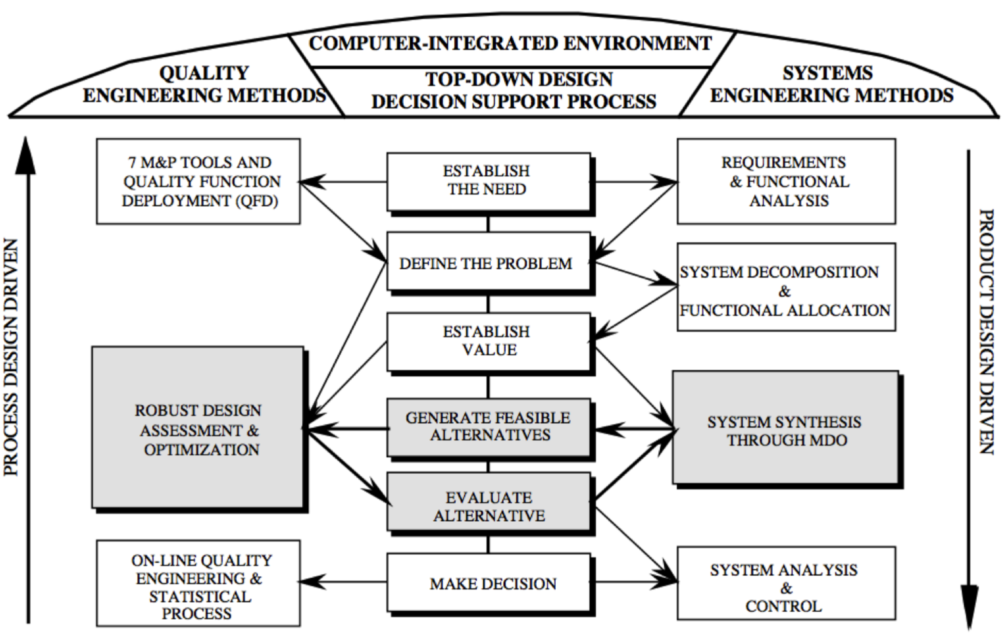


Figure 2: Georgia Tech's Generic IPPD Process [97]

It is important to understand that uncertainty is inherent in the design of these

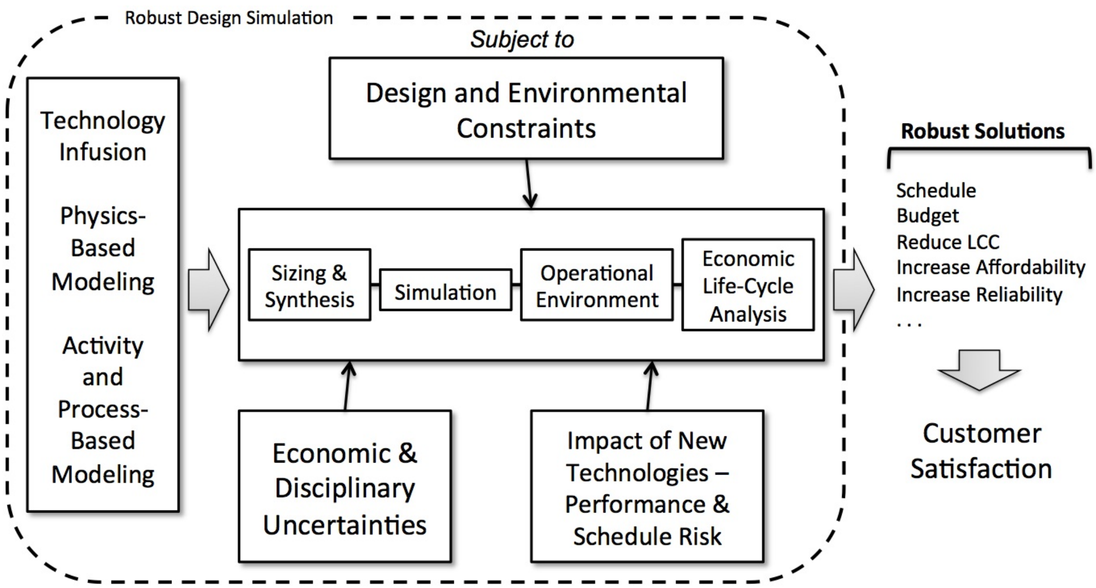


Figure 3: Robust Design Simulation [97]

complex systems, and thus means are necessary to account for, analyze, and control uncertainties [74]. Uncertainty exists at every stage of design from uncertainty inherent in subjective analysis early in conceptual design, to uncertainty in manufacturing precision, or even the uncertainty due to the variability of the operating environment of the final product. At the outset of design only the problem is known, and even that may be vague. As the design process progresses, and knowledge about the product system increases, more certainty is gained over the final qualities and performance (or costs) of the system. Uncertainty is especially important to consider in the earliest stages of design where qualitative and/or linguistic treatments are often used with respect to requirements, leading to ambiguity. The results of interactions in complex systems are often difficult to predict, and simpler qualitative evaluations can be ambiguous, making disagreement between designers common in these earliest stages of design.

An important element in the IPPD method is establishing value early in the design process and tracking that value as the design progresses. An Overall Evaluation Criterion (OEC) is a function often established early in design to correlate system effectiveness and cost, essentially providing a comprehensive metric for system affordability. The OEC can be monitored throughout the design process to perform system trade studies and gauge product and process improvement.

Managing system robustness through cycling of RDS throughout the design process, with continuous product and process improvement is illustrated with respect to an OEC in Figure 4. Here the uncertainty at the outset of conceptual design is considered the "Fuzzy Front End", and a goal of conceptual design is to define the OEC's probability distribution early in the design process. By quantitatively defining system uncertainty earlier in conceptual design, the designer can better understand the design space and begin to make more informed decisions at that time.

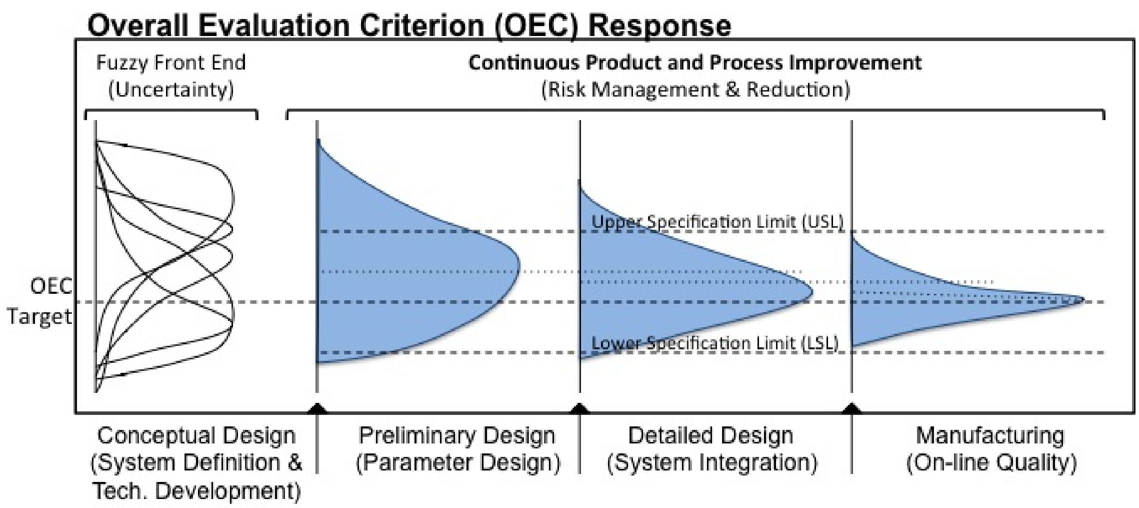


Figure 4: Continuous Product Improvement [97]

While over reliance on an OEC is not ideal in multi-criteria design situations, as it can lead to sub-optimal designs in non-convex problems [72], the same concept can be applied to any design metrics throughout the design process. By gathering more knowledge about the problem, decisions can be made leading to a higher likelihood of a desired and optimal outcome.

In the concept selection phase of design, it is indeed important to understand uncertainty and control variability in design outcomes. When a concept configuration is selected for continued development, the inherent performance, capabilities, and affordability available to that system are fixed to large degree, bound to the physics and realities that come with that concept. As illustrated roughly in Figure 1, this early design decision limits design freedom more than any future decisions. If mistakes are made early in the conceptual design process, their repercussions will be felt strongly in the realization of the final system.

Concept selection is an often over-looked and under-valued process in the conceptual design of complex systems. Even when requirements call for revolutionary missions and significant advancement throughout the system life-cycle, concept configuration is often selected as a result of designer bias, organizational inertia, existing design tools, or other reasons with limited engineering substantiation. These practices stifle innovation and limit system and industry potential. This serves as motivation of this research, which is meant to understand the gaps in available concept selection methods and fill them appropriately.

1.1 Concept Selection

There are an number of tools and methods available to designers during concept selection, however most of them are incomplete solutions, are of limited scale, or provide limited information. Morphological analysis, proposed by Fritz Zwicky in the 1960's, is a widely used method for generating and exploring alternative concept configurations in complex systems. Morphological analysis is "a method for identifying and investigating the total set of possible relationships or 'configurations' contained in a given problem complex" [89]. Here the problem is broken down into a set of physical and/or functional system parameters (or dimensions), which are then populated with a series of options. A functional decomposition often allows a broader view of the problem, generating system attributes such as "Means for Locomotion", and possibly lending more towards innovation and novel concepts. A physical decomposition can lend to a more specific, detailed approach to the problem when the general approach is already known, generating system attributes such as "Rotor Configuration". A short notional morphological matrix is illustrated in Table 1.

Table 1: Example Morphological Matrix for Notional Helicopter

System Attribute	Options		
Rotor Config	Single Main	Tandem	Coaxial
Anti-Torque	Tail Rotor	NOTAR	None
Landing Gear	Fixed	Retractable	Floats

Morphological methods often leave millions, billions, or more possible solutions for the designer to sort through. The Interactive Reconfigurable Matrix of Alternatives (IRMA), an extension of morphological analysis, was developed by Engler, et al in 2007 [40]. An IRMA can be utilized to reduce the size of the design space by filtering for inherent compatibility between alternatives, Technology Readiness Level (TRL), maturity, or risk, and incorporates desirable features such as allowing fast collaborative design, and providing flexible, reconfigurable matrices. A similar compatibility approach was also suggested earlier by Kirby and Mavris with respect to technology selection [56]. The IRMA process is often applied iteratively with some more detailed analysis and decision making done on the remaining options at each iteration. Even with these methods, designers can still be left with tens or hundreds of thousands of possible concept configurations to decide between.

Some concept selection methods have been proposed as directly coupled to morphological analysis. Weber and Condoor proposed a very simplistic means of concept configuration synthesis combining coupling functions with nested morphological matrices [132]. Strawbridge and McAdams created a mathematical approach to morphological analysis using a functional model of system [110], proposing that a computational approach could cover more of the design space. However their approach was deterministic and did not account for multiple objectives. Lafleur and Sharma use morphological matrices to guide the selection of a baseline architecture for a robotic

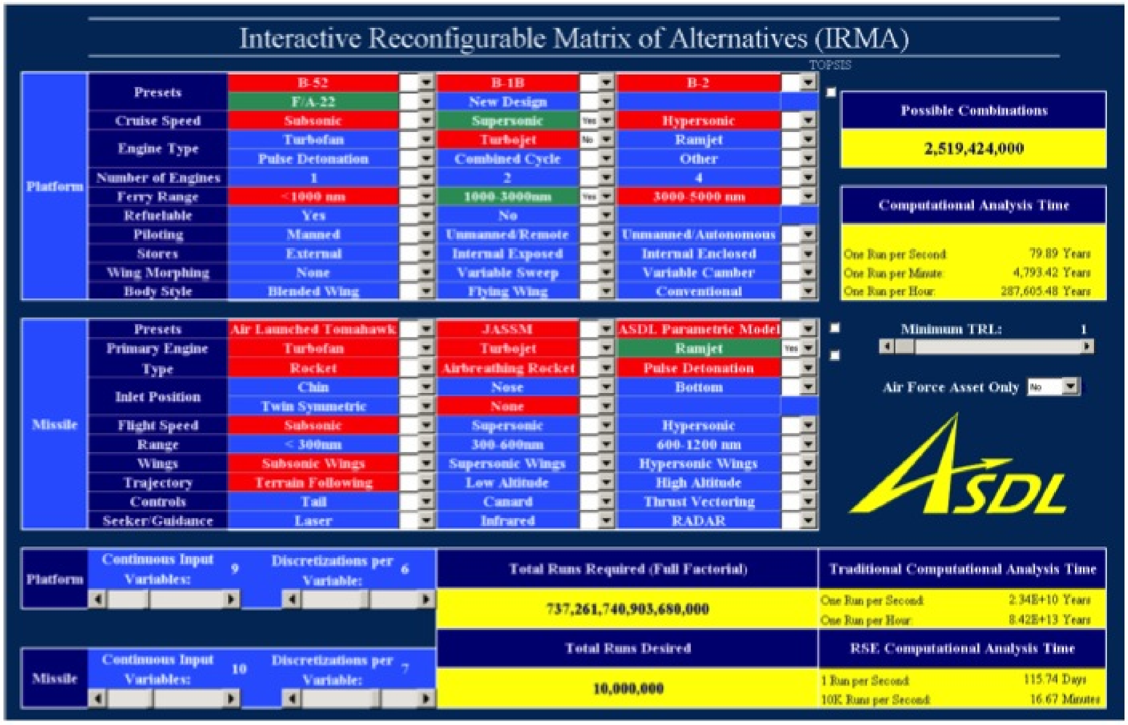


Figure 5: Interactive Reconfigurable Matrix of Alternatives (IRMA) [40]

space exploration payload [60]. The proposed process is multi-objective, but highly qualitative, and simplistic.

Regardless of the means used to generate alternatives for concept configuration, some method or series of methods must be used to eventually converge on a single selected concept to move forward design process. Currently available concept selection methods range from simple pairwise comparisons to those utilizing complex mathematical principles. Likewise, these methods assess potential concept configurations through methods ranging from the simplest qualitative comparisons to utilization of physics-based models and complex computational algorithms. Here, the most prevalent current methods for concept selection decision making are discussed along with their shortcomings.

1.1.1 Decision Matrices, Multi-Criteria Decision Making, and AHP

Perhaps the simplest and most common methods used in concept selection are variations on the decision matrix. Here, a set of concept alternatives are each evaluated with respect to a set of criteria desired in the concept [73]. Weights assigned to each criteria are often applied to convey the preferences of the designer. Various mathematical means of determining the best alternative are then used, such as summing the product of each alternative's scores and their criteria's respective weights.

The simplest variation on the decision matrix is the Pugh method [86], where concept alternatives are compared with respect to various evaluation criteria as better than, worse then, or the same as some baseline alternative. The best alternatives are simply those determined to be better than the baseline(s) with respect to the most criteria. A slightly more complicated version involves weighting the pluses and minuses based on their respective criteria. This quick and easy method involves only qualitative assessment and does not provide the rigor that defining an advanced concept configuration requires [78].

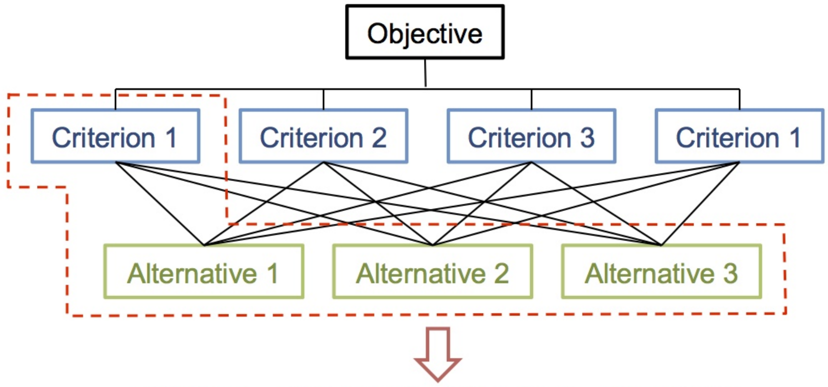
Decision matrices are often combined with more complex mathematics in attempt to improve the optimization process. For instance, Muller provided an approach using decision matrices and linear physical programming for concept selection [77]. As with most common decision matrix methods, this is deterministic and suited for a pre-determined set of alternatives.

Decision matrices are just one strategy (albeit a common one) of Multi-Attribute Decision Making (MADM), a general category of methods containing popular means for selecting a concept alternative. MADM methods are a subset of Multi-Criteria Decision Making (MCDM), a category of methodological processes and tools for modeling complex engineering problems where decisions must be based on multiple, possibly conflicting, objectives and constraints (the criteria). MADM methods, discussed further in Section 2.3, are defined by a discrete decision space with countable pre-defined alternatives. MADM methods generally utilize relatively simple mathematical models to distill alternative rankings from scorings with respect to a series of conflicting criteria [121]. Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), and Analytic Hierarchy Process (AHP) are two popular MADM methods, often used to select a concept configuration [15].

TOPSIS allows for the use of qualitative and simple quantitative scores with respect to weighted criteria, and ranks alternatives through their relative Euclidian distance from both ideal positive and ideal negative solutions [136]. Schrage proposes the use of both Pugh matrices and MADM techniques (specifically TOPSIS) to select a concept [97] in proposing a means for technology selection through IPPD. Engler, et al. propose iteratively using TOPSIS in coordination with his IRMA for a final concept selection [40].

AHP, a hierarchical method, decomposes large decision making problems into a series of smaller subproblems and allows the designer to make simpler pairwise comparisons among criteria and alternatives to develop weightings and then a hierarchy of given alternatives using several possible mathematical approaches [70, 91]. In a review of concept selection methods, Okudan identifies almost a dozen separate AHS approaches to concept selection in the last few decades[82].

These methods, like most MADM methods, require the designer to select a small, manageable set (10-30) of concept configuration alternatives or families to compare as each alternative must be scored with respect to each criteria. This inherently requires a contraction of the design space by the designer to generate a manageable set, reducing what could be tens of thousands or more alternatives to just a few alternatives or families of alternatives. In doing this, the designer risks losing viable design concepts and information about individual attributes of the design. Additionally, hand selecting concept alternatives from a morphological matrix introduces bias



Criterion 1	Alternative 1	Alternative 2	Alternative 3
Alternative 1	1/1	2/1	1/4
Alternative 2	1/2	1/1	1/3
Alternative 3	4/1	3/1	1/1

Figure 6: Analytic Hierarchy Process

into the design process, and often results in the designer selecting concepts that are familiar. In most cases, concept selection using these decision matrix and MADM approaches are approached deterministically, with each criteria weight and alternative score being a single fixed value. In most cases, this approach ignores the imprecision and uncertainty inherent in projecting the capabilities of systems that have yet to exist. However, some attempts at accounting for uncertainty in concept selection have recently been proposed, as discussed in the next section.

1.1.2 Concept Selection Methods Accounting for Uncertainty

The IPPD methodology and advances in modern conceptual design have been accompanied by the development of methods that use various means to try and model the uncertainty inherent in concept selection. Designers have applied probabilistic techniques in concept selection to account for some uncertainty in the selection process. Using Quality Function Deployment (QFD), Biltgen, et. al. derived multiple engineering characteristics from customer needs in an interactive environment, and evaluated a series of alternatives quickly with TOPSIS [15]. The importance weightings for customer needs were then modeled with simple probability distributions (uniform and triangular) and Monte Carlo Simulation (MCS) was utilized to generate histograms for each alternative's relative importance. This methodology can certainly capture the subjectivity of what a customer wants, but it does not attempt to account for uncertainty in alternative evaluation and it still has many of the short comings of MADM methods as it uses a small set of alternatives evaluated individually and deterministically. Some MADM methods have been modified to consider uncertainty in both customer's needs and the alternative evaluations, essentially assigning probability distributions to perviously deterministic values and using Monte Carlo to perform the MADM analysis, creating probabilistic results [11, 59]. In these methods, the assignment of score distributions (and thus robustness) in the assessments is commonly very subjective. There are some available methods that attempt to account for uncertainty in an alternative in more complex ways and identify robust alternatives.

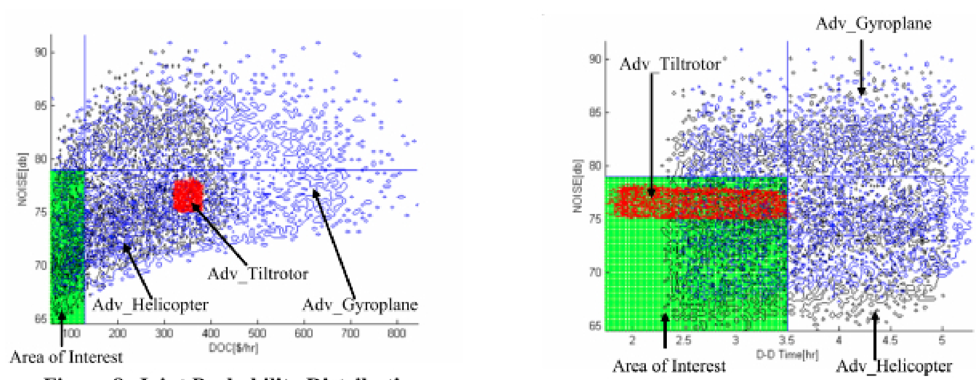


Figure 8: Joint Probability Distribution (Noise vs. DOC)

Figure 9: Joint Probability Distribution (Noise vs. D-D Time)

Figure 7: Concept Selection using JPDM [64]

More complex probabilistic concept selection methods have introduced limited physics-based modeling into the selection process. Li uses a combination of MCS and Response Surface Equation (RSE) to explore concept spaces and perform tradeoffs in mission requirements, design parameters, and technologies simultaneously with multiple concept configurations [64]. Joint Probabilistic Decision Making (JPDM) is then employed to select the concept configuration that is determined to have the highest probability of meeting various product and process design goals simultaneously, as illustrated in Figure 7. This methodology inherently injects an element of robustness into concept selection by utilizing probabilistic methods, and bypasses much of the subjectivity of expert evaluation by using physics-based modeling. In a similar approach, Villeneuve also combined RSEs and MCS in selecting a launch vehicle architecture, then used s-Pareto frontiers to aide in selecting a concept [124]. Mattson and Messac introduced the idea of s-Pareto frontiers in 2002, as a Pareto frontier both within a single concept and between a set of multiple concepts (shown in Figure 8) [72, 73]. Mattson also reviews a number of means to vary outputs to model uncertainty, including MCS, orthogonal arrays, importance sampling, and sensitivity-based methods such as Taylor's series expansion.

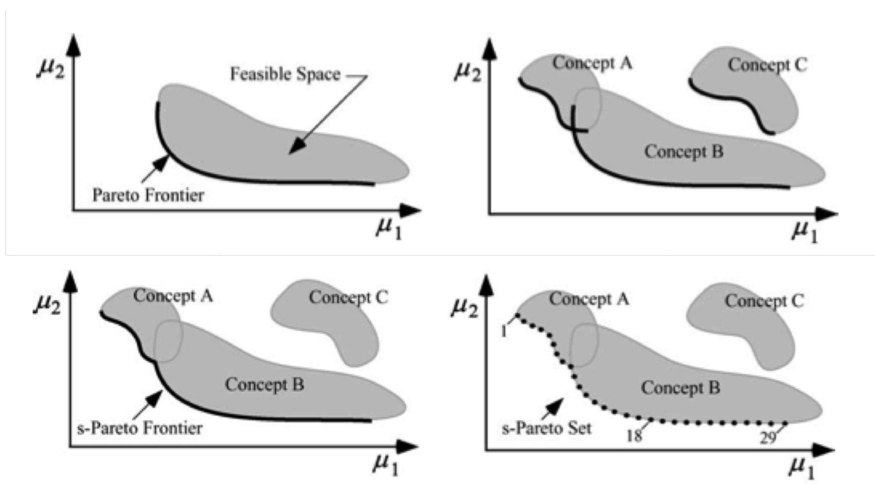


Figure 8: Development of an s-Pareto Frontier [71]

Smaling introduces what he refers to as "fuzzy Pareto frontiers" in the selection of an architecture for improving internal combustion engine through technology [105]. Design parameters for each of the alternative architectures are varied, and the individual concept Pareto frontiers are expanded by allowing solutions within some margin of the frontiers to be considered under the assumption that Pareto optimal solutions are those that are least robust against change in design parameters (any variance in these design would result in a loss of Pareto optimality).

These methods all require a small set of pre-selected alternatives to perform the analysis on, as modeling every alternative would be impractical. For each alternative, a valid model must exist or be built in order to produce the necessary variation in alternative design outcomes. Developing and integrating these models (often physics-based models or calibrated cost models) takes significant time and manpower. Typically only a very small handful of concepts or closely related concept families, even less than could be evaluated with most MADM methods, can be explored using this method if the designer wishes to keep this time manageable. In some cases it may not even be feasible to accurately model some elements of a radically new concept configuration at this early phase. Hypothetically, even if the models were to already exist for all feasible alternative systems for a given problem, generating the necessary data for thousands of possible alternatives would certainly be time prohibitive.

A number of approaches have also proposed the use of fuzzy methods for modeling uncertainty in concept selection [82], or proposed adjustments to MADM methods to include fuzzy methods as a means of accounting for uncertainty in decision making. A popular approach is to extended AHP to include fuzzy numbers, with Van Laarhoven, Ayağ, and Lee each proposing various means of a fuzzy AHP for concept selection [5, 58, 62]. Triantaphyllou and Lin propose a means of adapting several MADM techniques, including TOPSIS, to account for uncertainty using fuzzy numbers [122]. Wang and Elhag improve upon that approach to fuzzy TOPSIS by developing a more nuanced method that produces less skewed results Triantaphyllou [128]. Wang proposed a method using fuzzy outranking to identify non-dominated concept alternatives, including a sensitivity analysis to vary the levels of uncertainty in the selection and observe the results on concept rank order [126]. A number of other authors have proposed various fuzzy MADM methods for a number of applications with similarities to concept selection as well [54].

In a related method, Zhai and his colleagues asked design team members to propose notional quantitative values for each alternative's criteria performance, and combined the results using rough sets [141]. They then applied grey relation analysis to compare uncertainty in the results and recommend an alternative. This method does account for some uncertainty without the use of complicated models, but still requires the designer to select any feasible alternatives they wish to compare.

These modified MADM approaches have many of the same downfalls as traditional, crisp, MADM methods. They require individual scores for each criteria for all alternatives, and compute individual scores and ranking for every alternative, both becoming infeasible with an exponentially large number of alternatives. Thus these methods are better suited to a small number of pre-selected feasible alternatives.

1.1.3 Multi-Objective Decision Making/Optimization

Concept selection methods have also been developed to exploit Multi-Objective Decision Making (MODM) and optimization methods such as genetic algorithms [85, 19]. MODM methods, often based on multi-objective optimization techniques, seek to synthesize the best solution from a given design space, most often considering a number of continuous design variables. These methods are better suited than MADM for searching a large concept space, and can be developed to provide an element of robustness. The vast majority of methods for concept selection and design through MODM methods involve Genetic Algorithm (GA)s. GAs are ideal for their ability to relatively quickly find global optimums using a "black box" fitness approach, where no knowledge of the mathematical workings (in this case the models) of the systems analyzed is needed by the decision maker.

Mosher attempts to use GAs to select the concept for a spacecraft design, including selecting several categorical configuration options similar to those in a morphological matrix, but only considers 432 combinations of commercial off-the-shelf spacecraft components [76]. Buonanno uses various methods involving GAs to select a concept. Parallel, variable fidelity GAs are used to identify Pareto frontiers in multi-criteria decision space in one proposed method [19]. Another uses Interactive Genetic Algorithms (IGAs) to alleviate some of the issues of limited model fidelity in early design and allow a human expert to subjectively and qualitatively judge evolving designs occasionally for feasibility beyond what the modeling and simulation environment is capable of recognizing [18]. A hybrid approach to GAs is proposed by Chung, inserting gradient based optimization into the GA. Computationally expensive fitness values, such as those generated by computational fluid dynamics are modeled for intermediate solutions using Kriging. This requires rather complex models for all considered concepts.

These methodologies have many of the same limitations as the previous discussed robust methods, with the time to develop and run physics-based models limiting the number of concept families that can be considered. They do introduce the a limited number of discrete variables to affect the vehicle configuration, but is in essence exploring only a single, closely related family or limited number (2-5) of model-able families of concepts. When selecting between concept families that are drastically different, using a single model to capture all of their behaviors becomes prohibitively complex and time consuming. Thus many of these methods more accurately fall after an initial concept down-selection, as some means has to be used to identify a specific concept family or families to explore.

Mattson and Messac's approaches using s-Pareto frontiers can also be considered in it's relation to multi-criteria optimization [72]. Here the s-Pareto frontier is used to determine the optimal region of design space to explore, but their methods still require pre-selected alternatives and related concept models.

1.1.4 Other Related Methods

A variety of other selection methodologies have been proposed for conceptual design [73, 82]. In this section, some additional approaches to concept selection are outlined that do not necessarily fall into one of the popular approaches defined previously.

There are several concept selection and design methods focused on using utility theory, addressing customer preferences as well as accounting for uncertainty. Thurston proposes formal methods to compare alternatives using a multi-attribute utility function to determine overall utility based on those systems' varying performance [117, 118]. A similar method is developed by Fernández, et al., building utility functions for each attribute of a proposed system to be used to compare alternatives based on predicted levels for all attributes [41]. Uncertainty is captured by creating probability distributions for each alternative's levels of each attribute. These approaches provide a rigorous mathematical means to capture and analyze designer preferences across multiple criteria while accounting for some risk or uncertainty in the design process. However, the means the designer must define and manipulate a number of utility functions, and makes the approach time consuming and complicated even for a limited number of alternatives. Utility theory based approaches are best suited for when a small number of prospective alternatives have already been identified.

In another approach, Li provides a variation on his JPDM decision making technique, integrating Taguchi's loss function to provide a physical value to a concept's variability or deviation from target values [65]. This however, does not reduce the complexity of the required models or limitations on alternative's considered, instead adding a layer of computational complexity.

Tahid and Okudan propose a unique approach for accounting for uncertainty in concept design by utilizing fuzzy information axiom [114]. This utilizes axiomatic design principles, which use a hierarchical structure (illustrated in Figure 9) and essentially state that the best design has independent functional requirements and has the lowest "information content", giving it the highest probability of meeting a functional requirement. This is measured by using fuzzy triangular numbers to map compatibility of concept components to functional requirements and other components, combining information content for each to rank all possible alternatives. This simple approach does account for some uncertainty in the design and could be expanded to quantitative scoring, however it requires the computation of scores for all possible alternatives. With the simple example of 81 alternative combinations in the provided example this is quickly accomplished, but more complicated problems with more functional requirements and more concept options quickly become computationally infeasible as the number of combinations increases exponentially with the number of functions.

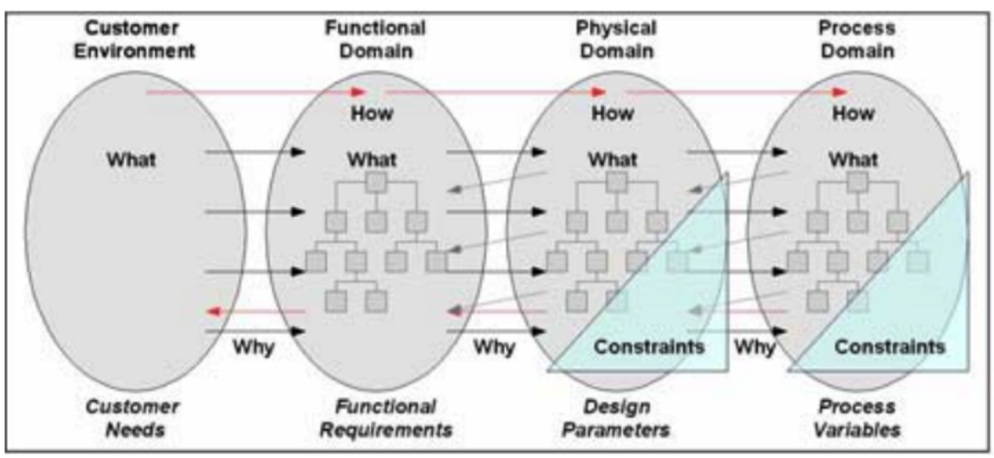


Figure 9: Domain, Mapping, and Spaces in Axiomatic Design [114]

Augustine, et al. propose a means for generating improved hybrid alternatives from a number of existing concepts using Fuzzy Inference Systems, but the method has limited scalability, requires significant designer interaction, and has limited applicability to complex multi-attribute systems [4]. Topology optimization is often useful for identifying and selecting vehicle geometries and layouts [135], but does not provide an ideal framework to assess performance and process objectives. Patel, et. al.

use physical programming and graph theory to select between subsystem alternatives and define a concept configuration, but the methodology lacks any means to account for robustness, and the tools to account for dissimilar configurations [83].

Providing an exhaustive list of every nuanced method proposed for concept selection in modern engineering design would be a huge task. However, the methods listed in this section provide a relatively comprehensive overview of the credible approaches that might be used to select a concept for the design of a complex and revolutionary new system in the face of multiple conflicting criteria. Despite a multitude of approaches explored in this section, there are still gaps in the concept selection that can be addressed. The following section identifies those gaps and proposes a means to address them.

1.2 Research Objectives

As a review of current concept selection methods demonstrates, there are are a number of excellent methods that have been proposed for concept selection and demonstrated with success. However, this author proposes that when the designer faces the task of identifying potential architecture configurations in the design an unconventional new system with profound new capabilities, there are significant gaps. Means to generate feasible new concept configurations such as an IRMA can leave in excess of hundreds of thousands of possible configuration alternatives (even after several iterations) that the designer must immediately reduce to a manageable set for methods such as MADM to be applied. This introduces significant bias in to the design process. MADM methods lack the ability to consider large numbers of alternatives without time intensive expert evaluation and/or modeling of each potential architecture. Moreover, these methods often cannot accurately account for considerable uncertainty early in the design process.

Methods that do consider uncertainty and use quantitative assessment require enough complexity and time investment to reduce the number of potential alternatives considered even further. These methods are often used in the last iterations of the conceptual design process, where the number of alternatives have been reduced to just a manageable few. Because of the simplicity of means available to downselect alternatives, this leaves little means of comparing the results of robust concept selection methods to the assessed alternatives that weren't carried forward. These gaps in the concept selection process are illustrated in Figure 10.

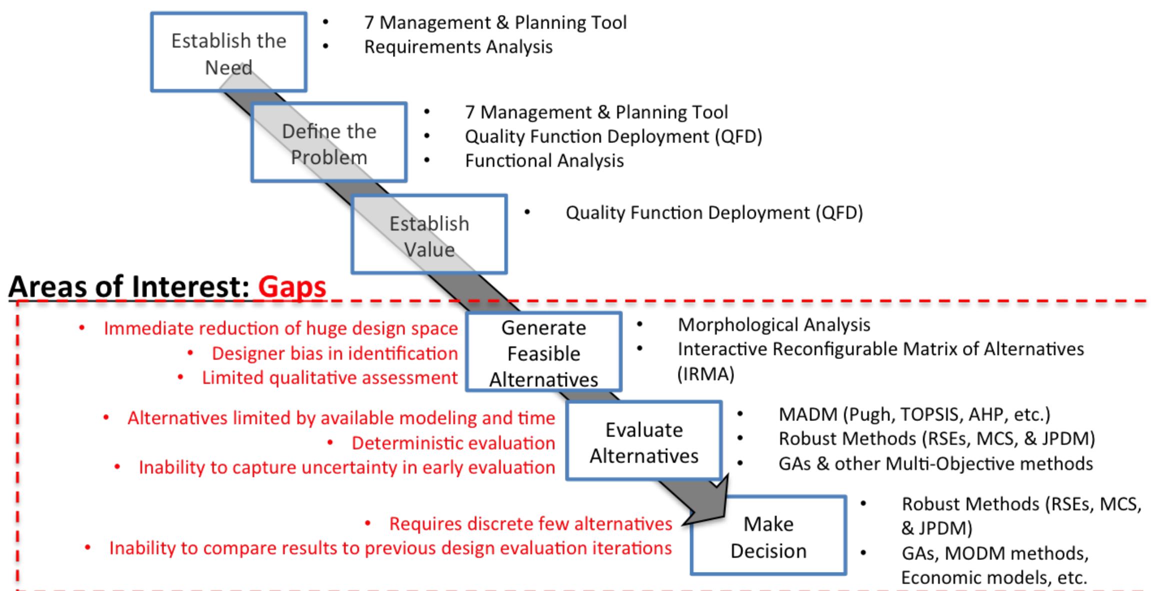


Figure 10: Gaps in Current Concept Identification Methods

The purpose of this research is to understand how the gaps present in concept selection methods for complex new systems might be filled. It is proposed here that many of the gaps can be filled by harnessing simple expert systems and models of first-order design tools to quickly evaluate architectural alternatives in much the same way many modern conceptual design methods use response surface methods to model higher fidelity design tools. Moreover, it is proposed that one means of building these tools involves the application of various methods and techniques related to fuzzy set theory. Fuzzy sets were first introduced by Zadeh in 1965 [138], and have since been studied and used in a wide range of domains from engineering and computing to business and health sciences. The idea of fuzzy sets was first used to represent the uncertainty inherent to human cognitive processes, and has since been expanded to help represent many forms of incomplete or uncertain information.

There were essentially only two choices to provide the logical and mathematical system to handle uncertainty here, probability and the aforementioned fuzzy set or "possibility" theory. A fuzzy set approach was selected for use due to it's ease for modeling human intuition and thought processes, which is particularly advantageous in a phase of design where expert judgement is often used to make most evaluations. There is, perhaps, some potential for a probabilistic approach to fill these gaps, but it is believed it would not be as simple or straightforward as an approach based on fuzzy sets and fuzzy systems. Probability is built on rigor and exacting techniques, while a fuzzy approach allows for flexibility in translation and application. Some methods like Bayesian Networks [107] might be used, but good probabilistic methods are driven by data, which is conspicuously absent at this stage of design. The cost of generating some reasonable database of expert analysis data for large frameworks is likely very high, but a probabilistic means for the larger framework remains a potential area of future research. With that in mind, the rest of this section describes the goals of this research, outlines how the application of fuzzy methods can be utilized to meet these goals, and asks the questions that this research is meant to answer.

In order to develop a measurably robust solution, it is important to capture uncertainty in concept evaluation at the earliest stages of design. As previously discussed, modern conceptual design has seen an influx of probabilistic and other related methods to measure and control uncertainty early in the design process. By gathering more knowledge about the design earlier in the design process, a better understanding of design decisions can be had and the design can be manipulated to be robust against future risks to the design process. Uncertainty can come in many forms, especially early in the design process. This includes, but is not limited to, the possibility of changes in requirements, variations in the impact of immature technology, and variations in manufacturing processes. There is also uncertainty in the projected capabilities and performance of immature concepts before models can be developed and the requirements applied explicitly to the conceptual design. Risk and uncertainty should be captured for each alternative as early as possible in the design process and propagated to the decision making phase of concept selection. This leads to the first objective of this research:

Research Objective 1. *Understand and capture uncertainty while assessing system concept performance at the earliest stages of design.*

In order to facilitate gathering knowledge of the design even earlier in the design process, quantitative evaluation of concept configurations should begin as early as possible. Unfortunately the most commonly available concept down-selection methods, MADM and related techniques, are often done qualitatively. One of the core motivations of this research is to provide a means to quantitatively define the "Fuzzy Front End" of the design process as soon as possible. Quantitative knowledge about the design space allows the designer to understand gaps between the design and requirements that may have to be bridged using technologies. It also provides a link to future design iterations involving physics-based and multi-disciplinary exploration of one or more conceptual designs. For these reasons, the research will primarily focus on quantitative means to assess uncertainty.

The most obvious gap identified in the process of selecting a new concept configuration is a lack of means to address the entire design space. Some more traditional missions and requirement sets obviate one or a select few concept alternatives. However, for the development of aerospace systems with radically new missions or capabilities, a complete investigation of the design space and possible concept configurations should be undertaken. Existing concept selection methods are either limited to a feasible number of pre-determined alternates (5-20), such as the MADM methods, or require some set of model tools capable of handling a small set of combinations in a family of similar designs, as when using a genetic algorithm. Figure 11 outlines many current concept selection methods with some of the limitations in their capabilities and the number of alternatives they can generally handle.

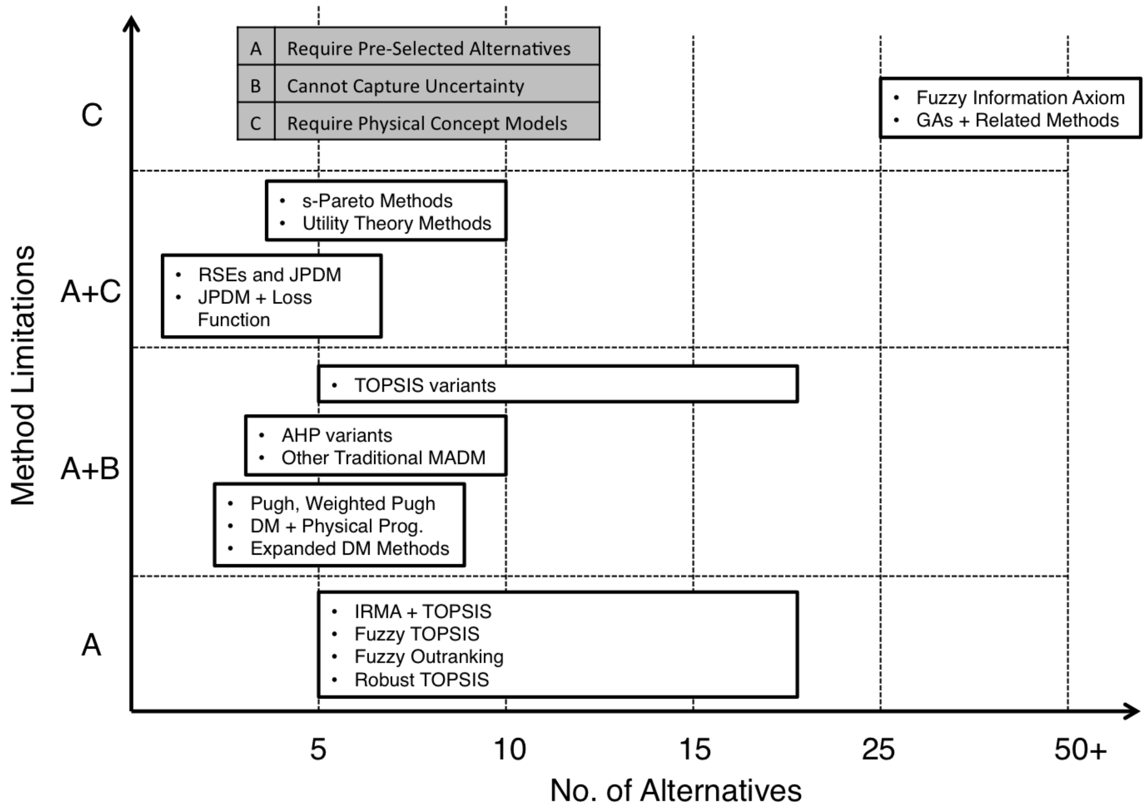


Figure 11: Charting the Capabilities of Existing Concept Selection Methods (adapted from [82])

In order to fill this gap, the primary focus of this research is to develop some framework capable of meeting the first research objective (capturing uncertainty), while exploring as much design space as the possible. For the purpose of this dissertation, that design space will be defined as any feasible, compatible combination defined within a morphological matrix built to generate alternatives for a given system. This means that alternatives or families of alternatives will not be preselected, nor will the assessed concept configurations be limited to fit available models or tools. The second objective of this research is as follows:

Research Objective 2. *Provide a means to quickly explore large design spaces including systems with dissimilar architectures.*

Morphological analysis is a powerful tool for generating potential concept alternatives, but existing methods only utilize a small subset of potentially thousands or millions of identified feasible alternatives. Once mature, the subject framework should allow a designer to explore the entire feasible space of concept configurations to determine the most promising configuration or configurations to proceed to a more detailed and objective analysis with. Evaluating every possible alternative generated by a large morphological matrix with respect to each criteria can be a time prohibitive exercise, even after the principles of an IRMA are applied. A method needs to be developed to allow for evaluations of every feasible, compatible alternative with respect to each criteria quickly and logically.

The use of fuzzy systems is proposed for this purpose of evaluating alternatives. Fuzzy systems provide a flexible basis to develop simple, logical expert systems and perform quick analysis of system attributes under uncertainty. Moreover, it is proposed in this research that some set of these simple expert systems and models of first-order design tools can be arranged into a sort of Design Structure Matrix (DSM) based on the dependance of their inputs. The matrix will be fed by design parameters based on some combination of selected options for each functional aspect of the system within the morphological matrix, as illustrated in Figure 12. It is believed that the resulting framework will provide a means to quickly capture expert opinion (and first order data) to assess potential architectures. Similarly to a DSM, the framework can then be harnessed by some means of exploring the design space, and seeking out desirable alternatives or designs. This research seeks to explore how feasible it is to generate, train, and drive these models/systems in the context of a larger framework, as well as how to utilize the framework in conceptual design.

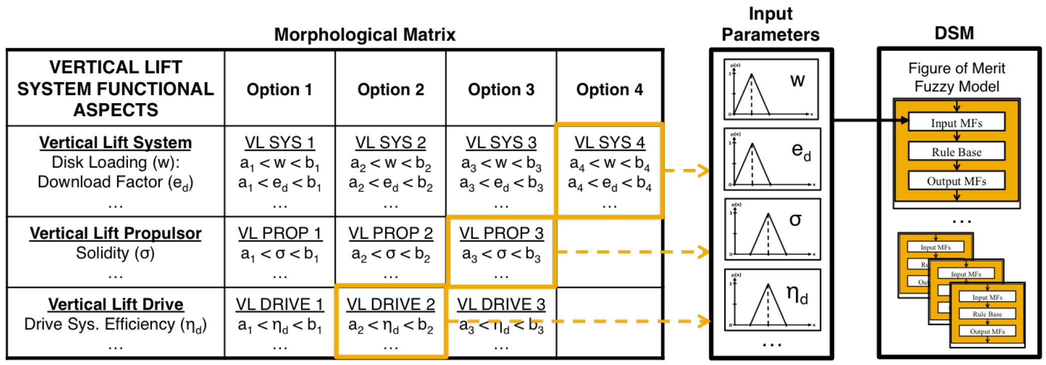


Figure 12: Generating Input Parameters for Expert Systems/Fuzzy Models)

The methodology developed in this research will be meant to identify robust concept architectures that project to perform well with respect to a given set of criteria. The framework will be able to quantitatively evaluate the system of performance of alternatives, including measuring uncertainty, throughout the design space without the need for developing costly physics-based models for every possible configuration. With these research objectives outlined in mind, the next section outlines the research questions that this research hopes to answer in order to complete the research and meet the given objectives.

1.3 Research Questions and Hypotheses

A survey of literature has exposed gaps in the available methods for concept selection, and several objectives have been outlined for this research, with the goal of developing a framework to fill these gaps. As discussed in Section 1.2, it is proposed that methods related to fuzzy set theory can provide a means to meet research objectives. Fuzzy sets, as discussed in Section 2.1, have been used consistently as a means to account for uncertainty in human judgement, or when complete information is unavailable. Additionally, fuzzy methods are inherently intuitive, designed as a means to interpret linguistic assessment. When very little is known about a problem's uncertainty and conservatism is the best approach, fuzzy method based approaches have even been shown to provide preferred results over probabilistic approaches [25].

In addition to being a fitting tool for assessing uncertainty in concept selection, fuzzy methods also make available a host of tools and techniques that have already been developed and verified. These tools include several means for modeling qualitative and quantitative evaluations of alternatives through expert assessment and modeling of existing data sets [54]. Fuzzy methods also include fuzzy systems (discussed further in Section 3.3), a unique and powerful tool that provides several systematic means of connecting fuzzy inputs and fuzzy outputs, including through expert opinion [7, 51]. Fuzzy MCDM methods are also a well developed area of research, as outlined in Section 2.3.3, with a large library of available methods for making decisions under uncertainty.

1.3.1 Capturing Uncertainty in Value

Before tools can be developed to model system performance in the face of uncertainty, it is prudent to understand the basis being used for these models. To that end, this research begins by asking the following question:

Research Question 1. *During concept selection, can fuzzy sets provide an interpretable measure of the uncertainty inherent in system performance?*

This question will be addressed during the benchmarking phase of this research. Fuzzy sets provide various options for accounting for the uncertain performance of an unrealized concept, the imprecision inherent in expert evaluation, disagreement between experts, and variation in potential historical data [49, 54, 133]. Various means of generating fuzzy numbers, similar to those shown in Figure 13, will be utilized for different methods of evaluation. A benchmark will seek to demonstrate the closest current state-of-the-art methods for selecting a concept architecture by employing probabilistic versions of MADM methods, namely TOPSIS and AHP [59]. To address Research Question 1, existing fuzzy versions of the same MADM methods (discussed further in Section 2.3.3) will applied to the same problem and compared to the probabilistic results. This application will explore various means of representing system performance with fuzzy sets, and attempt to interpret the results and their implications on concept selection.

It is hypothesized that fuzzy sets can indeed be used to model uncertainty in system criteria and play a role in guiding decision making when evaluating alternatives. While interpretation of fuzzy membership functions is perhaps less intuitive than probability functions such as a cumulative distribution function (CDF) or probability density function (PDF), they should be no less valid. Furthermore, it is hoped to show that results from both fuzzy and probabilistic MADM methods indicate similar paths forward in concept selection.

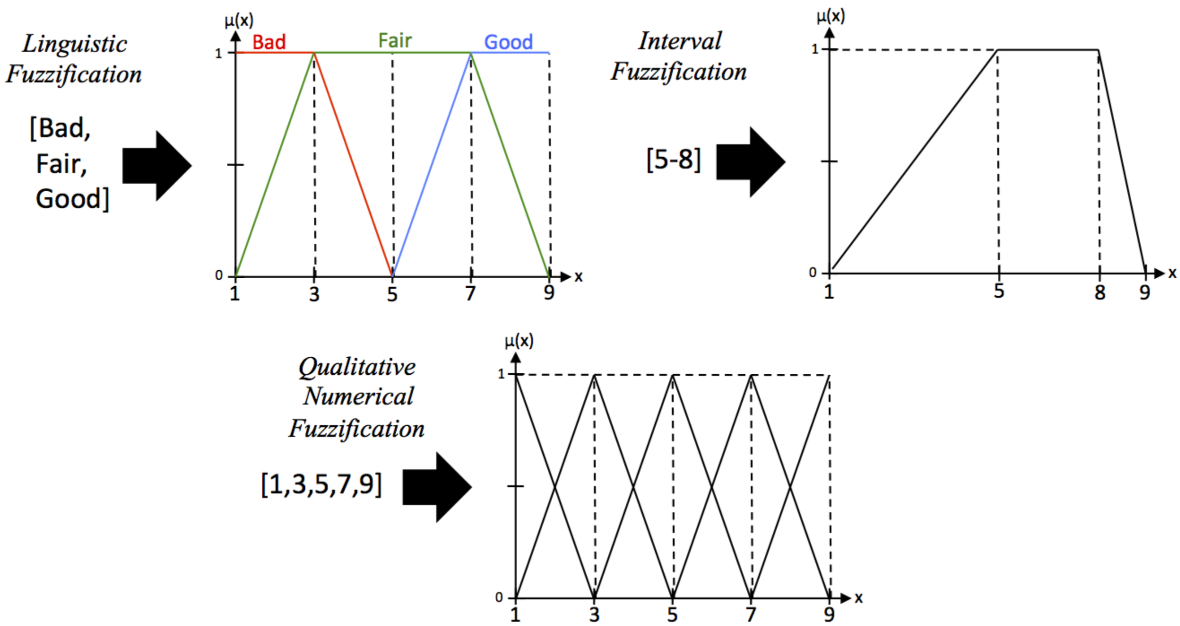


Figure 13: Examples of Fuzzification

1.3.2 Generating and Evaluating Alternatives

As previously proposed, in order to meet the Research Objective 2 and explore the entirety of the design space, a means of evaluating feasible architecture alternatives will be integrated into a framework. It is the supposition of this research that evaluated alternatives can be created through the integration of a morphological matrix and fuzzy inference system(s) as shown in Figure 12.

The intention of this research is to use fuzzy systems to build computational models of both expert opinion and simple physics-based design tools, allowing for quick and dynamic evaluation of large numbers of alternatives. Fuzzy systems, discussed in more detail in Section 3.3, have the ability to provide a logical means to evaluate alternatives based on their constituent parts. Various types of fuzzy systems have been used as expert systems or artificial intelligence [81], in the fields of medical diagnoses, investing, condition based maintenance, and more [103]. In order to implement fuzzy systems to evaluate system performance with respect to some criteria, the following research question must be asked:

Research Question 2. *How can fuzzy systems be used to evaluate architecture alternatives early in conceptual design?*

Because of the huge number of available fuzzy system architectures, and the numerous ways that fuzzy systems can be utilized, this research will attempt to demonstrate several relevant means of utilizing fuzzy systems as models for evaluating concepts, rather than attempt an exhaustive search. For that reason the answer to this question is expected to come in a number of parts, as some systems may be more viable than other for different applications. The systems explored will be driven by evaluating individual physical or functional attribute options, as previously stated,

It is hypothesized that if the desired criteria are only dependent on a few (2-4) primary system inputs, then can be evaluated with simple fuzzy inference systems. Systems driven by only a handful of inputs can be driven by a manageable set of expert defined if-then rules. Fuzzy inference systems, discussed in more detail in Section 3.3, can provide a simple (often linguistic) structure for mapping inputs to outputs, creating systems with potentially complex behaviors. In keeping with Research Objective 1, these systems can and will be driven with fuzzy inputs, rather than single crisp values. Moreover, the systems will be designed to produce fuzzy outputs in the form of membership functions that serve as a indicator of uncertainty in the performance indicated.

Research to answer this question will concentrate on developing fuzzy systems with a reasoning mechanism based on conjunction operators (most commonly a Mamdani or Larsen model) built around linguistic databases and linguistic if-then rule bases. An example database for an inference type system might notionally include the fuzzy sets illustrated in Figure 14, and include rules such as:

IF *FoM (V.L. System)* is *high* **AND** *Download (Fuselage)* is *low*

THEN *FoM (Aircraft)* will be *high*.

IF *FoM (V.L. System)* is *high* **AND** *Download (Fuselage)* is *NOT low*

THEN *FoM (Aircraft)* will be *med*.

IF *FoM (V.L. System)* is *medium* **AND** *Download (Fuselage)* is *NOT low*

THEN *FoM (Aircraft)* will be *low*.

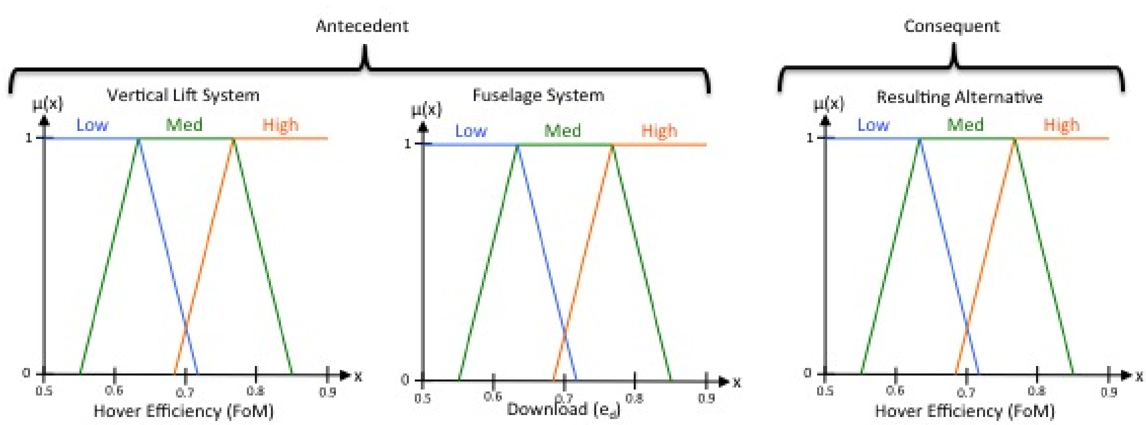


Figure 14: Example Database for an Inference System

Secondly, it is hypothesized that if physics-based methods can be used to evaluate system performance, then fuzzy inference systems can be trained with data from those simple methods. There exists a number of proposed methods [2, 127] for training fuzzy systems (learning rules and system tuning) based on existing data. These methods will be utilized to develop fuzzy models of first order physical data. Using fuzzy inputs and outputs will again allow the model to quickly take in uncertain inputs and provide results indicating uncertainty.

In addition to traditional fuzzy inference systems, another branch of fuzzy systems exists that combines fuzzy concepts and Artificial Neural Networks (ANN) to create what are referred to as Neuro-Fuzzy System (NFS). Discussed more in Section 2.2.4, these systems combine the ability of fuzzy sets to model human-like perception under uncertainty with an ANN's ability to efficiently learn patterns. It is hyothosized that Neuro-Fuzzy Systems will be easier to train and more accurate than rule-based inference systems when generating fuzzy models to evaluate system performance for both expert opinion and physics-based data.

In order to build and drive these fuzzy systems, knowledge of the available input parameters driven by architecture will be required (to define the parameters shown in Figure 12). The notion behind creating this preprocessing is based on the idea that definition of elements of a system's architecture inherently constrains or directs its input parameters, as exemplified in Figure 15. In addition to these inputs, some means to generate rules for fuzzy inference systems is needed. In order to understand the applicable inputs, their respective values, and mappings of inputs to outputs we have to ask the next research question:

Research Question 3. *How should expert data be elicited to build and drive the Fuzzy Expert System(s)?*

In order to determine the required system inputs and their values a simple survey elicitation will be used. It in conjectured that experts can be surveyed to provide the most important input parameters related to each functional aspect of the system and ranges of those inputs for each option related to an aspect. The best way to ask for input values will be explored, as well as the appropriate methods to translate those input values into fuzzy membership functions. In order to create rules for the fuzzy inference systems, rules will need to be defined. It is conjectured that experts can be simultaneously surveyed to provide simple correlations (direction and relative magnitude) between the architecturally driven inputs and design criteria, providing the necessary information to design a Fuzzy Expert System.

In addition to building fuzzy expert systems based solely on relative expert judgement, it is believed that historical data of many first order design parameters (wing loading, disk loading, power loading, L/D, etc.) could be utilized to improve system accuracy. Here the research will attempt to gather historical aircraft system data to map design parameters to decisions in architectural configuration for a number of existing aircraft, as notionally illustrated in Figure 15 (a simplified 2D example). The research will explore the best means to combine expert elected data with historically available values.

In order to create the second type of hypothesized system, one trained by physics-based data, some data will have to be used relevant to the problem at hand. This will be accomplished by generating customized data based on the specific problem being addressed using generalized sizing relations such as the R_F method [102], momentum theory, Breguet's equations, or simple sizing estimates [88]. This will allow for some elements of the concept evaluation to be customized to the specific engineering problem at hand, as well as adding physical gravity to solutions.

Lastly, in answering Research Question 2, it was hypothesized that NFSs could be trained to create accurate models. This will be explored for both physics-based models as well as expert system models. In order to train NFSs as expert systems, expert data will be needed to train the systems. It is hypothesized that if a system attribute/criterion is too complicated to be captured with just a few (2-4) inputs,

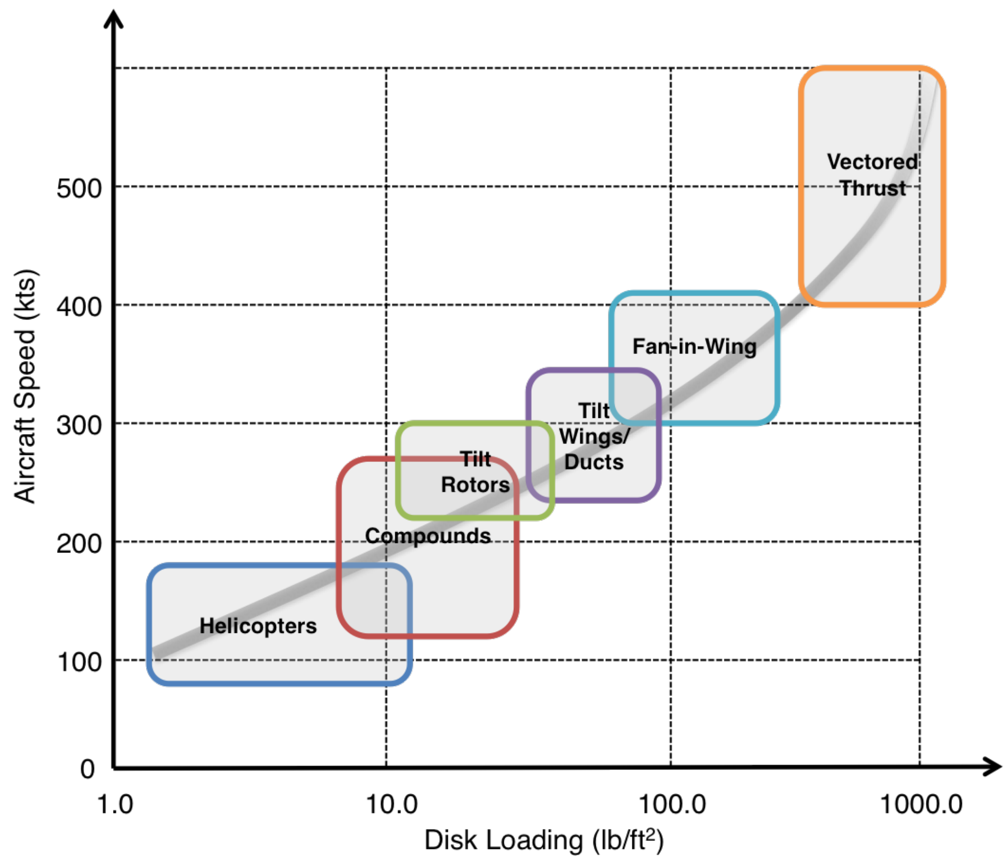


Figure 15: Notional Example of Mapping Historical First Order Design Parameters to Criteria (adapted from [31]

then expert data sets to train NFSs can be generated. This will be accomplished by having experts evaluate a set synthesized alternatives with respect to the criteria of interest. The best means of creating this data set, and number of required evaluations will be explored in this research.

1.3.3 Making a Decision

By answering Research Questions 2 and 3, a set of fuzzy models is created to quickly evaluate any feasible alternative that can be synthesized through a morphological matrix with respect to the each necessary criteria. These fuzzy models are expected to work like flexible first-order design tools, estimating a system architecture's performance with respect to a given criteria. As previously discussed, these models will then be combined into a larger framework for simultaneous multi-disciplinary analysis of compatible architectures. Then, once scores for a given alternative can be generated from the inputs of it's constituent attribute options, a means to account for the thousands or millions of feasible alternatives is still necessary. This is where MCDM or related optimization techniques will be utilized to determine the best feasible alternatives with respect to all of the criteria and any corresponding rankings. As shown in Figure 16, an optimization and decision making methodology will be used to complete the framework loop, using the morphological matrix and fuzzy inference system as a system model to quickly explore the design space. In order to accomplish this, an answer must be found to the question:

Research Question 4. *What is a feasible means to utilize the developed Fuzzy System Framework (from Research Objective 2) to explore very large design spaces and identify the best potential architectures?*

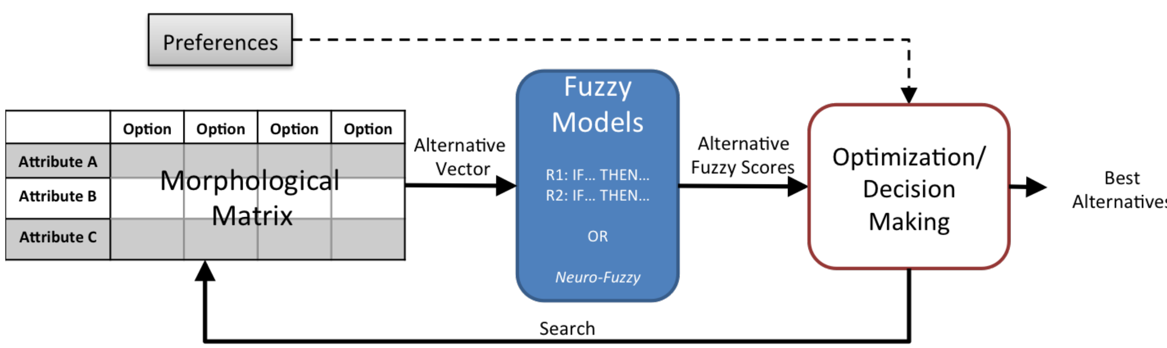


Figure 16: Framework: Combining Fuzzy System with Decision Making Methods

Given different simplicity of most fuzzy calculations, some smaller design spaces on the order of tens of thousands of potential alternatives may be able to be explored using computational brute force. This would involve evaluating every possible alternative through the fuzzy inference system while determining the best alternatives through some sort of fuzzy MADM method, such as fuzzy TOPSIS [106, 128], a weighted-sum model [122], or other approach. Other conceptual tools, such as those based on s-Pareto frontiers [73] could also be utilized to expedite finding the best solutions. However, larger design spaces with more complex quantitative fuzzy inference systems, will require more mathematical finesse, and most likely utilize MODM methods (or corresponding multi-objective optimization techniques).

Research question 4 is perhaps the most difficult to answer question for this particular research due to the large number of generally disparate MCDM methods, and the time involved in understanding and implementing any one of them. Thus, this research only seeks to establish a feasible approach for identifying concept configurations, leaving the possibility of future research for discerning better methods for the same purpose. In selecting any optimization or decision making method, a decision maker has to consider a number of issues [111]. In this case, the method should be able to use discrete inputs, as the morphological matrix does not necessarily provide for a continuous design space, but rather presents a discrete, combinatorial problem. Additionally, the approach should be able to account for multiple criteria/objectives of varying types. Perhaps the most important trait of the desired method is the need to be able to use the fuzzy inference system as a "black box" objective function. With the mathematical workings of the fuzzy system not being known in advance, using the morphological matrix/fuzzy system system without the need to know how it works is likely the ideal application. Some means may also be required to defuzzify (reduce to a crisp number) the results of the fuzzy systems to work in an optimization or framework designed for crisp problems, but the ability to consider the uncertainty in the model outputs would be ideal.

Given these requirements for the method, it is hypothesized that an evolutionary approach, specifically a genetic algorithm, to identify a Pareto frontier of best alternatives will be ideal. Genetic algorithms (GAs) are inspired by natural and biological principles, and largely based on the concept of natural selection. GAs iteratively generate and evaluate an evolving population of potential solutions using biologically principles such as inheritance, crossover, and mutation [54, 116]. A powerful optimization tool, GAs are capable of searching large, ill-structured, discrete solution spaces, and have no need to understand the mathematical foundation of the problem to consider it.

1.4 Research Methods

In order to demonstrate how the subject research is conducted, and to better illustrate how each of the research activities ties together the research progression is described here and illustrated in Figure 17. The research process will begin by definition of an example problem, outlined in Section 3.1, for application of the methods and tools to be researched. As a part of the example problem, a set of customer requirements is created to establish the need, and then be utilized to define the problem and establish value, following the first three steps Georgia Tech's generic IPPD methodology (Figure 2) [96]. A morphological matrix will also be generated at this time, as well as a compatibility matrix, which will both be carried through the entire research effort. This example problem, along with the problem definition has been purposely simplified to scale the problem to a reasonable size for this research effort. Moreover, during the creation of the morphological matrix, the size of the matrix has been purposefully managed to provide a feasible number of compatible alternatives to verify optimization analysis later in the research. Before exploring the creation of a framework, a benchmarking phase will be completed.

The first phase will include benchmarking the problem using probabilistic versions of both TOPSIS and AHP [59]. This is meant to represent the available methods that come the closest to address the identified gaps. The evaluations of alternatives completed for this phase will also provide a baseline to compare the results of fuzzy systems to later in the research process. In addition to the probabilistic methods, fuzzy set based versions of both MADM methods will be completed at the same time. This effort is expected to contextualize how fuzzy sets measure uncertainty and provide an an answer to Research Question 1.

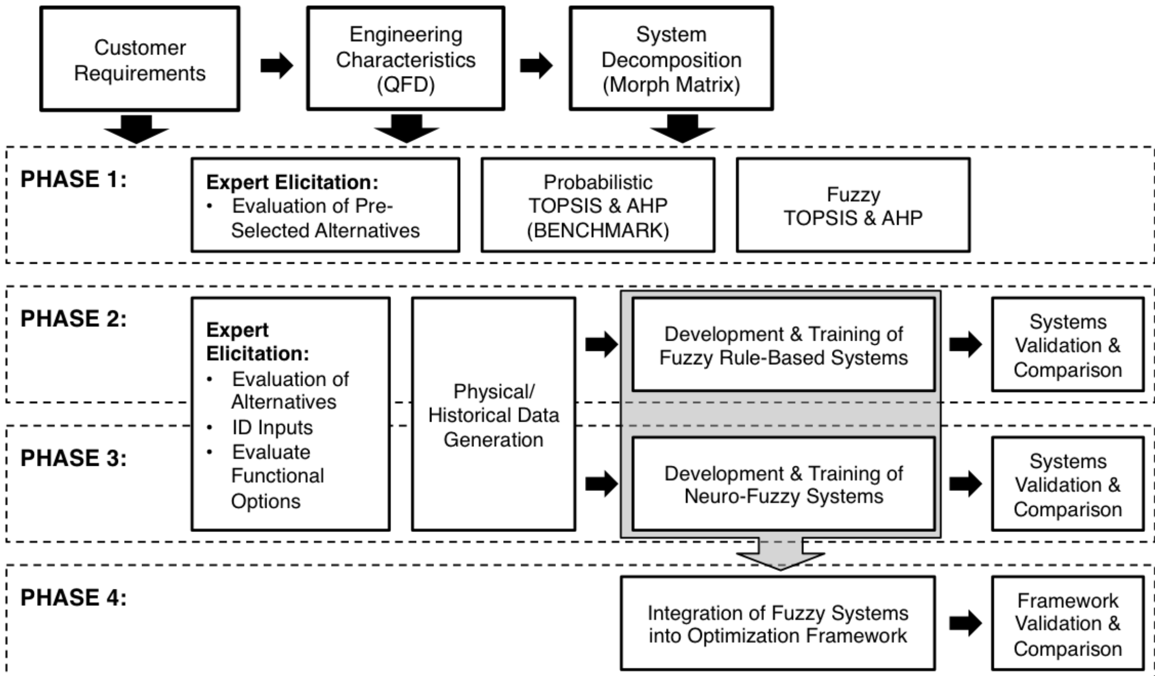


Figure 17: Research Progression

The second and third phases of the research effort are closely linked, as are Research Questions 2 and 3. These phases explore fuzzy rule based inference systems, and neuro-fuzzy systems respectively. Both phases are driven by the need for expert elicitation and collection of historical and physical data. Although these elements are addressed one at a time, they each feed into one another, with the systems' design being dependent on the data available and how its translated into membership functions, and the form of the elicitation being dependent on the required data for the systems. In these phases, various system architectures, fuzzy databases, rule bases, and reasoning mechanisms will be explored to try and understand how accurate fuzzy models might be created. These systems will be used to evaluate the pre-selected alternatives used in the benchmarking phase, and the results will be compared to the expert evaluation of those alternatives as part of the MADM methods.

In the last phase of research, the generated fuzzy models will be combined into the larger DSM-like framework. Research question 4 will then be addressed as the framework will then be used to explore the available design space defined by the morphological and compatibility matrices. During this phase, a MCDM approach using multi-objective design optimization and some decision making method(s) will be explored to try and provide some means to identify families of alternatives to carry forward into higher fidelity phases of conceptual design. Some of the best alternatives from these families can then be compared back to the benchmarked alternatives to understand if new potential architectural concepts have been identified.

CHAPTER II

TOOLS

2.1 Fuzzy Set Theory

Fuzzy sets were first introduced by Zadeh in 1965 [138], and have since been studied and used in a wide range of domains from engineering and computing to business and health sciences. A fuzzy set is a set that's members have varying degrees of membership as defined by a "membership function", often valued on the real unit interval [0,1]. Let X be some field of reference or set covering some some finite range. X may be discrete or continuous. Let A be some subset of X where individual members are permitted a degree of membership. Traditionally, degree of membership is determined by some membership function, defined as μ_A, whereby each element $x \in X$ has a degree or grade of membership in A of $0 \leq \mu_A(x) \leq 1$. Thus, an object with a membership grade of $\mu_A(x) = 1$ is considered to completely belong to the subset A. Conversely, if $\mu_A(x) = 0$, the object does not belong to the subset A [37]. This section provides an overview of some basic and necessary concepts in fuzzy set theory, including types and representations of fuzzy numbers, as well as the basics of operations on fuzzy sets.

In fuzzy set theory, traditional sets with traditional binary memberships are referred to as "crisp" sets, and can be considered a special case of fuzzy sets, with membership always being equal to 0 or 1. As such, many of the properties of crisp sets extend back to fuzzy sets, and much of the symbology and terminology of fuzzy set theory is derived from it's classical, crisp counterpart.

Fuzzy sets are used in a number of contexts, and for a number of purposes. The concept of *fuzziness* as introduced by fuzzy set theory is perhaps most often used to convey the inherent imprecision of subjective human judgement, whereby $\mu_A(x)$ is interpreted as a degree of possibility that x belongs to A [140]. While fuzzy concepts were first used in representing the uncertainty inherent to human cognitive processes, they have since been expanded to help represent many forms of incomplete or uncertain information [54]. The application of fuzzy concepts thus extend naturally to their use in this research, representing all of the vagueness and uncertainty associated with conceptual design and decision making.

This section gives an overview of the conceptual and mathematical basics of fuzzy sets as well as the methods and tools that are anticipated to be used in this research.

2.1.1 Types of Fuzzy Sets

Given the concept of a fuzzy set, a formal definition can now be given, as well some other useful related definitions. Most of the following definitions have been adapted from Dubois and Prade's definitions[37].

Definition 1. *Given some set of objects, X, with generic members $x \in X$, then the fuzzy set A in X is characterized the set of ordered pairs:*

$$A = \{(x, \mu_A(x) \mid x \in X)\} \tag{1}$$

Some specific types of fuzzy sets can be considered as especially important for the purpose of developing fuzzy concept selection methods. Fuzzy numbers can be used to represent uncertainty or imprecision about a particular value or values. A **fuzzy number** is characterized as a convex normalized fuzzy set A on the real line, $\mathbb{R}$, such that μ_A is piecewise continuous; and $\exists!\, x = m \in \mathbb{R},\ \mu_M(m) = 1$ (where m is called the mean value of A).

One of the more often used types of fuzzy numbers, the generalized L-R type fuzzy number is defined as:

Definition 2. *A **generalized L-R type** fuzzy number, $A_{LR} = \{c, a, b, d, \omega\}$, given the strictly increasing function, L, and the strictly decreasing function, R, is defined by the following membership function, given $0 < w \leq 1.0; a, b, c, d \in$ [24, 37, 47]:*

$$\mu_{A_{LR}}(x) = \begin{cases} 0 & x \leq c \\ L(x) & c < x \leq a \\ \omega & a < x < b \\ R(x) & b \leq x < d \\ 0 & x \geq d \end{cases} \tag{2}$$

One specific L-R type fuzzy number, is often referred to as a flat fuzzy number and can be used to model uncertainty or imprecision about an interval. A **flat fuzzy number**, which can be used to model a **fuzzy interval**, is a fuzzy number M such that $\exists(a, b) \in \mathbb{R}$, and $\mu_M = \omega = 1\ \forall x \in [a, b]$[37]. Fuzzy trapezoidal numbers are generalized L-R type fuzzy flat numbers with simple linear functions for L and R that are often used as a simple representation of a fuzzy interval. In the development of appropriate scoring methods for the subject framework, Definition 3 will be used to define a fuzzy trapezoidal number. Figure 18 illustrates the membership function of a trapezoidal fuzzy number.

Definition 3. *A **trapezoidal fuzzy number** is a fuzzy number, $A_I = \{a_1, a_2, a_3, a_4\}$, defined by the membership function:*

$$\mu_{A_I}(x) = \begin{cases} \frac{(x-a_1)}{(a_2-a_1)}, & a_1 \leq x < a_2 \\ 1, & a_2 \leq x \leq a_3 \\ \frac{(a_4-x)}{(a_4-a_3)}, & a_3 < x \leq a_4 \\ 0, & otherwise \end{cases} \tag{3}$$

Another extremely common type of fuzzy number, the fuzzy triangular number is a fuzzy trapezoidal number where $a_2 = a_3$. Triangular fuzzy numbers, as defined below, are often used to represent uncertainty or imprecision about a particular value. Figure 18 illustrates the membership function of a triangular fuzzy number.

Definition 4. *A **triangular fuzzy number** is a fuzzy number, $A_T = \{a_1, a_2, a_3\}$, defined by the membership function:*

$$\mu_{A_T}(x) = \begin{cases} \frac{(x-a_1)}{(a_2-a_1)}, & a_1 \leq x \leq a_2 \\ \frac{(a_3-x)}{(a_3-a_2)}, & a_2 < x \leq a_3 \\ 0, & otherwise \end{cases} \tag{4}$$

Another simple means of representing uncertainty about a single value is the Gaussian fuzzy number, based on the Gaussian (or normal) distribution [94]. The Gaussian fuzzy number is not as common as the triangular or trapezoidal due to the slightly more complex fuzzy math necessary, but can still be useful to model decision maker uncertainty. The Gaussian fuzzy number is defined below, and illustrated in Figure 18.

Definition 5. *A **gaussian fuzzy number** is a fuzzy number, $A_G = \{m, k\}$, defined by the membership function:*

$$\mu_{A_G}(x) = e^{-\frac{(x-m)^2}{2k^2}} \tag{5}$$

Each of the fuzzy numbers defined here can be useful for modeling various weights and scorings associated with various fuzzy methods. Triangular fuzzy numbers are a common means for the fuzzification of qualitative MCDM exercises. Triangular fuzzy numbers can quickly and easily model uncertainty through qualitative numerical fuzzification or linguistic fuzzification as shown in Figure 13. As mentioned previously,

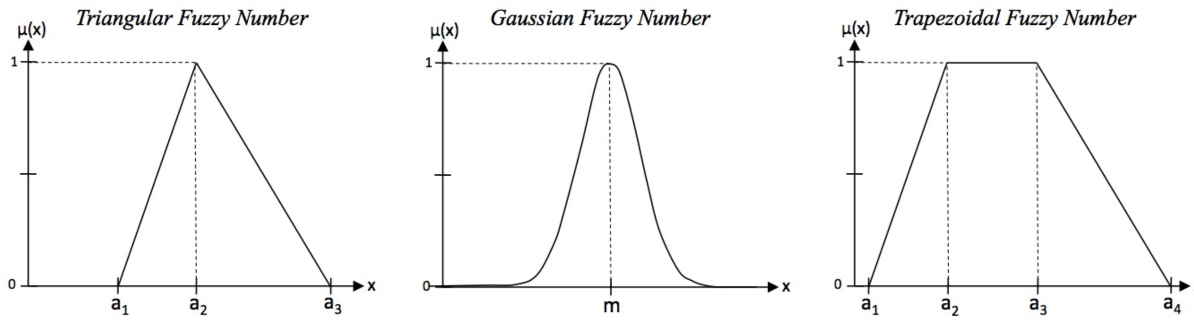


Figure 18: Types of Fuzzy Numbers

trapezoidal fuzzy numbers are particularly useful for the fuzzification of an interval, and will be explored for modeling quantitative interval estimates of engineering characteristics as assessed by subject matter experts.

2.1.2 Operations on Fuzzy Sets

A comprehensive framework of mathematical concepts and tools exists within the framework of fuzzy set theory to complement the theory and application of fuzzy sets [37]. This section presents a series mathematical concepts that are particularly necessary to the implementation of the fuzzy methods explored in this research, but is in no way a complete presentation of the calculus of fuzzy sets. Most of the fuzzy arithmetic utilized for fuzzy methods is rather simple (addition, subtraction, multiplication, division), but a few more complicated fuzzy mathematical concepts are occasionally utilized.

The classical union and intersection definitions for standard sets can be intuitively extended to fuzzy sets. The following definitions, 6 through 8, are proposed by Zadeh [37, 138].

Definition 6. *The classical union (∪) of two fuzzy sets, A and B, can be defined by the membership function:*

$$\mu_{A \cup B}(x) = max(\mu_A(x), \mu_B(x)) \; \forall x \in X \tag{6}$$

Definition 7. *The classical intersection ($\cap$) of two fuzzy sets, A and B, can be defined by the membership function:*

$$\mu_{A \cap B}(x) = min(\mu_A(x), \mu_B(x)) \; \forall x \in X \tag{7}$$

Definition 8. *The complement ($\bar{A}$) of any fuzzy set, A, can be defined by the membership function:*

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x); \forall x \in X \tag{8}$$

In addition to these most basic mathematical definitions, it is necessary to define some basic properties of fuzzy sets. Many of these properties are direct extensions of the properties of crisp sets. Proofs of the following properties of fuzzy sets can be found in the references [37, 57, 81, 138].

Table 2: Fundamental Properties of Fuzzy Set Operations

Commutivity	$A \cup B = B \cup A$
	$A \cap B = B \cap A$
Associativity	$A \cup (B \cup C) = (A \cup B) \cup C$
	$A \cap (B \cap C) = (A \cap B) \cap C$
Distributivity	$A \cup (B \cup C) = (A \cup B) \cup (A \cup C)$
	$A \cap (B \cap C) = (A \cap B) \cap (A \cap C)$
Idempotency	$A \cup A = A$
	$A \cap A = A$
Identity	$A \cup \emptyset = A$
	$A \cap \emptyset = \emptyset$
	$A \cup X = X$
	$A \cap X = A$
Involution	$\bar{\bar{A}} = A$

Fuzzy Arithmetic through α-Cuts

One approach to the arithmetic of fuzzy sets utilizes the concept of α-cuts [39, 57]. This approach to fuzzy arithmetic is based on interval mathematics, where each α-cut is a closed interval on which the membership of the given fuzzy number is at least α, as expressed in definition 9. The α-cut method can completely and uniquely represent any fuzzy number, as shown by Klir and Yuan [57]. Given this, and that α-cuts are closed on $[0, 1]$, arithmetic on fuzzy numbers can be expressed in terms of the arithmetic on its α-cuts.

Definition 9. *Given a fuzzy set, A, in X, and any real number $\alpha \in [0, 1]$, the α-cut of A, ${}^{\alpha}A$ is the crisp set [39]:*

$$^{\alpha}A = \{x \in X : \mu_A \geq \alpha\} \tag{9}$$

Using α-cuts, an expression for the four basic arithmetic operations can be formulated [57]. Letting $*$ represent any of the four basic operations (addition, subtraction, multiplication, division), and A and B be any two fuzzy numbers, any α-cut of the fuzzy set $A * B$ on can be defined:

$$^{\alpha}(A * B) =^{\alpha} A *^{\alpha} B \tag{10}$$

If $*$ is denoting division, it must be required that ${}^{\alpha}B \neq 0$ for all $\alpha \in [0, 1]$. Because a fuzzy number can be expressed as the union of all of its α-cuts (as can be shown through convergence theorem [57]), this fuzzy arithmetic can be expressed as:

$$A * B = \bigcup_{\alpha \in [0,1]} {}^{\alpha}(A * B) \tag{11}$$

Using this principle, the resulting $(A * B)$ is also a fuzzy number, and can be computationally approximated using a defined set of α-cuts. Thus a more complicated mathematical concept can be applied to the real intervals from a series of input α-cuts (i.e. [0, 0.1, 0.2, ..., 1.0]), and the result can be stitched together by interpolating between the resulting α-cuts.

Fuzzy Arithmetic Using the Extension Principle

The other common approach to arithmetic of fuzzy numbers utilizes the extension principle as introduced by Zadeh [37, 57, 58]. The extension principle is essence a fuzzy mapping, $f : \mathcal{F}(X_1, ..., X_n) \rightarrow \mathcal{F}(Y)$ of n fuzzy sets, $A_1, ..., A_n$, in $X_1, ..., X_n$ to the fuzzy set B in Y. The extension principle states:

$$\begin{aligned} &\mu_B(y) = \sup_{x_1,...,x_n \,|\, y=f(x_1,...,x_n)} min(\mu_{A_1}(x_1), ..., \mu_{A_n}(x_n)) \\ &\text{where } \ \mu_B(y) = 0 \ \text{ if } \ f^{-1}(y) = \emptyset \end{aligned} \tag{12}$$

Zadeh solved the simplest case of the extension principle, and when $n = 1$, showing that a fuzzy set can be mapped to itself in any domain, $\mu_B(y) = \mu_A(f^{-1}(y)) = \mu_A(x)$. The extension principle allows a generalization of many fuzzy operations, including the basic four operations. Letting $*$ represent any of the four basic operations and A and B represent two fuzzy numbers, then for all $z \in$, $A * B$ can be written:

$$A * B(z) = \sup_{z=x*y} min[A(x), B(y)] \tag{13}$$

Given this general relation, it can be shown that $A * B$ must be a continuous fuzzy number. Thus, the extension principle provides a second means for fuzzy arithmetic.

Each of the two methods presented here for fuzzy arithmetic, the α-cut method and the extension principle method, can be used to derive the same results when using fuzzy sets and fuzzy numbers. However, either method can be alternatively advantageous in developing computational tools to implement the fuzzy methods explored in this research. The more commonly used extension principle is often advantageous when determining simple relations between fuzzy sets. And the α-cut method can be useful for iteratively outlining the fuzzy results to more complicated mathematical procedures.

These principles can be used to derive specific equations for the arithmetic of the fuzzy numbers used more frequently in this research. Table 3 lists some common operations for the fuzzy numbers most often used in the research completed here [57, 122].

Table 3: Example Fuzzy Arithmetic

	Fuzzy Triangular	Fuzzy Trapezoidal
	$A_T = \{a_1, a_2, a_3\}$ $B_T = \{b_1, b_2, b_3\}$	$A_T = \{a_1, a_2, a_3, a_4\}$ $B_T = \{b_1, b_2, b_3, b_4\}$
α-cut	${}^{\alpha}A_T = [(a_2 - a_1)\alpha + a_1, a_3 - (a_3 - a_2)\alpha]$	
Addition	$A_T + B_T = \{a_1 + b_1, a_2 + b_2, a_3 + b_3\}$	$A_T + B_T = \{a_1 + b_1, a_2 + b_2, a_3 + b_3, a_4 + b_4\}$
Subtraction	$A_T - B_T = \{a_1 - b_1, a_2 - b_2, a_3 - b_3\}$	$A_T - B_T = \{a_1 - b_1, a_2 - b_2, a_3 - b_3, a_4 - b_4\}$
Multiplication	$A_T \times B_T = \{a_1b_1, a_2b_2, a_3b_3\}$	$A_T \times B_T = \{a_1b_1, a_2b_2, a_3b_3, a_4b_4\}$
Inverse	$1/A_T = \{\frac{1}{a_3}, \frac{1}{a_2}, \frac{1}{a_1}\}$	$1/A_T = \{\frac{1}{a_4}, \frac{1}{a_3}, \frac{1}{a_2}, \frac{1}{a_1}\}$

Ranking Fuzzy Sets

An important fuzzy concept that is critical to fuzzy decision making is a means for ranking fuzzy sets. As the name suggests, at the heart of any MCDM method is the need for a decision, and in order to make a decision, some means of preference must be derived. Thus, the process of any MCDM problem inherently involves some means of ranking various alternatives as a means to differentiate those most preferred. Ranking fuzzy sets is a much less straightforward problem than ranking crisp values, and a number of means have been proposed to rank fuzzy sets [6, 16, 17].

One commonly accepted means to rank fuzzy numbers was conceived by Baas and Kwackernaak, among the earliest progenitors of fuzzy MCDM [6]. Their given method is relatively simple to understand and easy to implement, especially with triangular or trapezoidal fuzzy numbers. Given two fuzzy numbers, A and B, a dominance measure is calculated as:

$$\delta(A,B) = \bigcup_x (\mu_{\leq A}(x) \cap \mu_B(x)) = \max_x(\min(\mu_{\leq A}(x), \mu_B(x))$$

$$\text{where: } \mu_{\leq A} = \begin{cases} 1.0 & for \;\; x \leq x^* \\ \mu_A(x) & for \;\; x > x^* \end{cases} \tag{14}$$

Here x^* is defined as $x^* = min(x|\mu_A(x) = 1.0)$, as illustrated in Figure 19, along with the fuzzy set $\leq A$. Both dominances, $\delta(A, B)$ and $\delta(B, A)$, are calculated and A is said to dominate B iif $\delta(A, B) = 1.0$ and $\delta(A, B) < Q$, where Q is some fixed positive value less than one. Q was varied by Baas and Kwackernaak through the range $[0.7 - 0.95]$ to understand its sensitivity [119, 122]. Most of the methods explored in this research, particularly those using triangular and trapezoidal fuzzy numbers will use $Q = 0.95$ to ensure rankings are achieved where necessary.

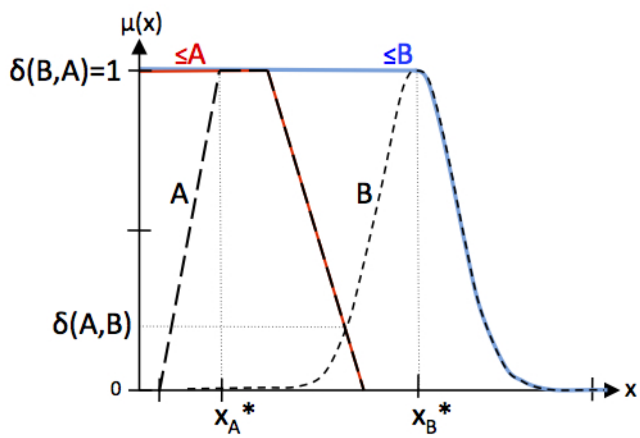


Figure 19: Dominance in Fuzzy Sets

Inherently related to the problem of ranking fuzzy numbers is the problem of measuring the distance between fuzzy numbers. While obtaining a fuzzy measure of the distance between two fuzzy numbers can be as simple as subtracting the two numbers as discussed above, repeating this fuzzy arithmetic can often lead to skewed or distorted results. Careful consideration should be given to the final results of a mathematical process when determining fuzzy distances. Furthermore, crisp results are often necessary to aid in the ranking of fuzzy sets, which adds more difficultly to the issue. A number of distance measures between fuzzy sets have been proposed by various authors[47, 120], some of them fuzzy and some of them crisp. Because of the large diversity in possible distance measures, any additional distance measures necessary for the fuzzy techniques explored in this research will be outlined as needed.

2.2 Fuzzy Systems

A fuzzy system is a "static or dynamic system which makes use of fuzzy sets or fuzzy logic and of the corresponding mathematical framework" [7]. A fuzzy system can utilize fuzzy sets in a number of different ways, including fuzzy inputs and/or outputs, fuzzy parameters within the system ($f(\mathbf{x}) = \tilde{2}x_1^3 - \tilde{6}x_2^2 + \tilde{7}x_3$). Most commonly, however, fuzzy sets are used with a fuzzy rule base describing the relationship between inputs and outputs. For instance, the following generic statement can be part of a fuzzy system:

IF *the aspect ratio is high* ***THEN*** *the lift-to-drag ratio will be high.*

This research focuses in part on this last type of fuzzy system, often referred to as a fuzzy inference system, a rule-based fuzzy system, or a fuzzy expert system [51]. These types of systems are often used in artificial intelligence systems and are usually an attempt to form a model of the knowledge of subject matter experts in linguistic terms, when a precise, crisp mathematical model is not available, not necessary, or too complicated to construct [81]. A fuzzy inference system can have fuzzy or crisp inputs (also referred to as the antecedents), but usually produces a fuzzy output (or consequent). The mathematical framework relating the inputs to the outputs is completed through some means of fuzzy inference or fuzzy calculus. This framework most often takes the form of a series of if-then rules of the following general form:

$$\mathcal{R}_i: \quad \textbf{If } \tilde{x} \text{ is } A_i \textbf{ Then } \tilde{y} \text{ is } C_i \quad i = 1, 2, ..., n \tag{15}$$

In this form, $\tilde{x}$ is a fuzzy antecedent (input) variable, and A_i are the antecedent constant terms, while $\tilde{y}$ is the consequent (output) and B_i are the consequent constant terms. The values $\tilde{x}$, $A_i(x)$, $\tilde{y}$, and $B_i(y)$ are fuzzy sets defined in their respective domains, $x \in X$ and $y \in Y$. In a fuzzy system based on the input of subject matter experts, the terms A_i and B_i, referred to as the "database", are often pre-defined linguistic sets of terms ("small", "large", etc.). The combination of the rule base, $\mathcal{R}_i | i = 1, 2, ...n$, and the antecedent and consequent terms, $A_i \in X$ and $B_i \in Y$, form the knowledge base of the fuzzy system [7].

The mathematical framework that allows for implementation of the rule base can take virtually any form of fuzzy inference or fuzzy mathematics that makes sense for the given model. The general process for this is a function relating the antecedent to the consequent, $f : X \to Y$. Thus each fuzzy rule, $\mathcal{R}_i$, is just a relation, $\mathcal{R}_i(x, y) : A(x) \to B(y)$, translating a some membership function in the domain X to its counterpart in the domain Y:

$$\mu_{\mathcal{R}} = F(\mu_A, \mu_B) \tag{16}$$

The procedure of inferring this relation from a set of if-then rules is often referred to as the "composition rule of inference", and the function F represents either a fuzzy implication, or some conjunction operator (through the process is often referred to generally as "implication") [139].

2.2.1 Mamdani Fuzzy Model

The Mamdani fuzzy inference system, also called "max-min" composition, is among the most common and widely used fuzzy inference systems [7, 51]. The system was developed as a linguistic if-then system by Mamdani, in 1975, to to control a steam engine and boiler system [68]. A minimum conjunction operator is used for the implication of inputs into outputs.

$$\mathcal{R}_i(x,y) : \mu_{\mathcal{R}_i}(x,y) = \min(\mu_{A_i}(x), \mu_{B_i}(y)) = (\mu_{A_i}(x) \wedge \mu_{B_i}(y)) \quad i = 1, 2, ..., n \tag{17}$$

The fuzzy system of all of these rules is then created by the union of the n individual rules, which is defined in fuzzy sets as the maximum.

$$\mathcal{R} = \bigcup_{i=1}^{n} \mathcal{R}_i \tag{18}$$

$$\mu_{\mathcal{R}}(x,y) = \max_{1 \le i \le n} (\mu_{A_i}(x) \wedge \mu_{B_i}(y)) \tag{19}$$

The result of the rule base contained within the fuzzy relation R is then often written using the composition operator, $\circ$.

$$\tilde{y} = \tilde{x} \circ \mathcal{R} \tag{20}$$

Composition can be demonstrated for the desired case of fuzzy inputs by supposing an input fuzzy set, A', and it's fuzzy output B'. The fuzzy relation is defined by its membership function, which is substituted into the composition. With the maximums and minimums taken over different domains, the associative property allows us to rewrite this new relation to define a crisp value, $\alpha_i = \max_X[\mu_{A'}(x) \wedge \mu_{A_i}(x)]$, often called the *degree of fulfillment*, or *firing strength*, of the i^{th} rule's antecedent. This process is illustrated in Figure 20.

$$B' = A' \circ \mathcal{R} \tag{21}$$

$$\mu_{B'}(x,y) = \max_{X}\left(\mu_{A'}(x) \wedge \max_{1\leq i\leq n}[\mu_{A_i}(x) \wedge \mu_{B_i}(y)]\right) \tag{22}$$

$$\mu_{B'}(x,y) = \max_{1\leq i\leq n}\{\max_{X}[\mu_{A'}(x) \wedge \mu_{A_i}(x)] \wedge \mu_{B_i}(y)\} = \max_{1\leq i\leq n}\{\alpha_i \wedge \mu_{B_i}(y)\} \tag{23}$$

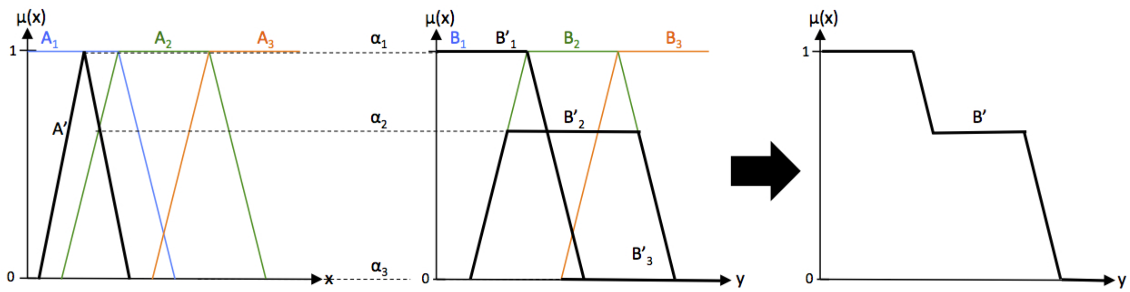


Figure 20: Mamdani Fuzzy Inference (Single Input)

So far, only a single antecedent rule and single output system has been considered (Single Input Single Output (SISO)), while in reality is more likely the fuzzy systems developed in this research will cover multiple inputs and single outputs (Multi-Input Single Output (MISO)). A glsMISO is written as a set of multiple input single output (talk) models, for p inputs, each of which is most commonly written in it's conjunctive form [7]:

$$\mathcal{R}_i: \quad \textbf{If } \tilde{x}_1 \text{ is } A_{i1} \textbf{ and } \tilde{x}_2 \text{ is } A_{i2} \textbf{ and } ... \textbf{ and } \tilde{x}_m \text{ is } A_{im} \textbf{ Then } \tilde{y} \text{ is } C_i \quad i = 1, 2, ..., n \tag{24}$$

The output can then be obtained through intersection of the composition of the rules for each input. Following the same approach as the single input case, it can be shown that the antecedent's degree of fulfillment can be determined by the minimum of the composition of each input, A_j, and the portion of the rule base it corresponds to, $\mathcal{R}_j$ [7]. The result is the minimum of each input's individual degree of fulfillment.

Evaluation of a MISO model rule base is illustrated in Figure 21.

$$B' = \min_{1 \leq j \leq m} \left(A'_j \circ R_j \right) \tag{25}$$

$$\alpha_j = \min_{1 \leq j \leq m} \{ \max_X [\mu_{A'_j}(x) \wedge \mu_{A_{ij}}(x)] \} = \min\{\alpha_{ij}\} \tag{26}$$

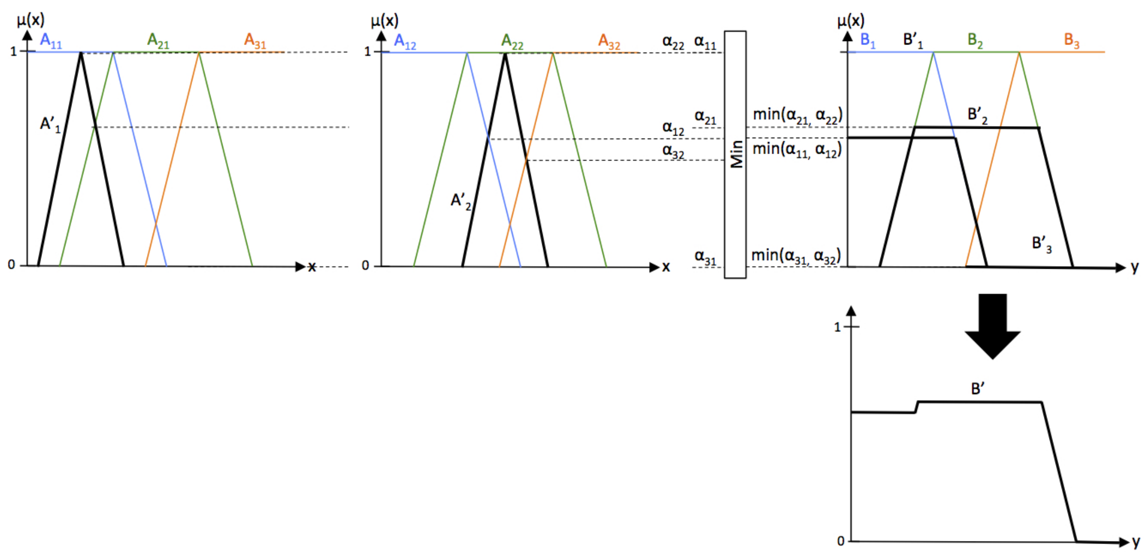


Figure 21: Mamdani Fuzzy Inference (Multiple Inputs)

Linguistic rules in a fuzzy system can take a number of logical forms. In the case of disjunction ("**or**") in a rule, the fuzzy maximum operator can be used in a means similar to equation 17, $\mu_{\mathcal{R}_i}(x, y) = (\mu_{A_i}(x) \vee \mu_{B_i}(y))$. The same process can be applied as in the case of the MISO model, giving a maximum degree of fulfillment, $\alpha_i = \max\{\alpha_{ij}\}$. A rule that requires a negation ("**not**") can be found using the complement operator, giving the rule membership function $\mu_{\mathcal{R}}(x, y) = (1 - \mu_{A_i}(x)) \wedge \mu_{B_i}$. Distributivity of the minimum and maximum operators continues to apply in these cases. The system allows for grouping the implementation of the rules into any logical order to determine the course of the antecedent [81].

2.2.2 Larsen Fuzzy Models

Similarly to the Mamdani model, the Larsen fuzzy inference model is another conjunction operator for implication in fuzzy systems. The Larsen model uses the product operator for implication, however, rather than the minimum operator used in the Mamdani model.

$$\mathcal{R}_i(x, y) : \mu_{\mathcal{R}_i}(x, y) = (\mu_{A_i}(x) \cdot \mu_{B_i}(y)) \quad i = 1, 2, ..., n \tag{27}$$

Composition for the Larsen model is identical to the Mamdani method with the exception of the implication into the consequent space. Degrees of fulfillment are still calculated for each input rule, and the minimum degree of fulfillment is then used, as shown in equation 26 to calculate the result of the consequent. It can be shown that the final result is similar to equation 23, but using the product implication, as shown in Figure 22.

$$\mu_{B'}(x, y) = \max_{1 \le i \le n} \{\alpha_i \cdot \mu_{B_i}(y)\} \tag{28}$$

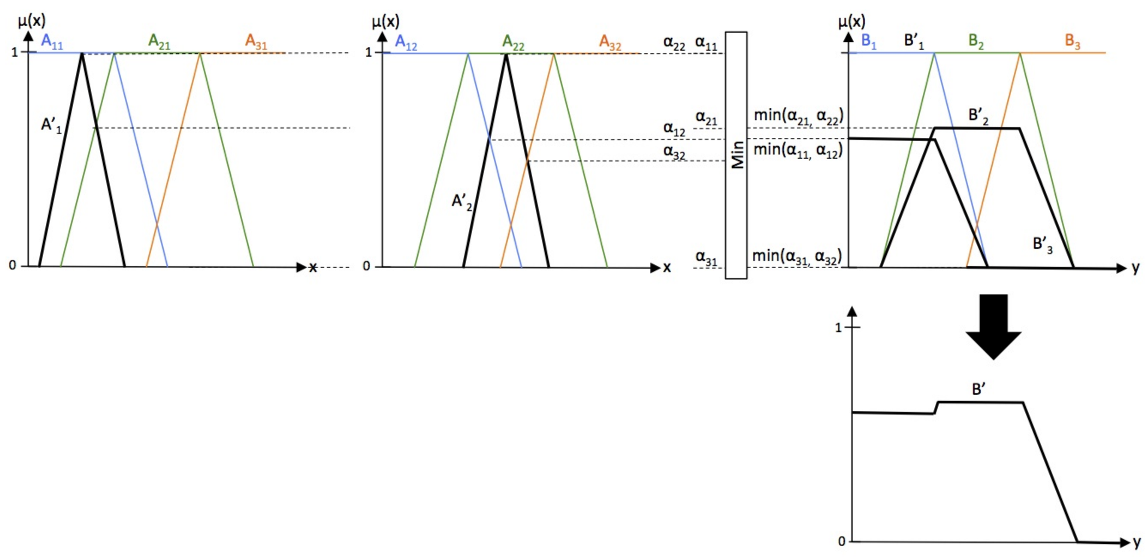


Figure 22: Larsen Fuzzy Inference (Multiple Inputs)

2.2.3 Other Fuzzy Models

There are several other common fuzzy models that, for reasons of suitability, were not considered as feasible for the framework being developed in this research. The Sugeno fuzzy model, sometimes called the Takagi-Sugeno fuzzy model, or TSK fuzzy model, was developed to generate a fuzzy system from a dataset [7, 113]. The output of the system is a crisp function, usually a polynomial based on the results of the antecedent.

$$\mathcal{R}_i : \quad \textbf{If } \tilde{x} \text{ is } A_i \textbf{ Then } y = f(x) \quad i = 1, 2, ..., n \tag{29}$$

Because a Sugeno model outputs a crisp function, the model cannot propagate fuzziness from inputs or the rules. Additionally the model's inference system is not conducive to fuzzy inputs. As a common fuzzy model, the Sugeno model bears mentioning, however for these reasons it is unlikely to be a useful fuzzy model in the context of this research.

The Tsukamoto fuzzy model has a consequent that has a monotonical membership function. Each rule produces a crisp output based on the firing strength of the antecedent, and the result is the weighted average of the rules' outputs. Once again, this model isn't likely to be useful as the desired system output is fuzzy to capture uncertainty inherent to alternatives.

2.2.4 Neuro-Fuzzy Systems

Various methods and techniques to develop intelligent systems have different properties, strengths and weaknesses. Fuzzy concepts and fuzzy logic provide a means for human-like inference under uncertainty, and are generally good at reasoning and providing a mathematical explanation for their decisions. However, they are not particularly suited for other desirable characteristics of intelligent systems, such as learning and adaptation. Hybrid systems have been proposed to account for these short coming and take advantages of the strengths of fuzzy systems. The combination of fuzzy systems with computational glassANN has been a particularly strong area of research. Neural networks complement the strengths of fuzzy systems by providing a technique with capabilities like learning, adaptability, and fault-tolerance [43]. These hybrid systems take may forms and are often referred to as fuzzy-neuro or glassNFS.

Neuro-Fuzzy Systems can involve many different ways to combine fuzzy systems with ANNs, and can be created through a number of different means. Most commonly, however, neuro-fuzzy systems refer to a hybrid system created by incorporating fuzzy system concepts into neural networks [43, 123]. This maintains the humanistic aspects of fuzzy systems, while taking advantage of the learning and structural aspects of neural networks to generate fuzzy rules, build membership functions, and tune the system. Originally conceived in the early 1990's [123] to aide in process control, neuro-fuzzy systems have since expanded, with many different architectures proposed, and applications in a number of fields.

Figure 23 illustrates a simple notional neuro-fuzzy system that could be used to represent a Mamdani type inference system with two inputs and a single output. In this simple example, the first layer (input layer) represents in the inputs to the fuzzy system. The second layer of neurons represent the antecedent database, with each neuron having an activation function corresponding to an antecedent membership function. The third layer represents the rule base, with the neuron weights representing the *normalized degree of confidence* of that particular rule. The fourth layer represents the consequent, with each neuron again having a membership function for its activation function. Finally, the output layer can provide a defuzzification of the fuzzy output of the system.

Because the system is in essence a multi-layer ANN, learning algorithms such as back-propagation can be applied to "train" the fuzzy system to better perform its desired task. The membership functions of the antecedent and consequent are

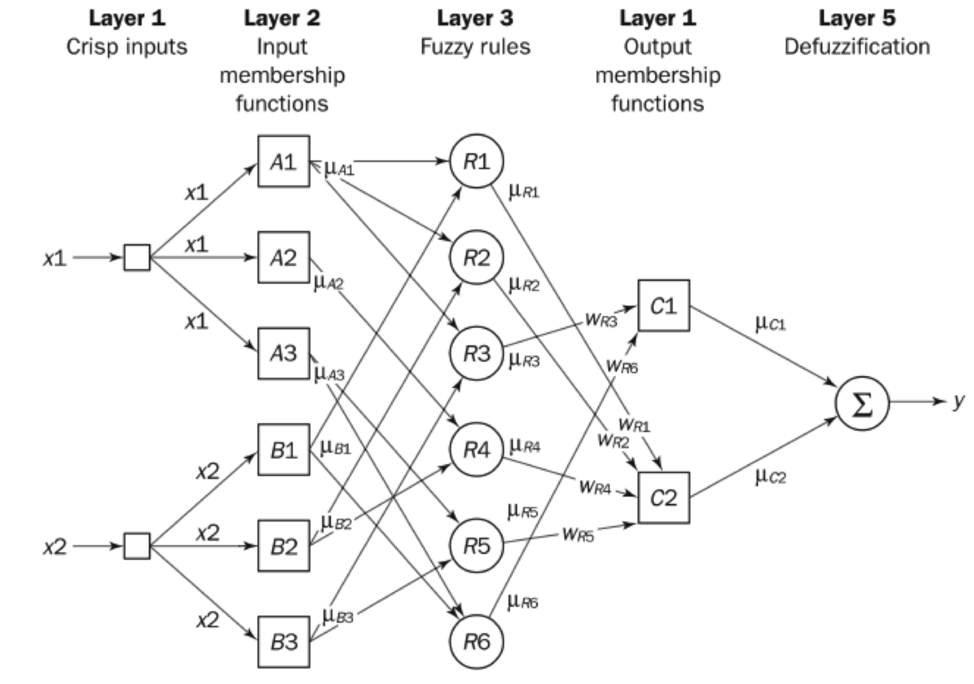


Figure 23: Simple Notional Neuro-Fuzzy System [81]

variable in nature, and are often tuned in the same manner as synaptic weights in a traditional ANN. The normalized degree of confidence for each rule can flux in the same manner to allow the rule base to be tuned with actual data.

The architecture of a neuro-fuzzy system can change drastically to represent a variety of different fuzzy system implementations, or provide varying degrees of interoperability between the neural network and fuzzy system.

2.3 Multi-Criteria Decision Making

Multi-Criteria Decision Making (MCDM) is a category of methodological processes and tools for modeling complex engineering problems where decisions must be based on multiple, possibly conflicting, objectives and constraints (the criteria). A large variety methods have been developed in the 40 years since MCDM's inception, and a vast amount of literature exists, proposing, explaining and expanding various MCDM methods, as well as suggesting methods to use for various problems [35, 37, 54, 94, 121]. Generally MCDM problems fall under one of two categories: Multi-Attribute Decision Making (MADM), where a best alternative is selected from among a discrete many; and Multi-Objective Decision Making (MODM) where the best alternative is created from a given, and often continuous, design space.

In general, an MCDM problem can be defined by minimizing a set of m criteria:

$$\min_{S}[f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x})] = F(\mathbf{x}) \tag{30}$$

Subject to a set of l constraints, defining a feasible region S:

$$S = \{x \in X | g_i(\mathbf{x}) \leq 0\}, \quad i = 1, ..., l \tag{31}$$

The n-dimensional vector of decision (or design) variables, $\mathbf{x}$, can be continuous (MODM) or predetermined based on existing alternatives (MADM). The decision maker seeks optimum solutions, $\mathbf{x}^*$, such that $F(\mathbf{x}^*) > F(\mathbf{x})$, for all $x \in S$. Furthermore, $\mathbf{x}^*$ may be defined as a non-dominated (or Pareto-optimal) solution, iff no $x \in S$ exists where $f_i(\mathbf{x}) \geq f_i(\mathbf{x}^*)$ for all i, but there exists some $j \neq i$, such that $f_j(\mathbf{x}) \geq f_j(\mathbf{x}^*)$, where $1 < j < n$. That is to say, that a non-dominated solution is one in the feasible decision space, where no gain may be made with respect to any criteria without resulting in loss to another.

The first category of MCDM methods, Multi-Attribute Decision Making (MADM) methods, solve problems with discrete decision spaces and countable few and predetermined alternatives to decide among. Here "attribute" indicates a defined state or characteristic of the system by which that system can be evaluated [9]. MADM methods are generally mathematically simple, requiring some combination of decision maker evaluations with respect to the different attributes (or criteria) and attribute weights. Perhaps, as a result of this, MADM is the most commonly used MCDM approach, with a large number of proposed methods to choose from.

Multi-Objective Decision Making (MODM) methods make up the second category, and often assume a continuous solution space and solve for optimal compromise solutions using more elegant mathematical models. In a MODM problem there are often an unmanageably large number of feasible alternatives, and the method designs the best alternative by synthesizing possible solutions based on the criteria preferences. Because of this there are no predefined alternatives, and the decision maker must use some mathematical framework to optimize their own solutions. Thus, "objective" here refers to a goal or direction of improvement for the as yet undetermined system [9]. MODM often involves both alternative identification and selection. Because of their similar nature, MODM approaches often mirror multi-criteria optimization methods.

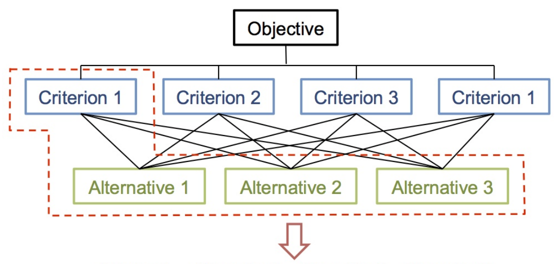
Though there is some overlap, MODM methods are often sorted into three categories: *a priori methods*, where preference information is articulated prior to running the solution algorithm; *a posteriori methods*, which generally attempt to identify a Pareto optimal set for the decision maker to select from; and *interactive methods*, where the decision maker articulates preferences as information from the solution algorithm becomes available.

Because the various MCDM methods have different assumptions, different information requirements, and different rules and constraints, selection of the appropriate MCDM methods for a particular problem is often, in essence, an MCDM problem unto itself [111]. Traditional, crisp, MCDM approaches utilized in this research will be limited to more classical approaches most commonly used in concept selection methods, namely TOPSIS and AHP.

2.3.1 Traditional AHP and TOPSIS

AHP

Analytic Hierarchy Process (AHP) is based on pairwise comparisons of a hierarchical structure of alternatives and criteria. Figure 24 illustrates a generic outline of AHP, where the decision objective or objectives are broken into a series of criteria, which are then used to evaluate alternatives in a series of pairwise comparisons, often using qualitative measures. The pairwise comparisons are postulated to make it easier for decision makers to accurately assess larger numbers of alternatives and obtain better results. Criteria can also be weighted using a pairwise comparison of the criteria themselves. AHP has been extensively reviewed and developed, and is a frequently used MADM method spanning numerous applications.



Criterion 1	Alternative 1	Alternative 2	Alternative 3
Alternative 1	1/1	2/1	1/4
Alternative 2	1/2	1/1	1/3
Alternative 3	4/1	3/1	1/1

Figure 24: Analytic Hierarchy Process

Equation 32 shows a nominal pairwise comparison matrix for n alternatives (or criteria). Each element of the matrix, p_{ij}, represents the preference or superiority of alternative or criterion i to that of j with respect to the particular criterion or objective associated with the matrix. As such, the matrix is ideally a reciprocal matrix, where for any element $p_{ij} = 1/p_{ji}$. Pairwise priority can be established through qualitative means, such as a $[\frac{1}{9} - 9]$ scale, or through a quantitative means associated with the given criterion.

$$P_k = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \cdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix} \tag{32}$$

Once the pairwise comparison matrices are complete, a prioritization method can be applied to derive alternative priority vectors. The two most common prioritization methods are the additive normalization method and the eigenvector method. Other methods that have been proposed to solve for priorities from the comparison matrices include those using direct/weighted least-squares, logarithmic least-squares, goal programming, normalization of reciprocals of column sums, arithmetic mean of column sums, and others. Various studies illustrate advantages and disadvantages to various methods, while some authors claim combinations of methods work best[70, 108, 137]. The same method can be used to determine weightings, $W = [w_1, ..., w_m]$, from the pairwise matrix comparing the criteria.

Applications of crisp AHP in this effort utilize additive normalization to calculate priority vectors. This method involves normalizing the comparison matrices by the sum of each column and then averaging the normalized rows, as shown in equations 33 and 34. This process is repeated for each criterion and objective.

$$a'_{ij} = \frac{a_{ij}}{\sum_{i=1}^{n} a_{ij}}, \quad i, j = 1, ..., n \tag{33}$$

$$v_i = \frac{1}{n} \sum_{j=1}^{n} a'_{ij}, \quad i = 1, ..., n \tag{34}$$

The priority vector, $v = [v_1, ..., v_n]$, for each comparison matrix quantifies the alternative priorities with respect to that particular criterion. Once all of the vectors have been calculated, they are multiplied with the criteria weights, $W = [w_1, ..., w_m]$, to determine the overall priority for each alternative, as shown in equation 35.

$$s_i = \sum_{k=1}^{m} v_{ik} w_k, \quad i = 1, ..., n \tag{35}$$

TOPSIS

Technique for Order Preference by Similarity to an Ideal Solution (TOPSIS) is based on the premise that the best alternative will be the one with the shortest Euclidian distance to the ideal alternative, and furthest from the negative-ideal solution. The ideal and negative-ideal solutions are formed as a composite of the best and worst criterion scorings from all of the possible alternatives.

Supposing a MADM problem with n alternatives and m decision criteria or attributes, TOPSIS is performed in seven general steps [128]:

1. Each alternative, B_i, is evaluated and scored with respect to each criteria, , C_j, forming a decision matrix $X = (x_{ij})_{n \times m}$. A matrix of criteria weights may also be identified as $W = (w_1, ..., w_m)$.

2. The decision matrix is normalized by the square root of the sum of squares of all alternatives for each criterion:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{k=1}^{n} x_{kj}^2}}, \quad i = 1, ..., n;\ j = 1, ..., m \tag{36}$$

3. The weighted normalized decision matrix, $D = (d_{ij})_{n \times m}$, is then calculated, where:

$$d_{ij} = w_j r_{ij}, \quad i = 1, ..., n;\ j = 1, ..., m. \tag{37}$$

4. The positive and negative ideal solutions are calculated. In this case, the criteria are all assumed to be performance or benefit criteria, where a greater score/evaluation is more advantageous.

$$B^* = \{d_1^*, ..., d_m^*\} = \{max_i(d_{ij}) \,|\, i \in 1, ..., n\} \tag{38}$$

$$B^- = \{d_1^-, ..., d_m^-\} = \{min_i(d_{ij}) \,|\, i \in 1, ..., n\} \tag{39}$$

5. The Euclidian separation between each alternative and the positive and negative ideal are then calculated as follows,

$$S_i^* = \sqrt{\sum_{j=1}^{m}(d_{ij} - d_j^*)^2},\ i = 1, ..., n \tag{40}$$

$$S_i^- = \sqrt{\sum_{j=1}^{m}(d_{ij} - d_j^-)^2},\ i = 1, ..., n. \tag{41}$$

6. A measure of each alternative's relative closeness to the ideal solution is then defined and calculated as

$$RC_i = \frac{S_i^-}{S_i^+ + S_i^-},\ i = 1, ..., n. \tag{42}$$

7. Alternatives are then ranked with respect to their relative closeness, where a higher score is relatively closer to the ideal solution, and thus a better alternative.

2.3.2 Probabilistic Approaches to Traditional MADM

In order to answer the first research question, and provide a benchmark to the proposed concept selection method, a probabilistic approach to MADM is used. Following in the theme of this research, this approach is used as a way to consider inherent uncertainty while using the traditional MADM approaches that might be conventionally applied to the example problem in Section 3.1. Probabilistic approaches to AHP and TOPSIS in this research are accomplished through Monte Carlo simulation as outlined in Figure 25 [11, 59]. First, some means is used to define the problem, generating criteria to evaluate each alternative by, and determine value for each criteria. Alternative concept configurations are then selected, and evaluations are elicited from experts with respect to each criteria. The generated weights and expert evaluations are used to generate Probability Density Functions (PDFs) for each required weight and score (specifically, uniform and triangular distributions are applied). PDFs are also created for each comparison within the alternative pairwise matrices to complete AHP. For the purpose of consistency, pairwise matrices for criteria were not utilized, but rather the same weighting vector was used for both AHP and TOPSIS processes.

Within the Monte Carlo Simulation (MCS), each required weight, pairwise comparison, and/or score is generated from the PDFs. The deterministic AHP and TOPSIS methods are then performed with the crisp values generated. Alternatives are ranked, and the results are recorded. After a set number of iterations are complete, the results can then be visualized and analyzed, and the most desirable alternatives selected.

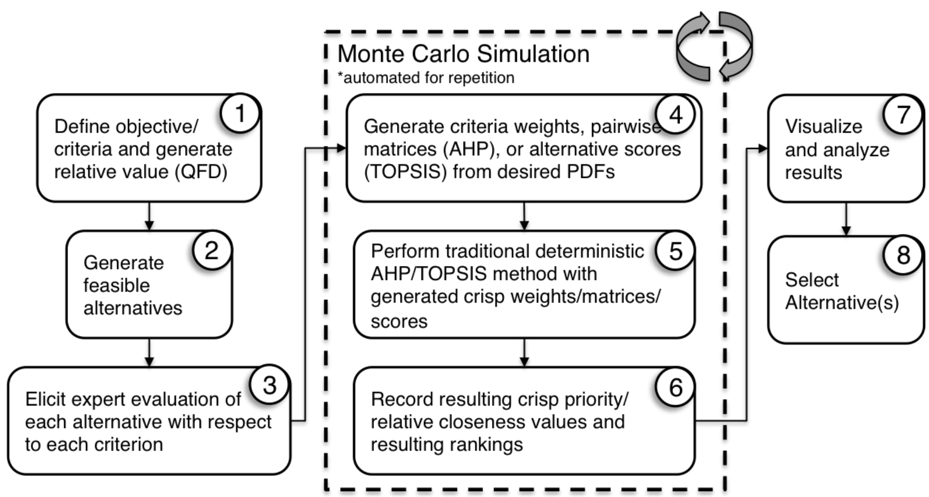


Figure 25: Probabilistic AHP and TOPSIS Methodology

2.3.3 Fuzzy Multi-Criteria Decision Making

In general, classic MCDM achieve unambiguous results through the utilization of crisp values to score alternatives and weight criteria. However, in real-life decision making, precise, certain, and easily agreed upon data is rarely encountered. This is especially true in the early stages of the design process, when subjective expert evaluation of notional concepts is by and large the only means available. In many cases the anticipated performance or effectualness of alternatives is imprecise, and can only be expressed in qualitative linguistic terms or as potential intervals. Because the systems being evaluated are most likely only notional, the information available is also inherently uncertain in nature. This is even true of criteria weights, which are usually derived from the anticipated wants and needs of the customer, and are often subject to change. Additionally, alternative evaluation and criteria weights are almost always subject to disagreement among even expert decision makers.

These inherent problems with classic MCDM techniques lend themselves perfectly to the inclusion of fuzzy set theory in MCDM. Authors Baas, Kwakernaak, Bellman, Zadeh, and Zimmermann all introduced fuzzy sets into the field of MCDM through various means in the 1970s [6, 54]. In the following decades, dozens of methods have been proposed to incorporate fuzzy concepts into all manner of MCDM techniques through every phase of decision making, with substantial work still being done in the field today.

Introducing fuzzy concepts into MCDM methods adds significant complexity when compared to their crisp counterparts, and can result in drawbacks and limitations. Commonly, fuzzy MCDM methods are limited by the nature of the fuzzy math models available. Sometimes there is no fuzzy mathematical model available to represent the original crisp MCDM problem, and even when one exists, it is most commonly more complicated than its crisp counterpart. Additionally it is sometimes difficult, or at least more complicated, to rank alternatives represented by fuzzy sets. Consequently it can be difficult to measure the quality of the resulting solution [54].

MADM approaches often provide the easiest opportunity to integrate fuzzy methods, with membership functions easily applied to represent uncertainty in a decision matrix or preference weights. Various means for applying fuzzy scoring have been explored, as well as their applications. AHP, TOPSIS, Outranking Methods, Weighted-Sum models, and most of the classic MADM methods have all been expanded to incorporate fuzzy methods, often in multiple ways.

Performing MODM methods, or searching ill-structured solution spaces with FMCDM under fuzziness is often a more challenging process, however [54]. Approaches such as fuzzy multi-objective goal programming, fuzzy multi-objective linear programming, and other fuzzy multi-objective dynamic programming approaches are common, though not as applicable to the problem presented within this research. Other approaches, such as stochastic methods like genetic algorithms or particle swarm optimization have a growing number of approaches recently. Still, this area remains a rich topic of current research across a number of fields and applications.

Fuzzy AHP

As one of the most common MADM methods, AHP was one the first and most popularly utilized methods to see the development of corresponding fuzzy MADM techniques. The majority of approaches, however, result in crisp final priority scores. Propagating the uncertainty of expert evaluations and criteria preferences was one of the primary goals of this process, so in addition to alternative rankings, a means to assess the uncertainty of those rankings (fuzzy final priority scores) is desired. Reducing the fuzzy inputs to a crisp output runs contrary to this, and methods with fuzzy outputs were emphasized. Moreover, a comprehensive, widely accepted α-cut based method for fuzzy AHP was not found. This is likely due to the scale of optimization problem necessary to solve the combinatorial problem posed by $m + 1$ pairwise matrices

Chang's extant analysis [54] method is a common approach to fuzzy AHP, using an addictive normalization method on the pairwise matrices and then determining priority vectors through evaluating the degree of possibility that any alternative could be better than all the others for a given criteria. However, this method results in crisp priority vectors and criterion weights, and thus crisp final priority scores, and was not utilized. Van Laarhoven and Pedrycz's approach to fuzzy AHP [58] is able to solve for varying numbers of evaluations in a single reciprocal matrix, and provides a fuzzy final priority vector. The method uses a logarithmic least squares approach to determining the priority vector by solving a series of linear equations. However, there is not always a solution to these equations, and it is limited to the use of triangular fuzzy numbers.

A method proposed by Buckley [17] uses trapezoidal fuzzy numbers, which are more useful as a general form of triangular fuzzy numbers. The method uses a geometric mean technique to normalize pairwise matrices and create priority vectors, similar to some methods in crisp AHP. With relatively reasonable results, and a well established technique, this method was selected for use in this effort. The generalized procedure is outlined as follows:

1. Each pairwise comparison matrix, $\tilde{P}_k$, $k = 1, 2, ...m + 1$, is constructed by decision makers, where each element is a fuzzy trapezoidal number $\tilde{p}_{ij} = (a_{ij}, b_{ij}, c_{ij}, d_{ij})$.

2. For each matrix, priority (or weighting) vectors are determined by normalizing row geometric means by their sum, where

$$\tilde{z}_i = \left(\prod_{j=1}^{n} \tilde{p}_{ij} \right)^{1/n}, \quad \text{for } i = 1, 2, ...n \tag{43}$$

   and the priority vector can be calculated as

$$\tilde{v}_i = \tilde{z}_{ij} \times \left(\sum_{i=1}^{n} \tilde{z}_i \right)^{-1}, \quad \text{for } i = 1, 2, ...n. \tag{44}$$

3. Fuzzy weights, $\tilde{w}_k$, and priority vectors are aggregated into a final fuzzy priority

vector for the alternatives with respect to the overall objective.

$$\tilde{s}_i = \sum_{k=1}^{m} \tilde{w}_k \tilde{v}_{ik}, \quad \text{for } i = 1, 2, ...n \tag{45}$$

Fuzzy TOPSIS

In a review of proposed fuzzy implementations of the TOPSIS method, all but a few methods produce a crisp relative closeness [54, 122, 128]. Again, this runs contrary to not only the need to assess the relative robustness of alternative rankings, but also the intuitive idea that fuzzy weights and fuzzy scorings should lead to a fuzzy value for relative closeness. A fairly comprehensive list of fuzzy TOPSIS approaches can be found on page 167 of [54]. Simple approaches, like Triantaphyllou and Lin's proposal produce fuzzy relative closeness results by essentially replacing crisp arithmetic in the TOPSIS process with fuzzy counterparts directly. The resulting fuzzy relative closeness membership functions are often skewed and non-sensical, with values well above 1.0.

Wang and Elhag present another methodology for performing fuzzy TOPSIS, utilizing α-cuts [128]. The authors argue that Triantaphyllou and Lin's method provides distorted and exaggerated relative closeness values that stem from not considering the effects of fuzzy arithmetic on the result. Their method solves a non-linear programming model (with linear constraints) to determine intervals for relative closeness at each α-cut in an attempt to keep the relative closeness values on a scale closely related to the normalized decision matrix and weights. Given their superior results, Wang and Elhag's α-cut approach to fuzzy TOPSIS was selected for use in this research, and is detailed as follows:

1. The fuzzy decision matrix is constructed by providing fuzzy scores, $\tilde{x}_{ij}$, for each alternative with respect to each criterion.

2. The decision matrix is normalized by the the maximum element of all a criterion's fuzzy scores, rather than the sum of squares, making the normalized scores:

$$\tilde{r}_{ij} = \frac{\tilde{x}_{ij}}{\tilde{f}_j^*}; \quad \tilde{f}_j^* = max_i(max_y\{\tilde{x}_{ij}(y)\}) \tag{46}$$

3. With the decision matrix normalization resulting in fuzzy values on the range $[0, 1]$, the positive and negative ideal for each criterion are considered to be $B^* = \{1, ..., 1\}$ and $B^- = \{0, ..., 0\}$, respectively

4. Relative closeness α-cuts, $[(\widetilde{RC}_i)^L_\alpha, (\widetilde{RC}_i)^U_\alpha]$, are calculated given the normalized α-cuts, $(\tilde{r}_{ij})_\alpha = [(\tilde{r}_{ij})^L_\alpha, (\tilde{r}_{ij})^U_\alpha]$, and criterion weight α-cuts, $(\tilde{w}_j)_\alpha = [(\tilde{w}_j)^L_\alpha, (\tilde{w}_j)^U_\alpha]$. The interval at given α-cuts can be determined by solving the following non-linear objectives with linear constraints:

$$(\widetilde{RC}_i)^L_\alpha = \min \frac{\sqrt{\sum_{j=1}^{m}(w_j(\tilde{r}_{ij})^L_\alpha)^2}}{\sqrt{\sum_{j=1}^{m}(w_j(\tilde{r}_{ij})^L_\alpha)^2} + \sqrt{\sum_{j=1}^{m}(w_j((\tilde{r}_{ij})^L_\alpha - 1))^2}} \tag{47}$$

given $(\tilde{w}_j)^L_\alpha \leq w_j \leq (\tilde{w}_j)^U_\alpha, \ j = 1, ..., m$

$$(\widetilde{RC}_i)^U_\alpha = \max \frac{\sqrt{\sum_{j=1}^{m}(w_j(\tilde{r}_{ij})^U_\alpha)^2}}{\sqrt{\sum_{j=1}^{m}(w_j(\tilde{r}_{ij})^U_\alpha)^2} + \sqrt{\sum_{j=1}^{m}(w_j((\tilde{r}_{ij})^U_\alpha - 1))^2}} \tag{48}$$

given $(\tilde{w}_j)^L_\alpha \leq w_j \leq (\tilde{w}_j)^U_\alpha, \ j = 1, ..., m$

5. Once the relative closeness α-cuts have been determined by solving the non-linear programming models above, the relative closeness for each alternative

can be generated by combining its α-cuts:

$$\widetilde{RC}_i = \bigcup_{\alpha} [(\widetilde{RC}_i)^L_\alpha, (\widetilde{RC}_i)^U_\alpha], \quad 0 \leq \alpha \leq 1 \tag{49}$$

2.4 Computational Tools

In order to perform the research necessary for this effort, and manifest the methods described computationally, a custom module was created using Python. Python is a flexible, open source programming language that continues to grow in popularity in the scientific and engineering community. This section is meant to give a brief overview of the computational resources used to complete this research. The core of the fuzzy set mathematics and fuzzy logic in the module was handled by a SciPy toolkit under development, Fuzzy Logic SciKit [130]. This tool provides a basic toolkit of independently developed fuzzy logic algorithms.

The module includes a series of classes for each fuzzy system that incorporate all of the fuzzy system functionality, as well the any necessary training algorithms. The fuzzy rule-based systems are built based on plain text script files (.fcl) that mostly follow the standard form of Fuzzy Control Language as defined by IEC 61131-7 [27]. Several examples of these files can be found in Appendix C. The Fuzzy Rule Based System (FRBS) class can read and write these files to allow for saving/loading of a given system. Once the system is instantiated and the .fcl file is loaded, the run function can be called with a dictionary of inputs and will return a dictionary of the outputs. Similar the Discrete Fuzzy Expert System (DFES) system is instantiated with an input granularity, output granularity, and number of hidden nodes. The trained system weight matrices can be written to a comma-separated value file (.nwf) by command and then read again to allow for trained systems to be recreated as required.

A series of training and optimization functions were also developed as part of the research and are included in the module for training and optimizing the FRBSs. These systems have dependancies in SciPy and NumPy, as they can use several of the SciPy optimization methods. A custom genetic algorithm is also included for FRBS optimization. The backpropagation training algorithm for the DFES system is contained in its system class and is called with training parameters and given a training data set. A NEFPROX training algorithm is provided as well, despite the system not being used in the final framework.

In order to combine the systems into a single framework, OpenMDAO, an open source multi-disciplinary platform based in Python for systems analysis, was utlized. The module includes OpenMDAO components used to read and translate the expert data used, as well as OpenMDAO component versions of the implemented fuzzy systems. Several example assemblies were developed from these components to perform the optimization and decision making portions of the approach.

In addition to these elements, the MADM methods used in the benchmarking approach were developed in Python and are included in the module, as well a number of examples for training and validating systems, as well as post-processing of results. While it is not prudent or feasible to include the entirety of the module code in an Appendix, a working copy of the module can be found in the Github repository (https://github.com/pattersoniv/FFAS.git) for FFAS (Fuzzy Framework for Architecture Selection).

CHAPTER III

RESEARCH

In this chapter, the specific research and experiments completed in pursuit of meeting the outlined research objectives and answering the proposed research questions is discussed. The seminal goal of the research effort is to demonstrate the development of a framework to facilitate the identification of a set of ideal architectures or families with respect to the outlined example problem. Before discussion is began on the framework itself, the example problem for the design of an advanced experimental high-speed Vertical Takeoff and Landing (VTOL) aircraft is outlined. A benchmark is performed using probabilistic TOPSIS and AHP methods, to illustrate the application of a method that can handle some uncertainty and considers a number of alternatives. As part of the benchmark, the same methods are also performed using fuzzy set theory approaches, to better understand how fuzzy set theory can measure uncertainty in concept selection.

Once the problem has been setup and the benchmark performed, the Sections 3.2 and 3.3 introduce the fuzzy systems as a model for analyzing concept alternatives with respect to the various system attributes of interest. Fuzzy system development and expert data elicitation are closely linked in this research, and much of the research for these two efforts was done simultaneously, however the elicitation portion of the research is presented first. Several types of fuzzy systems and methods for their development are explored. These systems are proposed to be used similarly to how response surface methods are utilized in many modern conceptual design tools [64], able to quickly model an expert or first-order tool evaluation of a concept, but while capturing the uncertainty therein. After comparing these fuzzy models to the benchmark scores, the models are integrated into a single framework, driven by a selected architecture from the morphological matrix. In Section 3.4, the models are combined into a single framework, and the potential for MCDM is introduced. The framework is wrapped in an optimization algorithm to try and identify the best potential architecture alternatives for future design activities. This is hoped to be the primary contribution of this research, to provide an approach to identify potentially suitable architectures for new system designs, without being stuck using the same alternatives based on designer bias and industrial momentum.

3.1 Benchmarking: Application of MADM

In order to develop the proposed approach for concept down-selection, a simplified applicable concept selection problem is outlined in this section. Next, a benchmark is developed by applying a probabilistic version of two traditional MADM methods (TOPSIS and AHP) to illustrate how current methods might be used to reduce the problem to a manageable number of concepts while accounting for uncertainty. Finally, MADM methods based on fuzzy set theory are applied to the problem to understand how fuzzy sets can be used as an alternative means to represent uncertainty and explore their similarities and differences from the probabilistic techniques. These MADM methods are proven methods for the type of multi-attribute decision making problem that concept down-selection represents, and are used later for comparison of the proposed approach.

3.1.1 Setting up the Problem

The Defense Advanced Research Project Agency (DARPA) 2013 solicitation for a VTOL Experimental Aircraft (X-Plane) was issued with the intent to "advance the design and development of new and improved technologies, sub-systems, aircraft concepts and configurations to demonstrate vertical lift aircraft with fundamentally enhanced performance capabilities" [30]. The solicitation is focused on developing a

10,000 - 12,000 lbs X-Plane demonstrator aircraft with radical system improvements in cruise speed, hover efficiency, cruise efficiency, useful load fraction, and payload fraction. Specifically, the five main performance objectives desired by DARPA are as follows:

- Capable of sustained flight speeds at 300 to 400 knots.
- A system Figure of Merit (FM) of at least 0.75.
- A system cruise lift-to-drag ratio of at least 10.
- A vehicle useful load fraction of at least 40%.

Proposers are encouraged to submit conceptual designs with unique technical solutions and explore non-traditional VTOL configurations. The open-ended nature of the system requirements, the motivation to consider a new range of innovative system configurations, and the uncertainty about the performance of new configurations all make this problem an ideal application of the methods being developed here.

Quality Function Deployment (QFD), a widely accepted, systematic means of translating customer requirements into technical requirements, is used to develop a simple list of criteria and determine their respective weights. Over the last 15-20 years, a number of different QFD approaches have been proposed utilizing fuzzy concepts in various ways [13, 23, 35, 55, 99]. There are a number of in-depth and complex methods, including those to account for multiple evaluators and varying confidence in each, as well as taking into account the correlation matrix and other effects. As the focus of this research is concept selection methods, and not defining the problem through QFD, a relatively simple framework for fuzzy QFD is utilized, similar to the approach described by Bevilacqua [13]. Fuzzy arithmetic is used to sum the product of each characteristic's impact relationships and requirement fuzzy weights. For m customer requirements (WHATs), and n engineering characteristics (HOWs), the importance weights can be calculated as:

$$w_{(HOW)j} = \sum_{i=1}^{m} r_{ij} \otimes w_{(WHAT)i} \quad \text{for } j = 1, 2, ..., n \tag{50}$$

The resulting fuzzy importance weightings are multiplied by the scaler risk weighting, and then normalized by the maximum, $max(x)$ where $w_{(HOW)j} = \mu(x)$ for $j \in [1, n]$. This procedure was employed for several types types of fuzzy numbers that might be required for various decision making methods, and can be repeated if others are needed.

To establish a need for the system, a set of eighteen basic customer requirements were derived from the Broad Agency Announcement (BAA), and are listed in Table 4. These requirements represent the desired capabilities required of the system. A prioritization matrix was utilized to develop customer importance weights for each of the requirements. Fuzzy versions of these weights were also created to account for the possibility of indecision or variability in the customer's priorities between requirements. Given a vector of scores for a given requirement, $\{s\}$, in the prioritization matrix, triangular fuzzy importance weights were defined for each requirement as $A_T(a_1, a_2, a_3) = (min[s], \bar{s}, max[s])$. Trapezoidal fuzzy importance weights were defined as $A_I(a_1, a_2, a_3, a4) = (min[s], max(\bar{s} - \sigma_s, min[s]), min(\bar{s} + \sigma_s, max[s]), max[s])$. Both crisp and fuzzy weights are listed in Table 4.

In order to evaluate possible concept configurations and future design iterations, the customer requirements were then translated into engineering characteristics that could represent a system. The engineering characteristics represent the means by which to meet the capabilities defined by the customer requirements. An impact relationship matrix is then created defining the relationships between the customer requirements and the engineering characteristics as shown in Figure 26. A non-linear scale is utilized as defined in Table 5. By multiplying the impact of characteristics impact relationship values with the cooresponding importance weights and summing them, a technical importance rating is calculated. Each of these is then multiplied by a

Table 4: Customer Requirements (WHATs)

Requirement		Crisp Weight	Fuzzy Triangular Weight	Fuzzy Trapezoidal Weights
Basic Performance	High Speed Flight	7.00	(5.00, 7.00, 9.00)	(5.00, 5.47, 8.53, 9.00)
	Hover Efficiency	6.78	(3.00, 6.78, 9.00)	(3.00, 5.11, 8.44, 9.00)
	Cruise Efficiency (L/D)	6.44	(3.00, 6.44, 9.00)	(3.00, 4.94, 7.95, 9.00)
	Useful Load Fraction	6.22	(3.00, 6.22, 9.00)	(3.00, 4.04, 8.41, 9.00)
	Payload Fraction	6.00	(3.00, 6.00, 9.00)	(3.00, 3.70, 8.30, 9.00)
Vehicle Attributes	Controllability	5.89	(3.00, 5.89, 7.00)	(3.00, 4.48, 7.00, 7.00)
	System Simplicity	3.11	(1.00, 3.11, 7.00)	(1.00, 1.51, 4.72, 7.00)
	System Efficiency	3.78	(1.00, 3.78, 7.00)	(1.00, 1.94, 5.61, 7.00)
	Vibratory Response	4.33	(3.00, 4.33, 9.00)	(3.00, 3.00, 6.50, 9.00)
	Acoustic Signature	2.78	(1.00, 2.78, 7.00)	(1.00, 1.00, 4.82, 7.00)
	Innovative System	4.33	(3.00, 4.33, 7.00)	(3.00, 3.00, 6.15, 7.00)
System Viability	Transportable (Ship/Air)	2.89	(1.00, 2.89, 7.00)	(1.00, 1.28, 4.49, 7.00)
	Reliability	3.67	(1.00, 3.67, 7.00)	(1.00, 1.39, 5.94, 7.00)
	Safety	6.67	(5.00, 6.67, 9.00)	(5.00, 5.49, 7.90, 9.00)
Program Development	Technology Maturity	4.89	(3.00, 4.89, 7.00)	(3.00, 3.00, 6.89, 7.00)
	Technology Scalability	4.33	(3.00, 4.33, 7.00)	(3.00, 3.00, 6.01, 7.00)
	Airworthiness Likelihood	4.44	(3.00, 4.44, 7.00)	(3.00, 3.00, 6.36, 7.00)
	Program Cost	6.44	(3.00, 6.44, 9.00)	(3.00, 4.79, 8.10, 9.00)

risk weighting value associated with the risk of developing that particular engineering characteristic. The result is a risk weighted importance weight for each engineering characteristic. These will serve as criteria weightings for MCDM methods to select concept configurations.

Table 5: Impact Relationship Values

Relationship	Crisp	Fuzzy Triangular	Fuzzy Trapezoidal
None	0	$(0, 0, 3)$	$(0, 0, 1, 3)$
Weak	1	$(0, 1, 3)$	$(0, 1, 3, 5)$
Moderate	3	$(1, 3, 5)$	$(1, 3, 5, 7)$
Strong	9	$(3, 9, 9)$	$(5, 7, 9, 9)$

The fuzzy QFD approach allowed for several varieties of fuzzy weights to be generated for the engineering characteristics in addition to traditional crisp weights. The resulting system engineering characteristics used as criteria, and their fuzzy triangular and trapezoidal weights are shown in Table 6. The triangular fuzzy weights were also translated directly into Probability Density Function (PDF)s, with the fuzzy number $W_{TRI} = (a, b, c)$ translated as $W \sim TRI(a, b, c)$. To create uniform distribution, the minimum and maximum from the triangular (the same min and max as the trapezoidal weights) were used to define the range, $W \sim U(a, c)$. Examples of all

					Column #	1	2	3	4	5
					Direction of Improvement	▼	▲	▲	▲	▲
					Categories	Vehicle Performance				
Row #	Relative Weight	Customer Importance	Maximum Relationship	Categories	Customer Requirements (Explicit and Implicit) / Functional Requirements	Empty Weight Fraction	Max Cruise Speed	Vertical Lift Efficiency	Aerodynamic Efficiency	Propulsive Efficiency
1	8%	126	9	Basic Performance	High Speed Flight	▽	●	○	●	●
2	8%	122	9		Hover Efficiency	○	○	●		▽
3	7%	116	9		Cruise Efficiency (High L/D)	○	○	▽	●	●
4	7%	112	9		Increased Useful Load Fraction	●	●	○	▽	▽
5	7%	108	3		Increased Payload Fraction	●	○	▽	▽	▽
6	7%	106	1	Vehicle Attributes	Controlability	○	▽		▽	▽
7	3%	56	3		System Simplicity/Elegance	○		▽	○	
8	4%	68	9		System Efficiency	○	○	●	●	●
9	5%	78	3		Reasonable Vibratory Response	○	▽	▽		○
10	3%	50	3		Reasonable Acoustic Signature		▽		○	▽
11	5%	78	9		Innovative System	○	●	○	▽	○
12	3%	52	1	System Viability	Transportable (Ship/Air)	○			▽	▽
13	4%	66	1		Reliability	▽	▽			
14	7%	120	3		Safety	●	○	▽	○	
15	5%	88	9	Program Development	Mature Technology	●	●	●	○	○
16	5%	78	9		Scalable Technology (4k - 24k lbs)	●	●	●	○	○
17	5%	80	1		Airworthiness Likelyhood		▽			
18	7%	116	9		Cost	●	○	●	○	○

Figure 26: QFD Relationship Matrix

four weighting types used to represent uncertainty are shown in Figure 27. All of the weights were normalized across the same scale for consistency in comparing MADM methods.

3.1.2 Generating Benchmark Alternatives

Once the QFD process was completed and value had been established, a morphological matrix was constructed to generate alternatives. The morphological matrix was based largely on functional system aspects (vertical lift function, forward propulsion function), but has some physical elements. Vertical lift without forward airspeed is

Table 6: Engineering Characteristics (Criteria): Weights

Characteristic		Crisp	Triangular Fuzzy	Trapezoidal Fuzzy
Empty Weight Fraction	(%)	0.51	(0.09, 0.51, 1.00)	(0.09, 0.26, 0.77, 1.00)
Max Cruise Speed	(knots)	0.44	(0.08, 0.44, 0.90)	(0.08, 0.21, 0.68, 0.90)
Vertical Lift Efficiency	(FM)	0.26	(0.04, 0.26, 0.60)	(0.04, 0.13, 0.43, 0.60)
Aerodynamic Efficiency	(L/D)	0.22	(0.04, 0.22, 0.56)	(0.04, 0.10, 0.38, 0.56)
Propulsive Efficiency	(η)	0.22	(0.04, 0.22, 0.55)	(0.04, 0.10, 0.37, 0.55)

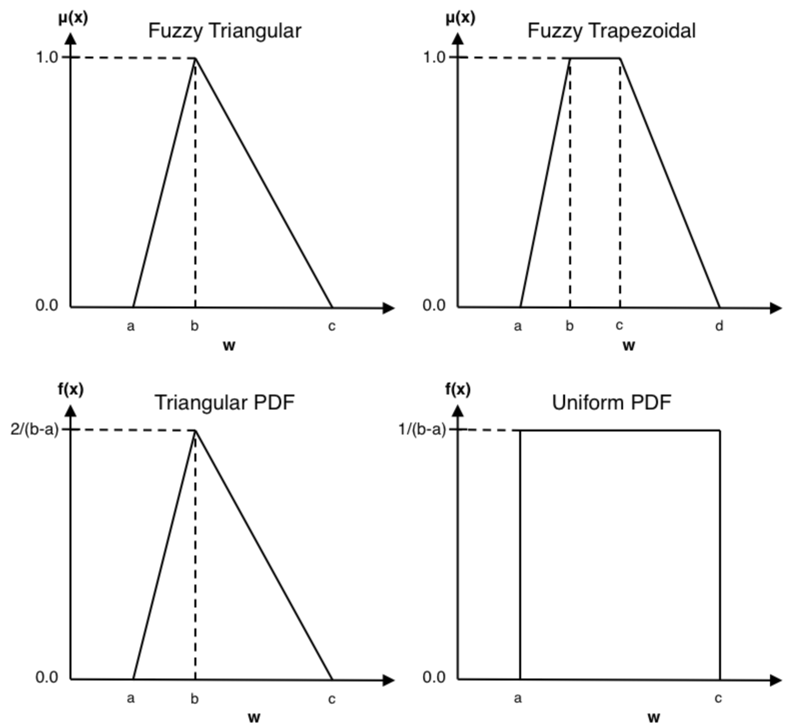


Figure 27: Weighting Methods Representing Uncertainty

treated as a separate function, while horizontal flight is assumed to accomplished through a physical lifting wing and a physical forward propulsion method as a pure. Tilting lift systems are assumed to be used for both vertical and horizontal propulsion. For the purpose of this research, the combination of a single selected option for each aspect of the system constitutes an concept configuration alternative.

The morphological matrix was deliberately sized to create a feasible number of alternatives to allow for a full factorial search of the design space within reasonable time constraints. This will create a "truth" data set for comparison with results of

Table 7: Morphological Matrix

System Aspects		Options					
Vertical Lift	Vertical Lift System	Single Main	Traverse Ext. Sys.	Tandem Ext. Sys.	Propulsor(s) In-Wing	Propulsor(s) In-Body	Tailsitter
	Vertical Lift Propulsor	Propellor(s)	Rotor(s)	Ducted Fan(s)	Direct Thrust		
	Vertical Lift Drive Sys.	Shaft Driven	Reaction Drive	Tip Blown	Direct Thrust		
	Vertical Lift Technology	None	Var. RPM	Stopped Rotor	Variable Dia.	Gyrodyne	
Fwd Propulsion	Fwd. Propulsor	Propellor(s)	Rotor(s)	Ducted Fan(s)	Direct Thrust		
	Fwd. Drive System	Shaft Driven	Reaction Drive	Direct Thrust			
	Fwd. System Type	Fixed	Tilting VL Sys.	Clutched/ Disabled	Tilting Non-VL Sys.		
Wing	Primary Wing Type	Conven-tional	Delta Wing	Flying Wing	Blended Body	Tilting Wing	Stopped Rotor
Engine	Engine Type	Turbo Shaft/ Prop	Turbojet	Turbofan	Hybrid Electric		

the proposed MODM search methods, as well as evaluation of the fuzzy system.

The morphological matrix identifies $552,960$ individual alternatives. A compatibility matrix was also constructed (not shown due to its size) identifying 74 of the individual 670 combinations as incompatible within the morphological matrix, and reducing the design space required to explore.

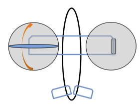
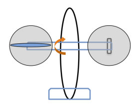
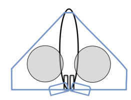
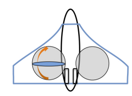
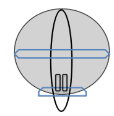
With a complete morphological matrix, the IPPD process can continue by generating feasible alternatives from the matrix. For the purpose of baselining the design process and providing alternatives for the early phases of research, ten alternatives were identified for evaluation, as listed in Tables 8 and 9. Concept alternatives were selected in an attempt to create configurations that were likely to be feasible for the desired DARPA VTOL X-Plane mission and requirements, as well as explore as many of the available options outlined in the morphological matrix as possible. The selected alternatives generally fall into one of two categories: more conventionally explored high speed VTOL concepts (tilt-rotor, tilt-wing) that may be able to be evolved to meet far reaching requirements through redesign and technology infusion; and unconventional concepts that may provide the base for a more advanced system, but may not be feasible without unproven technologies or advances in design.

Alternatives concepts explored include for several next generation tilt-rotors [3, 12, 33, 90, 98, 115] and a tilt-wing [84] concept. Other concepts were considered that had been demonstrated or proposed but never gone into production, such as a tilting rotor-in-wing [3], fan-in-wing turbofan [45], and a stopped rotor concept [90]. The Tip-Driven Gyrodyne is based on the high speed Heliplane concept developed in the early 2000's by Groen Brothers Aviation and the Georgia Institute of Technology as funded by DARPA [87, 131]. Also included as an alternative is a more conventional propeller driven gyrodyne, similar to the Slowed-Rotor/Compound™ demonstrated on a small scale, with proposed scaling, by Carter Aviation Technologies [20, 42]. Additionaly, two alternatives were added to approximate the two concepts that had been publicly identified at the time this research began as being selected for the first phase of DARPA's VTOL X-Plane program. These are Sikorsky and Lockheed Martin's Rotor Blown Wing concept [61], and Boeing's Phantom Swift [46]. A number of sources were used for inspiration in alternative generation, notably including the 1991 studies sponsored by NASA Ames to identify high speed rotorcraft concepts and the technologies that might be required to expand rotorcraft capabilities up to 450 knots [12, 33, 90, 98]. The alternatives inspired by these reports and other various sources also attempt to include some of the options identified by the morphological matrix.

3.1.3 Evaluating the Alternatives

With the problem defined, the alternative concept configurations were evaluated with respect to their suitability to each criterion. To tie the process to the desired goals, quantitative measures were utilized for this exercise wherever possible. For each system criterion, subject matter experts were asked to provide the minimum and maximum possible system level outcome for each synthesized alternative. This was accomplished as part of the survey created in Section 3.2, using the same interval elicitation method. Only empty weight ratio was evaluated qualitatively, on $[9 - 1]$

Table 8: Alternatives 1-5

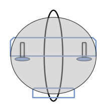
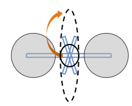
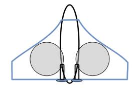
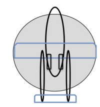
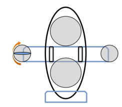
		Alternative 1	Alternative 2	Alternative 3	Alternative 4	Alternative 5
	System Aspects	Var. RPM Tilt-Rotor	Var. RPM. Tilt-Wing	Fan-in-Wing Turbojet	Tilting Fan-in-Wing	Stopped Reaction Rotor
Vertical Lift	VL System	Traverse Ext. Sys.	Traverse Ext. Sys.	Propulsor(s) In-Wing	Propulsor(s) In-Wing	Single Main
	VL Propulsor	Rotor(s)	Propellor(s)	Ducted Fan(s)	Ducted Fan(s)	Rotor(s)
	VL Drive	Shaft Driven	Shaft Driven	Tip Blown	Shaft Driven	Reaction Drive
	VL Technology	Var. RPM	Var. RPM	None	Var. RPM	Stopped Rotor
Fwd Propulsion	Fwd. Propulsor	Rotor(s)	Propellor(s)	Direct Thrust	Rotor(s)	Direct Thrust
	Fwd. Drive System	Shaft Driven	Shaft Driven	Direct Thrust	Shaft Driven	Direct Thrust
	Fwd. System Type	Tilting VL Sys.	Tilting VL Sys.	Fixed	Tilting VL Sys.	Fixed
Wing	Primary Wing Type	Conventional	Tilting Wing	Delta Wing	Blended Body	Stopped Rotor
Engine	Engine Type	Turbo Shaft/ Prop	Turbo Shaft/ Prop	Turbojet	Turbo Shaft/ Prop	Turbofan
						

(Best to Worst) scale, due to the difficulty in predicting this attribute at such an early stage. If a participant felt that they did not have the expertise to provide a meaningful score, then no evaluation was recorded for that particular alternative and criterion. In this way uncertainty due to both vagueness of the early stage problem and disagreement between decision makers was considered to be captured.

Probabilistic versions of both TOPSIS and AHP, as described in Section 2.3.2, were employed to evaluate the 10 alternatives. This could be considered to be a current method to account for uncertainty in concept selection when considering a large ($\geq> 10$) alternative architectures this early in the design process.

To address the validity of using fuzzy sets to model uncertainty in concept selection, fuzzy versions of the same TOPSIS and AHP methods will be applied to the problem as well. The results of these fuzzy versions, procedurally defined in Section 2.3.3, are compared to the results probabilistic methods to understand how various types of fuzzy sets perform in representing uncertainty. While no attempts are made to compare how fuzzy sets and probabilistic means model the same uncertainty mathematically (this is an ongoing debate more suited to a thesis in theoretical

Table 9: Alternatives 6-10

		Alternative 6	Alternative 7	Alternative 8	Alternative 9	Alternative 10
	System Aspects	Auto-Gyro	Twin Rotor Tail-Sitter	Fan-in-Wing Fixed Pusher	Heliplane	Fan-in-Body Tilt-Duct
Vertical Lift	VL System	Single Main	Tailsitter	Propulsor(s) In-Wing	Single Main	Propulsor(s) In-Body
	VL Propulsor	Rotor(s)	Rotor(s)	Ducted Fan(s)	Rotor(s)	Ducted Fan(s)
	VL Drive	Reaction Drive	Shaft Driven	Shaft Driven	Reaction Drive	Shaft Driven
	VL Technology	Gyrodyne	Var. RPM	None	Gyrodyne	None
Fwd Propulsion	Fwd. Propulsor	Propellor(s)	Rotor(s)	Propellor(s)	Direct Thrust	Ducted Fan(s)
	Fwd. Drive System	Shaft Driven	Shaft Driven	Shaft Driven	Direct Thrust	Shaft Driven
	Fwd. System Type	Fixed	Fixed	Fixed	Fixed	Tilting Non-VL Sys
Wing	Primary Wing Type	Conventional	Conventional	Blended Body	Conventional	Conventional
Engine	Engine Type	Turbo Shaft/ Prop	Turbo Shaft/ Prop	Turbo Shaft/ Prop	Turbofan	Hybrid Electric
						

mathematics), the MADM problems are setup to try and use and compare similarly structured weights and input evaluations.

In order to create singular input evaluation scores to exercise the TOPSIS methods, evaluation data in the form of ranges for each criterion was transformed into uniform and triangular PDFs as well as triangular and trapezoidal fuzzy numbers. Uniform PDFs were created by averaging both the minimum provided scores and the maximum provided scores, $X \sim U(\overline{[min(x_i)]}, \overline{[max(x_i)]})$. Triangular fuzzy scores were created by adding an overall average score to the ranges, $\tilde{r}_{i\,TRI} = ((\overline{[min(x_i)]}, \overline{(x_i)}, \overline{[max(x_i)]})$. The triangular PDFs were then developed from the triangular fuzzy scores using the same method applied to develop the weight PDFs. Trapezoidal fuzzy scores were defined as $\tilde{r}_{i\,TRAP} = ((\overline{[min(x_i)]}, \overline{[min(x_i)]}, \overline{[max(x_i)]}\ \overline{[max(x_i)]})$, similarly to uniform PDFs. Figure 28 shows the resulting fuzzy triangular and trapezoidal scores, and could also represent normalized triangular and uniform PDFs.

To maintain a reasonable number of required evaluations (600 pairwise comparisons would have been required to populate all the AHP matrices), and keep consistency among MADM methods, AHP pairwise comparison matrices were created

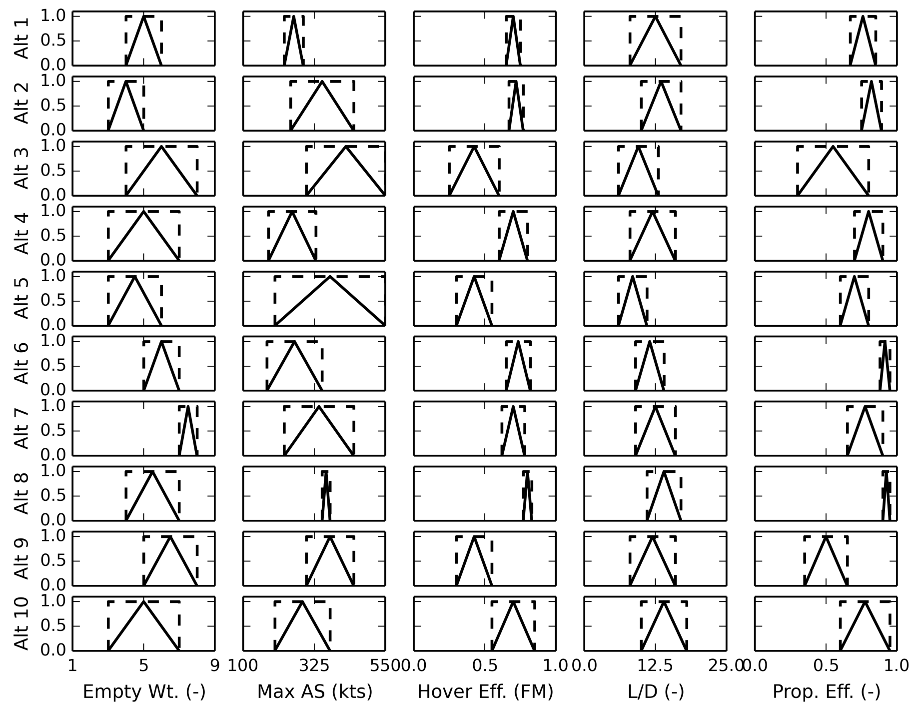


Figure 28: Combined Fuzzy Triangular and Trapezoidal Scores as Generated from Expert Elicitation

using the same elicited data. For each criterion, matrices were generated using ratios of each of the alternatives' scores, where each element $\tilde{a}_{ij}$ was calculated as

$$\tilde{a}_{ij} = \frac{\tilde{x}_i}{\tilde{x}_j},\ i = 1, ..., n;\ j = 1, ..., n. \tag{51}$$

Values to populate each alternative pairwise comparison matrix for each input data type were each generated in this manner using the scores of the same type from the TOPSIS method. The same fuzzy weights were used for both fuzzy AHP and fuzzy TOPSIS methods, and the same weight PDFs were used for both probabilistic MADM approaches. While a pairwise approach to determining criteria preferences is usually part of the AHP process, using constant weights was thought to aide in comparing the MADM methods on equal terms.

In the Monte Carlo Simulation, steps 4-6 from Figure 25 were repeated 10,000 times with new crisp inputs generated from the weight and score PDFs at each iteration to generate statistically significant results. The resulting relative closeness or overall priority values, as well as the resulting weightings for each iteration were recorded. This takes approximately 4-8 minutes on a standard personal computer for each method, while a solution to the exercise for each of the fuzzy methods takes only a few seconds.

3.1.4 Making a Decision

Once results had been generated for all four methods, some post-processing was necessary to analyze the results. Normal distributions were fitted to the relative closeness and overall priority results for each alternative by calculating maximum likelihood estimators for mean and standard deviation parameters. Then those parameters were converted into normal PDFs and cumulative distribution functions (CDF) to model output probabilities.

To help analyze the results of the fuzzy methods, resulting relative closeness and overall priority fuzzy membership functions for each alternative were defuzzified using a centroid technique. The centroid is one of the most prevalent and physically obvious means of defuzzification, defining a single crisp centroid value, x^*, as

$$x^* = \frac{\int \mu_{\tilde{A}}(x)\dot{x}\, dx}{\int \mu_{\tilde{A}}(x)dx}. \tag{52}$$

In comparing fuzzy methods to probabilistic methods, no attempt was made to directly translate between the resulting "possibilistic" fuzzy membership functions, and probabilistic PDFs or CDFs. Disagreement on the direct correlation of these concepts still exists [104], and it was thought that recommendations in this area would be best left to more practiced mathematicians. Instead, the methods are compared for their sensitivity to inputs, resulting ability to convey any uncertainty about the results, and then finally their consistency of performance.

Tables 10 and 11 shows the resulting probabilistic means and fuzzy centroids for each of the four methods and each input type explored, along with the alternative ranks in each of those methods as determined by those crisp measures. In addition to the fuzzy rankings resulting from the fuzzy centroids, These tables also indicates the ranks as taken from the mean value of an α-cut at the maximum membership function value of the final priority or relative closeness in gray parenthesis. Results here can be indicative of consistency across each of the MADM methods, as well as sensitivity of each method to the type of inputs.

Table 10: Ranking Results for AHP Methods with Crisp Measure ($\alpha = \overline{\max(\mu)}$ Rank)

#	Alternative	Prob. AHP (Uniform)	Prob. AHP (Triangular)	Fuzzy AHP (Triangular)	Fuzzy AHP (Trapezoidal)
1	Variable RPM Tilt-Rotor	7 ($\mu = .27$)	9 ($\mu = .26$)	9 (9) ($c = .61$)	8 (8) ($c = .73$)
2	Variable RPM Tilt-Wing	6 ($\mu = .28$)	7 ($\mu = .28$)	8 (7) ($c = .62$)	9 (9) ($c = .73$)
3	Fan-in-Wing Turbojet	4 ($\mu = .29$)	3 ($\mu = .35$)	2 (2) ($c = .84$)	2 (2) ($c = 1.00$)
4	Tilting Fan-in-Wing	9 ($\mu = .23$)	8 ($\mu = .27$)	7 (8) ($c = .68$)	7 (7) ($c = .81$)
5	Stopped Rotor Turbofan	10 ($\mu = .20$)	10 ($\mu = .23$)	10 (10) ($c = .55$)	10 (10) ($c = .66$)
6	Auto-Gyro	3 ($\mu = .35$)	4 ($\mu = .32$)	6 (5) ($c = .72$)	6 (6) ($c = .85$)
7	Twin Rotor Tail-Sitter	1 ($\mu = .44$)	2 ($\mu = .36$)	1 (1) ($c = .86$)	1 (1) ($c = 1.02$)
8	Fan-in-Wing Fixed Pusher	2 ($\mu = .37$)	1 ($\mu = .37$)	3 (3) ($c = .84$)	4 (4) ($c = 99.$)
9	Heliplane	5 ($\mu = .29$)	5 ($\mu = .32$)	4 (4) ($c = .83$)	3 (3) ($c = .99$)
10	Fan-in-Body Tilt-Duct	8 ($\mu = .25$)	6 ($\mu = .32$)	5 (6) ($c = .79$)	5 (5) ($c = .94$)

Each of the four methods (probabilistic AHP, fuzzy AHP, probabilistic TOPSIS, and fuzzy TOPSIS) show some sensitivity to varying the form of uncertainty modeling in the input scores. Namely there are some small differences in the mean and crisp measures for each alternative (which are relative values), and their resulting

Table 11: Ranking Results for TOPSIS Methods with Crisp Measure ($\alpha = \overline{\max(\mu)}$ Rank)

#	Alternative	Prob. TOPSIS (Uniform)	Prob. TOPSIS (Triangular)	Fuzzy TOPSIS (Triangular)	Fuzzy TOPSIS (Trapezoidal)
1	Variable RPM Tilt-Rotor	5 ($\mu = .39$)	8 ($\mu = .44$)	6 (8) ($c = .69$)	4 (5) ($c = .68$)
2	Variable RPM Tilt-Wing	6 ($\mu = .35$)	9 ($\mu = .39$)	9 (9) ($c = .65$)	9 (9) ($c = .64$)
3	Fan-in-Wing Turbojet	9 ($\mu = .25$)	6 ($\mu = .47$)	7 (5) ($c = .68$)	7 (6) ($c = .66$)
4	Tilting Fan-in-Wing	7 ($\mu = .29$)	7 ($\mu = .45$)	8 (7) ($c = .67$)	8 (8) ($c = .66$)
5	Stopped Rotor	10 ($\mu = .13$)	10 ($\mu = .25$)	10 (10) ($c = .63$)	10 (10) ($c = .62$)
6	Auto-Gyro	2 ($\mu = .57$)	3 ($\mu = .60$)	3 (3) ($c = .75$)	2 (2) ($c = .74$)
7	Twin Rotor Tail-Sitter	1 ($\mu = .78$)	1 ($\mu = .74$)	1 (1) ($c = .78$)	1 (1) ($c = .76$)
8	Fan-in-Wing Fixed Pusher	3 ($\mu = .49$)	2 ($\mu = .61$)	2 (2) ($c = .75$)	3 (3) ($c = .74$)
9	Heliplane	4 ($\mu = .40$)	4 ($\mu = .52$)	5 (4) ($c = .69$)	5 (4) ($c = .68$)
10	Fan-in-Body Tilt-Duct	8 ($\mu = .28$)	5 ($\mu = .49$)	4 (6) ($c = .69$)	6 (7) ($c = .68$)

ranks. The TOPSIS methods are considerably more consistent than the AHP methods in terms of the methods themselves, but not necessarily the input types. The fuzzy methods for both approaches appear to be more consistent with respect to the different input types than the probabilistic versions. Each of the TOPSIS methods seem to agree on a smaller group of best alternatives, namely the Auto-Gyro, the Fan-in-Wing with Fixed Pushers, and the Twin Rotor Tail-Sitter, and the worst alternative, the Stopped Rotor. The probabilistic TOPSIS methods also seem to show a more clear differentiation between the top and bottom concept candidates, while the other method's priority rankings are more closely grouped.

Using a fuzzy centroid to rank the alternatives allows for capturing of the spread and skew of the fuzzy membership function that the centroid represents, while the nature of the probabilistic means do not account for this as the outputs are modeled as normal. The ranks based on the α-cuts appear to more closely match the probabilistic results in general, though they mostly match the centroid rankings. These ranks are largely based on the results with the highest "possibility" of resolving, and not the skew of the membership function below them. The fuzzy TOPSIS results particularly reflect this case. They show significantly lower ranks for alternatives like the Fan-in-Wing Turbojet, where scoring was indecisive and indicated larger ranges for the scoring criteria, and higher ranks for the alternatives like the Tilting Fan-in-Wing, which received more consistent scores. This ability to capture the skew of the results in a single value is a noted advantage to using the fuzzy methods.

In order to better understand the uncertainty captured by the MADM methods' results, several means of visualizing their results are explored next. The remaining analysis was completed using the triangular probabilistic and fuzzy inputs, as these inputs seem to provide the most consistent and differentiable results, and limited uncertainty to reasonable levels in all cases.

To compare the inherent uncertainty and relative ranking of fuzzy results for both methods, α-cuts at $\alpha = 0.9$ were plotted along with the fuzzy centroid for each alternative. Similarly, the fitted PDFs from probabilistic results were used to create 90% confidence intervals for each alternative, which were plotted along with their mean relative closeness values. It should be noted that the 90% and 0.9 selected for this have no direct numerical correlation to each other, but instead are commonly used values for these techniques. These are illustrated in Figures 29 and 30, comparing the results of the probabilistic methods to the fuzzy methods for the top 4 alternatives for AHP and TOPSIS each, as defined by each alternative's probabilistic mean.

It is immediately evident that the fuzzy AHP results show centroid values well outside the α-cuts for each alternative. This indicates a heavy skew of less possible values in the membership functions to higher priorities, while lower priorities are much more possible. All of the possible/probable AHP priority ranges also appear on smaller and more even ranges than the relative closeness ranges from TOPSIS. Both of these phenomena are seemingly a function of the mathematical methods themselves.

The fuzzy methods again offer a means to see the skew of the underlying membership function, while the resulting confidence intervals are inherently symmetrical from being modeled as normal. For example, in comparing the fuzzy TOPSIS results for alternative alternative 8 (Fan-in-Wing Fixed Pusher) to alternative 9 (Heliplane), alternative 8 shows a higher relative closeness centroid, with an α-cut mostly centered around the centroid. Conversely, alternative 9 shows a centroid that indicates a skew outside and lower than the 0.9 α-cut. This indicates the two alternatives were thought to both be good choices, and that mostly likely alternative 8 is the better choice, but there is some possibilities (albeit lower) that 9 could perform better. Similarly a smaller confidence interval for alternative 6 can be seen, indicating less uncertainty in the experts' evaluations.

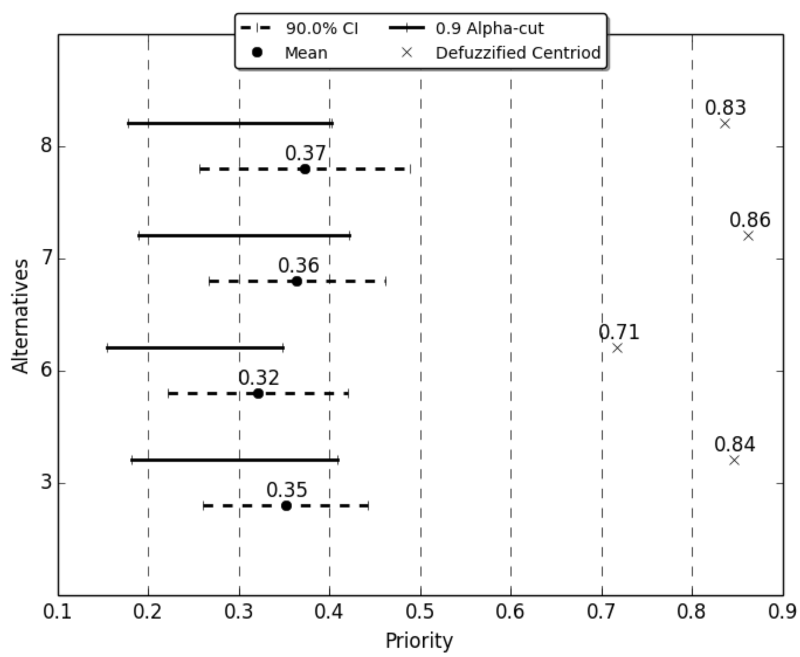


Figure 29: Comparison of Uncertainy in AHP Results (Triangular Inputs)

So far the top alternatives have been ranked by a single measure the mean (or

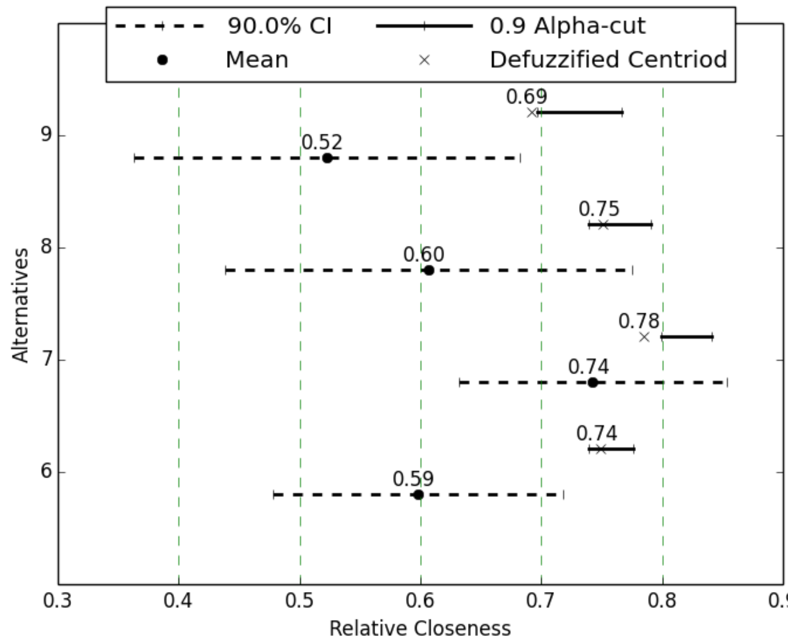


Figure 30: Comparison of Uncertainy in TOPSIS Results (Triangular Inputs)

centroid) priority or relative closeness measure. To help identify several solutions to carry forward to a more detailed phase of concept selection, decision makers might generate plots shown in Figures 31 and 32, similar to those suggested by [59]. Here the probability of an alternative being the top N designs is plotted as a function of N. Intersecting lines illustrate varying concentrations of uncertainty in the resulting priorities or relative closenesses among alternatives. Concept alternatives that experts rate more consistently or with smaller possible ranges of performance across criteria are more likely to have narrow priority or relative closeness distributions, and appear to be more vertical functions in this visualization.

Probabilisitic AHP results indicate three alternatives, alternatives 8, 7 and 3, that separate themselves as have much higher probabilities of ranking in the top 4-5 alternatives. Another grouping of three are very closely grouped in the middle with moderate probabilities of ranking highly, while the other 4 alternatives have almost no chance of being ranked highly. Probabilistic TOPSIS results show alternative 7 (Twin Rotor Tail-Sitter) as a clear overall best alternative, with a probability of 0.8

of being ranked as the best alternative, and a probability of close to 1.0 of being ranked in the top 4. Beyond that, alternatives 6 and 8 differentiate themselves as the next best alternatives. The more horizontal nature of alternative 3 indicates a less robust alternative, and at higher levels of certainty it is only guaranteed to be in the top 5-6 alternatives instead of ranked as third.

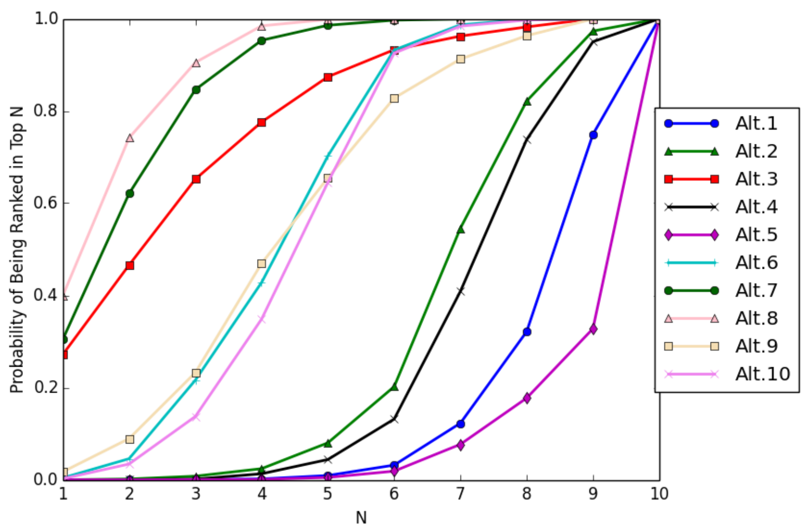


Figure 31: AHP Results (Triangular Inputs): Probability of Ranking in the Top N Alternatives

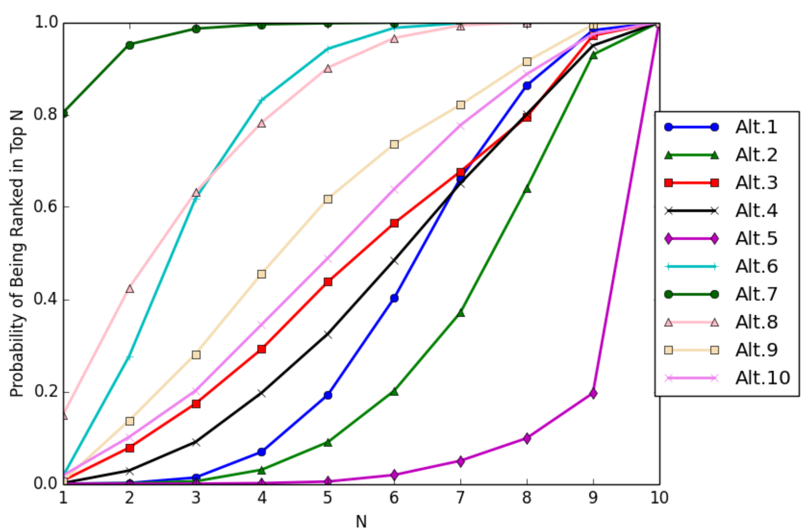


Figure 32: TOPSIS Results (Triangular Inputs): Probability of Ranking in the Top N Alternatives

For the results of fuzzy set based MADM methods, the same approach of identifying probabilities as a function of subset size is not applicable. In order to help identify a family of top solutions, Baas and Kwackernaak's dominance measure [6] is calculated for each alternative with respect to every other alternative. These values can be roughly interpreted as the degree of possibility that one alternative outranks another. This "possibility matrix" is augmented by the minimum of each alternative's dominance values, indicating the degree of possibility that the alternative will be ranked the highest, as well as the average alternative dominance. The matrix has also been shaded to more quickly visually indicate the most dominate alternatives.

Figure 33 shows the possibility matrix for the fuzzy AHP results. The dominance values are all closely grouped, with the lowest overall value being 0.94, indicating a relatively high degree of uncertainty in ranking in the fuzzy results. Alternative 7 is shown to have the highest possibility of ranking as the top alternative, but alternatives 3 and 8 are very close, with dominance values of 0.99. Some scaling may be necessary to further differentiate alternatives using this technique. The possibility matrix for the TOPSIS results is shown in Figure 34. The fuzzy TOPSIS possibility matrix shows alternative 7 to have the best possibility of ranking as the top alternative as well, with alternative 8 being the next closest at a value of 0.88, only being dominated by alternative 7. Here, relative dominance of alternatives are more easily discerned due to the wider range of dominance values.

Consistency Analysis

In addition to analysis of the results above, the methods were briefly analyzed with respect to their ability to produce consistent, and thus more meaningful, results. Rank reversal, a phenomenon where a MADM method's resulting ranks are changed by the deletion (or addition) of an alternative, is a common cause for concern and discussion when deciding on decision making approaches [129]. The legitimacy of AHP

Alternative	1	2	3	4	5	6	7	8	9	10	Min.	Avg.
1	-	0.99	0.95	0.99	1.0	0.97	0.95	0.96	0.97	0.98	**0.95**	**0.98**
2	1.0	-	0.96	1.0	1.0	0.98	0.96	0.96	0.97	0.99	**0.96**	**0.98**
3	1.0	1.0	-	1.0	1.0	1.0	0.99	1.0	1.0	1.0	**0.99**	**0.99**
4	1.0	0.99	0.96	-	1.0	0.98	0.96	0.96	0.97	0.98	**0.96**	**0.98**
5	0.99	0.98	0.95	0.99	-	0.97	0.94	0.95	0.96	0.97	**0.94**	**0.97**
6	1.0	1.0	0.98	1.0	1.0	-	0.98	0.98	0.99	1.0	**0.98**	**0.99**
7	1.0	1.0	1.0	1.0	1.0	1.0	-	1.0	1.0	1.0	**1.0**	**1.0**
8	1.0	1.0	0.99	1.0	1.0	1.0	0.99	-	1.0	1.0	**0.99**	**0.99**
9	1.0	1.0	0.98	1.0	1.0	1.0	0.98	0.99	-	1.0	**0.98**	**0.99**
10	1.0	1.0	0.97	1.0	1.0	0.99	0.97	0.98	0.99	-	**0.97**	**0.99**

Figure 33: AHP Ranking Possibility Matrix (Triangular Inputs)

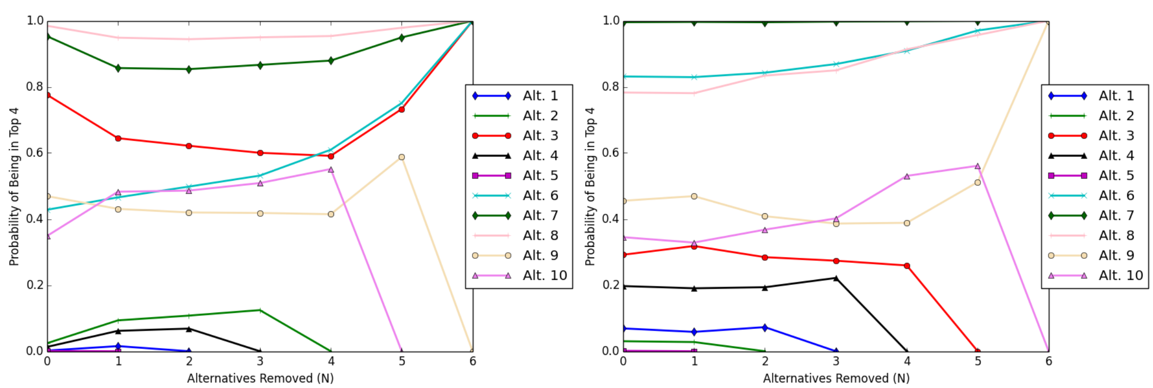
has been criticized and doubted over its possibility of producing rank reversals. In order to illustrate each of the discussed methods' tendencies to produce rank reversal, each method was repeated several times on the same set of alternatives, using the same scores and weights, but removing an alternative each time. For AHP methods, this meant removing the appropriate rows and columns of each pairwise comparison matrix.

For the two probabilistic methods, the alternative with the worst mean relative closeness from the original probabilistic TOPSIS iteration was removed, and the method was repeated on the remaining subset. At each iteration, the probability of being in the top 4 was recorded for each alternative, with the results illustrated in Figure 35. In this figure, any time two lines cross indicates a rank reversal. The probabilities of each alternative should grow steadily until they drop to 0.0 as they are removed, or rise to 1.0 when only 4 alternatives remain. The AHP sensitivity results showed a number of rank reversals, particularly among the closely grouped middle alternatives (6,9, and 10). The TOPSIS method had a few cases of rank reversal,

Alternative	1	2	3	4	5	6	7	8	9	10	Min.	Avg.
1	-	1.0	0.94	0.98	1.0	0.79	0.64	0.82	0.91	0.94	**0.64**	**0.90**
2	0.94	-	0.91	0.94	1.0	0.78	0.66	0.82	0.89	0.91	**0.66**	**0.88**
3	1.0	1.0	-	1.0	1.0	0.92	0.78	0.92	0.98	1.0	**0.78**	**0.96**
4	1.0	1.0	0.95	-	1.0	0.83	0.69	0.86	0.93	0.96	**0.69**	**0.92**
5	0.88	0.95	0.87	0.89	-	0.72	0.61	0.77	0.85	0.86	**0.61**	**0.84**
6	1.0	1.0	1.0	1.0	1.0	-	0.84	0.98	1.0	1.0	**0.84**	**0.98**
7	1.0	1.0	1.0	1.0	1.0	1.0	-	1.0	1.0	1.0	**1.0**	**1.0**
8	1.0	1.0	1.0	1.0	1.0	1.0	0.88	-	1.0	1.0	**0.88**	**0.98**
9	1.0	1.0	1.0	1.0	1.0	0.95	0.81	0.94	-	1.0	**0.81**	**0.97**
10	1.0	1.0	0.99	1.0	1.0	0.91	0.76	0.91	0.97	-	**0.76**	**0.95**

Figure 34: TOPSIS Ranking Possibility Matrix (Triangular Inputs)

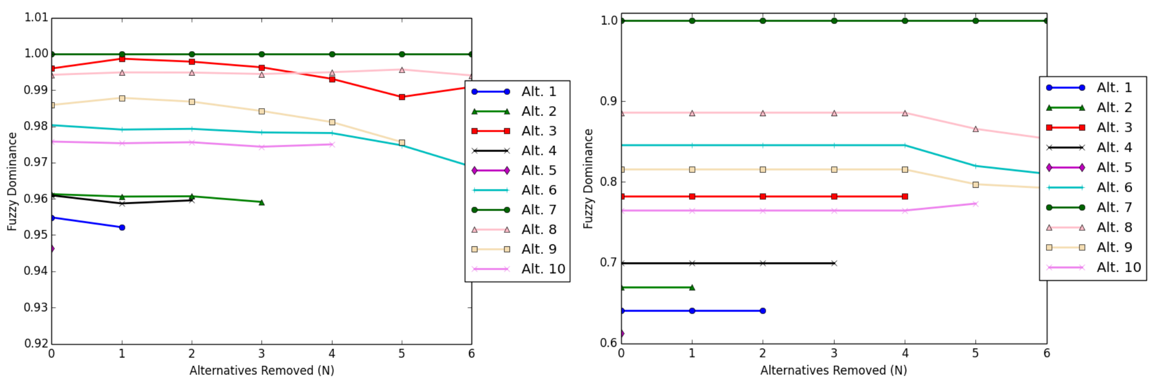
most notably between alternatives 9 and 10, and some undesired negatively sloped probability trends, but for the most part the probabilities remained consistently separated.



(a) (b)

Figure 35: Probability of Alternative Being in the Top 4 with N Alternatives Removed for Probabilistic AHP(a) and TOPSIS(b)

In order to analyze the occurrence of rank reversal in the fuzzy methods, a similar method was used. The minimum dominance values from the previously constructed ranking possibility matrix were tracked through iterations of removing the worst alternative (with respect to probabilistic relative closeness). The results of this exercise for both fuzzy AHP and TOPSIS are shown in Figure 36. Fuzzy AHP shows only one real instance of rank reversal, though it is between 2 of the top 3 alternatives. Fuzzy TOPSIS performs the best in this analysis, with no rank reversals shown, and remarkably consistent scores as alternatives are removed. This consistency is likely due to the best and worst scores for the attributes belonging to alternatives that are not eliminated until the fourth alternative is removed (Alternative 3).



(a) (b)

Figure 36: Possibility of Ranking as Top Alternative with N Alternatives Removed for Fuzzy AHP(a) and TOPSIS(b)

3.1.5 Benchmarking Discussion

The benchmark probabilistic MADM methods all seem to indicate that alternatives 7 and 8, the Twin Rotor Tail-Sitter and the Fan-in-Wing with Fixed Pushers, provide the two best alternatives, with potentially alternative 6, the Autogyro, or even alternative 3, the Fan-in-Wing Turbojet being worth a look as well. In a conventional design process, these designs might be carried forward to be modeled in higher fidelity.

Beyond the results of the benchmark, this activity was also meant to answer Research Question 1, which essentially asked if fuzzy sets could provide an interpretable measure of the uncertainty inherent in system performance during concept selection. The results of this research do not provide any indication that fuzzy set theory could not be used as an uncertainty measure during concept selection, giving no reason to reject the given hypothesis that this approach would be viable.

Each of the four explored MADM methods, probabilistic AHP and TOPIS and fuzzy AHP and TOPSIS, shows evidence of providing a viable means for identifying the best concept architectures with respect to multiple criteria. Likewise, each method provides the means to capture and measure the uncertainty inherent to expert evaluation during the earliest phases of concept selection. However the uncertainty indicated in the results is interpretable more as the likelihood or possibility of achieving some score or rank, and is not directly relatable to an alternatives robustness with respect to a given attribute (this is inherent in the scores). When careful interpretation, results of the four methods are relatively consistent in terms of rank and corresponding distributions of final scores.

Each of the two general method types provides distinct advantages and disadvantages. The probabilistic approaches provide a more instinctive and easily translatable result in the form of probability functions, such as the probability of a concept being ranked in the top N alternatives. However, creating probabilistic input functions from expert evaluations can be a bit arbitrary outside of uniform functions indicating any possible value over some range. The example used here with 10 alternatives and 5 criteria takes a few minutes of computation to achieve statistically relevant results with Monte Carlo simulation, however a large number of alternatives and/or criteria could require more nuanced or computationally powerful approaches.

Alternatively, the fuzzy methods provide several advantages and disadvantages of their own. Fuzzy membership function inputs can be be more intuitive, with a "possibility" based approach, and may even be able to be based on linguistic analysis. However the fuzzy methods also provide results as membership functions, which can be less intuitive than probabilities to compare and interpret across various scenarios. Defuzzification techniques, such as the centroid method used here, can help to quickly translate skew and spread in the results, as well as be directly used for rankings. In contrast to Monte Carlo simulation, the fuzzy methods quickly provide a result in just seconds on a standard desktop computer, even using rather simple search algorithms to solve the non-linear programming problem presented by fuzzy TOPSIS. The fuzzy methods have the potential to provide an excellent additional check for more readily accepted probabilistic methods, and with wider understanding could stand on their own as decision making methods.

With a basis established for representing uncertainty in concept selection with fuzzy sets, the primary elements of the research could be continued. In the next two sections, the research done to generate fuzzy systems is described. These systems work as models of expert opinion (and first-order design data), providing a computational, online means of generating evaluations similar to those completed for this benchmarking effort.

3.2 Data Elicitation

In order to develop the fuzzy systems responsible for evaluation of architectures, several means of gathering relevant data were necessary. This section discusses the development of means to elicit and reduce data for the training and operation of the models developed as a part of this research. Developing the fuzzy systems meant a simultaneous development of the mathematical and computational systems themselves, as well as the data needed to drive and train them. In this report, the process of gathering data will be covered first, then the development of fuzzy systems. In reality, the two efforts are closely linked.

There is a great deal of literature outlining methods and techniques for eliciting expert opinion for various means of fuzzy decision making, modeling, or other activities [14]. Several factors were considered in developing the approach outlined below. Among the most important was the time it would take a SME or small team of experts to provide the necessary data to create the framework. This was one reason real participation was not pursued, but a framework laid for future research and true participation. If several hundred evaluations are required to train a system model, the evaluations should be able to be made quickly and accurately. The elicitation should also be flexible enough to not require SMEs to have a strong background in fuzzy set theory, only knowledge relevant to the system(s) under development. With these reasons in mind, the methods considered for this research are among the most simple techniques for elicitation.

A number of proposed elicitation methods are based on various interpretations on the source(s) of fuzziness as it relates to the problem. This is primarily related to the interpretation of what "grade of membership" actually means, how it is measured, and how it can be meaningfully used [14, 22, 93]. While the rigorous mathematical and theoretical discussion behind fuzzy set theory and how it relates to other fields is outside the scope of this project, it is safe to say that membership can be roughly interpreted in this project with a combination of two semantics. The first being a random-set viewpoint, where some percentage of the population on the defined universe falls in the interval (essentially an α-cut) corresponding to the grade of membership for it's descriptive set (e.g. $\mu_A(x) = 0.5$ corresponds to x falling in the interval containing 50% of the population with the possibility of being A). The second, a likelihood viewpoint, describes $\mu_A(x) = 0.5$ as a 50% likelihood that x is A given errors in measurement, incomplete information, expert disagreement, and any other sources of fuzziness.

Considering these interpretations, among the proposed means for eliciting data to create fuzzy sets are:

- *Polling* - asks the expert (yes/no) if different alternatives belong to various previously identified fuzzy sets.

- *Direct Rating* - asks the expert which previously identified fuzzy set an alternative belongs to.

- *Reverse Rating* - a membership function is given, and the expert is asked to select corresponding alternatives.

- *Interval* - the expert identifies an interval in the universe the alternative should fall in.

- *Membership Exemplification* - the expert is directly asked to define the membership function.

- *Pairwise* - a linguistic based method, the expert is asked to compare pairs of alternatives with respect to a fuzzy linguistic term.

- *Fuzzy Clustering* - a mathematical technique to construct membership functions from a set of existing data.

The elicitation method selected for this research was an interval method. For the majority of the expert elicitations, the SMEs are asked to provide an interval or range of values that bound the alternative with respect to the characteristic in question. The interval can then be translated into a fuzzy membership function at the discretion of the designers and decision makers. This technique was selected for a number of reasons, including it's flexibility with respect to creating fuzzy functions and the lack of background in fuzzy set theory required, and the ease of translation into a domain the SME could understand. Moreover with this method, the expert or experts was not limited to a discrete pre-selected set of linguistic terms, evaluations could be easily translated and understood among teams, and a large number of evaluations could be generated relatively quickly. Other means of elicitation were not explored in depth due to the lack of participation in the initial surveys. Future research may be necessary in this area. The specific process of elicitation is discussed next.

Some standardized means to collect the information required from experts was required. Following the conjectures made about eliciting data, a simple paper survey was created to gather data for creating the architecturally dependent inputs. The survey can found in Appendix A. The survey also includes a preliminary section to setup the example problem and collect the evaluations for pre-selected alternatives used in Section 3.1. While a simple survey eventually proved sufficient to collect this data, other more complicated means could be used during this phase. Future research might involve understanding more complicated and efficient means to capture the necessary data during elicitation. The following sections discuss the specific nature of the data collected and how it was used to form inputs to the fuzzy system models used.

3.2.1 Gathering Architecture Input Data

In order to facilitate evaluation of a given architecture from the morphological matrix, the architecture would have to be first be translated into some set of appropriate engineering characteristics. These characteristics would then in turn be used to determine system performance with respect to the design criteria or metrics identified (in this case, empty weight ratio, hover figure of merit, etc.).

Thus, the first step in gathering input data for the potential architectures was to identify any design parameters affecting the identified criteria that would be subject to architectural choices foreach functional aspect of the concept. These inputs would have to vary some between identified options in the morphological matrix. For example, a selection of vertical lift propulsor architecture (rotor, ducted fan, etc.) would greatly affect or constrain the system's disk loading.

The survey asked to consider whether each of these identified inputs could be evaluated quantitatively or qualitatively, and if applicable, what metric or units should be used to evaluate the input. The effects of architecture selection on some inputs, such as empty weight ratio and flat plate drag area, proved to be difficult to estimate quantitatively. Here a familiar SME could say roughly which architecture choices might prove better or worse for a given input, but quantitative ranges proved too dependent on other inputs, the final sizing, and the conceptual design of the system.

Once the inputs were identified, the applicable units and potential ranges of those inputs, specific to the given system, were also identified. Here, it was critical to again identify the specific units and values that a given expert could understand and relate to. For some cases it might be necessary to elicit data in both metric and imperial units and translate answers before implementing the framework. In the cases where quantitative measures were not possible, a qualitative scale of 1(worst) to 9 (best) was utilized. For these qualitative evaluations it was found to be important to identify a baseline by which to judge the various architecture choices, to try and allow experts to consider these inputs from the same vantage.

Experts were asked to evaluate a possible range of each input characteristic when each option was selected to perform its respective system functional aspect. Here each option was considered the only defined functional aspect of the system, and then each input was evaluated with respect to that particular architecture choice. For example, an expert might indicate that if the forward propulsor was a ducted fan, any disk loading would be possible in the range of 70 lbs/ft^2 to 150 lbs/ft^2. It was noted that some options may have had no appreciable impact on a given input, or that input may not have been applicable to the option. In these cases, the full range of possible input options was recorded.

In addition to expert evaluation, some of the input data for architectural options was available from historical sources. Because of the nature of the example VTOL X-Plane problem used, the number of applicable systems was limited, as was the amount of applicable data. Only aircraft with applicable subsystems, operating in the desired mission conditions, were considered. As this approach is generally only relevant to new system developments, where architecture definition is difficult and critical, using historical data is a difficult proposition. Careful attention was given to apply data only where it's historical counterpart had correspondingly similar architecture and operating conditions for the given input and option. A list of compiled historical data is shown in Appendix B. Several means were considered to combine historical and expert based data for a given option, including interval math and weighting, throughout the course of the research. The inputs considered can play a significant role in the results of the fuzzy models created in the next section, particularly the simpler fuzzy rule-based systems. In the end no one approach to combining data seemed to make sense for all the inputs, so the data was combined on a case by case basis. The input ranges used in the final system training and framework are listed in Table 12.

3.2.2 Gathering Architecture Performance Data

In addition to input data, it was necessary to gather some data to train the more complicated fuzzy rule-based systems and neuro-fuzzy systems as discussed further in Section 3.3. Several means to collect the appropriate data were explored, including Design of Experiments. Even though the design space is categorical, the options in the morph matrix represent continuous design variables. With the nature of the way the combinations represent this continuous space and the spacing of incompatible alternatives within the compatibility matrix, creating a matrix of experiments that was orthogonal and compatible proved more difficult than was worth the time to find if the solution existed. Instead, a randomized sample was used to attempt to collect

Table 12: Architectural Dependent (Morphological) Input Data

Vertical Lift System

Input	Range	Units	Single Main	Transverse Sys.	Tandem Sys.	Prop(s) In-Wing	Prop(s) In-Body	Tailsitter
Empty Weight (ϕ)	1 - 9	Qual	[6-8]	[3-6]	[4-8]	[3-7]	[5-8]	[6-8]
Eq. Drag Area (f)	1 - 9	Qual	[2-5]	[2-7]	[3-6]	[5-8]	[6-8]	[6-9]
Disk Loading (w)	1 - 150	lb/sqft	[1-15]	[15-80]	[1-25]	[50-120]	[60-150]	[55-100]
Download Fac. (e_d)	0.0 - 0.3	%GWT	[0.05-0.15]	[0.05-0.3	[0.08-0.3]	[0-0.05]	[0-0.05]	[0.02-0.1]
Thrust/Power (TP)	0.1 - 20.0	lb/hp	[2.0-13.0]	[1.5-12.0]	[3.0-14.0]	[0.8-2.0]	[0.8-2.0]	[2.0-6.0]

Vertical Lift Propulsor(s)

Input	Range	Units	Propellor	Rotor	Ducted Fan(s)	Direct Thrust
Empty Weight (ϕ)	1 - 9	Qual	[3-6]	[5-8]	[3-7]	[2-7]
Disk Loading (w)	1 - 150	lb/sqft	[28-100]	[5-22]	[60-115]	[150-150]
Solidity (σ)	0.0 - 0.4	-	[0.15-0.35]	[0.05-0.17]	[0.2-0.4]	[0.01-0.01]

Vertical Lift Drive

Input	Range	Units	Shaft Driven	Reaction Drive	Tip Blown	Direct Thrust
Empty Weight (ϕ)	1 - 9	Qual	[4-7]	[6-9]	[5-9]	[7-9]
Drive Sys. Eff. (η_d)	0.5 - 1.0	%	[0.8-0.98]	[0.5-0.7]	[0.5-0.8]	[0.9-1.0]

Vertical Lift Technology

Input	Range	Units	None	Var. RPM	Stopped Rotor	Var. Dia. Rotor	Gyrodyne
Empty Weight (ϕ)	1 - 9	Qual	[1-9]	[1-8]	[1-6]	[1-5]	[1-8]
Eq. Drag Area (f)	1 - 9	Qual	[1-9]	[1-9]	[1-6]	[1-8]	[4-8]
Disk Loading (w)	1 - 150	lb/sqft	[1-150]	[1-150]	[1-150]	[1-20]	[1-15]
Cruise L/D (LD)	5 - 25	-	[5-25]	[5-25]	[3-8]	[5-25]	[8-14]

Forward Propulsor(s)

Input	Range	Units	Propellor	Rotor	Ducted Fan(s)	Direct Thrust
Empty Weight (ϕ)	1 - 9	Qual	[5-8]	[4-7]	[2-6]	[5-8]
Prop Efficiency (η_p)	0.6-1.0	%	[0.8-0.98]	[0.6-0.8]	[0.6-0.9]	[0.5-0.7]

Forward Drive System

Input	Range	Units	Shaft Driven	Reaction Drive	Direct Thrust
Empty Weight (ϕ)	1 - 9	Qual	[4-7]	[6-9]	[7-9]
Drive Sys. Eff. (η_d)	0.5 - 1.0	%	[0.85-0.96]	[0.5-0.75]	[0.9-1.0]

Forward System Type

Input	Range	Units	Fixed	Tilting VL Sys.	Clutched/ Disabled	Tiliting Non-VL
Empty Weight (ϕ)	1 - 9	Qual	[1-9]	[1-7]	[2-9]	[1-7]
Prop Efficiency (η_p)	0.6-1.0	%	[0.6-1.0]	[0.6-0.8]	[0.6-1.0]	[0.75-0.95]
Eq. Drag Area (f)	1 - 9	Qual	[1-9]	[4-8]	[1-9]	[1-8]
Thrust/Power (TP)	0.1 - 20.0	lb/hp	[1.5-15]	[3.0-6.0]	[0.1-20]	[0.1-20]

Wing System Type

Input	Range	Units	Conventional	Delta Wing	Flying Wing	Blended Wing/-Body	Tilting Wing	Stopped Rotor
Empty Weight (ϕ)	1 - 9	Qual	[4-7]	[6-8]	[6-9]	[4-8]	[3-6]	[2-7]
Cruise L/D (LD)	5 - 25	-	[8-20]	[5-10]	[7-15]	[9-22]	[6-15]	[4-10]
Wing Loading (WS)	15-300	lb/sqft	[4-170]	[11-141]	[11-65]	[5-58]	[50-100]	[125-250]
Eq. Drag Area (f)	1 - 9	Qual	[3-7]	[6-9]	[6-9]	[5-9]	[3-7]	[2-6]

Propulsion System Type

Input	Range	Units	Turboshaft	Turbofan	Turbojet	Hybrid-Electric
Empty Weight (ϕ)	1 - 9	Qual	[6-8]	[5-7]	[4-6]	[3-7]
Fuel Eff. (SFC)	1 - 9	Qual	[4-8]	[2-5]	[1-4]	[6-9]

data evenly across design space of compatible architecture options.

In order to facilitate data collection, a simple web-based evaluation tool was developed and used, as illustrated in Figure 37. Usable in any browser, the user is presented with a morphological matrix for the given problem, and a choice of which criteria to score alternatives for. A javascript application then generates a random alternatives from the morphological matrix until a compatible one is found. The user is asked to evaluate the alternative and provide a range (min,max) in which they think an optimized design for the given architecture will perform, similarly to the benchmark exercise. An option is also available for the expert to indicate that they think the alternative has incompatible options that had not been previously accounted for. These can be captured from the provided data and added to the compatibility matrix manually if relevant. The user can also choose to produce a new random alternative if they feel they cannot evaluate the one given. Once data has been collected, it can be submitted at any time via an automatically generated email.

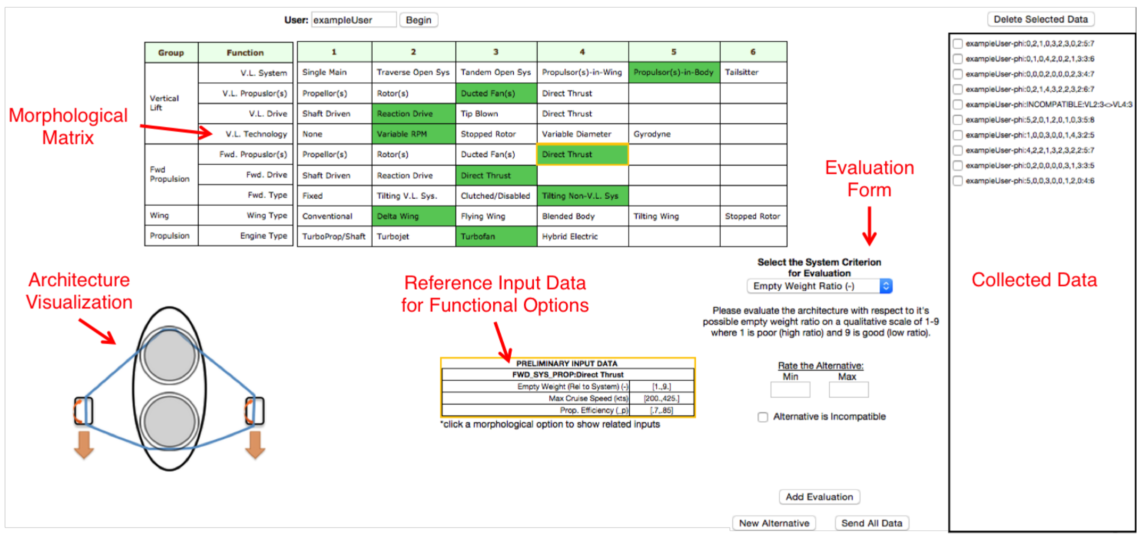


Figure 37: Online Concept Evaluation Tool

The possibility of using a more interactive means to collect the data, like Delphi method [29, 50], is also worth consideration. In Delphi method approaches for gathering expert opinion, a moderated panel of experts provides their opinions iteratively with sanitized, anonymous evaluations or opinions being fed back to the panel. Ideally this process leads to convergence of accurate expert evaluations. A number of variations to this general procedure exist, including means to perform web-based evaluations, to include expert systems, and means to combine evaluations using fuzzy sets.

Because this portion of the research only lays a framework for elicitation, without setting up a lengthy participation phase to recruit and schedule experts, the Delphi approach was not actively utilized. However some future research may consider further exploring the elicitation phase and implementing this approach. To keep the timeframe for this research to a reasonable span, the models based on expert evaluation were completed using only data collected from the author of this research. The data was reviewed several times to try and achieve better consistency in a Delphi-like exercise. However, it is conjectured that using multiple SMEs to generate data in a semi-structured format for both training and architecture inputs would not only save time, but provide more accurate results. Because it is important to understand how the evaluations of various experts can be combined to produce a single input or training data set, Section 3.2.3 explains the capabilities of fuzzy methods to accommodate this process.

3.2.3 Data Aggregation

Once data had been gathered from experts to use and train the systems, some means of reducing and treating that data to make it appropriate for system training was necessary. Aggregation of expert opinion has become a comprehensive field unto itself, with means ranging from the very simple to entire MCDM methods based on aggregation of expert evaluation of alternatives [50, 38]. These methods range from the simplest operations, to Ordered Weighting Averaging (OWA) operators [134] (a flexible category of weighted summations of the ordered data), or even complex methods like the Choquet or Sugeno integrals [69].

To demonstrate the basic flexibility of fuzzy data elicitation techniques, some simple options for aggregating the expert evaluations are explored. To demonstrate disagreement between experts, two sets of data representing "pseduo-random experts" were generated based on the author's evaluations of Empty Weight, representing additional experts that did not complete the survey or provide evaluations. The mean of each original evaluation's range was adjusted based on a normally distributed random bias, and a new min and max for the range were created around that bias based again on normal distributed random values. The distribution parameters were adjusted to represent a generally conservative but less certain pseduo-random expert and a more optimistic and certain pseduo-random expert as illustrated for a select number of evaluations in Figure 38. The figure also illustrates the averages (minimum and maximum), intersection, and union of these ranges. Respectively these options represent approaches for aggregation representing a middle ground, less-conservative, and more-conservative approaches with respect to capturing the uncertainty expressed.

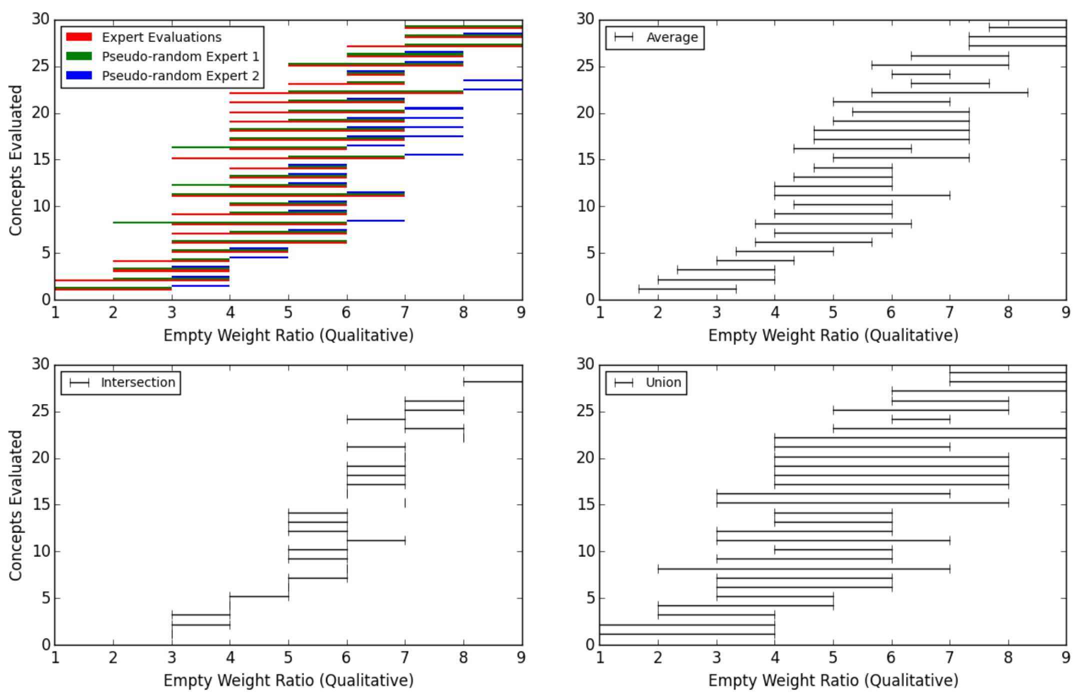


Figure 38: Selected Examples of Expert Evaluation (Empty Weight Ratio) with Pseudo-Random Experts and Data Combination Examples

Once the interval data has been aggregated, it can be translated into fuzzy membership functions to be inputs into the fuzzy systems, or to train fuzzy systems, as is discussed in more detail in Sections 3.3.1 and 3.3.6. Options explored for representing the expert evaluations as fuzzy numbers include gaussian, triangular, and trapezoidal membership functions. These common and simple fuzzy membership functions allow for flexibility in representing the intervals identified as fuzzy membership functions. This fuzzy data then will serve as inputs to the system models developed (representing uncertainty in system design parameters based on a selected architecture), and training data for developing those systems (representing uncertainty in the system performance for a given architecture). The effects of using various membership functions for data are explored more in Section 3.3.5.

Figure 39 illustrates an example of the fuzzification of the combined expert evaluations for average, intersection, and union. Here the gaussian function has the same mean as the aggregate interval and a standard deviation of 25% of the interval width, the triangular function is defined as $[a_1, a_2, a_3] = [min(I), mean(I), max(I)]$, and the trapezoidal function is defined as $[a_1, a_2, a_3, a_4] = [min(I), min(I), max(I), max(I)]$. These definitions are used throughout the research to create membership functions from the input data and training data. In general, the fuzzy rule-based system training with fuzzy data discussed in Section 3.3.5 proved more successful the narrower and closer to a singleton value the membership functions are, while the neuro-fuzzy system training in Section 3.3.7 worked better if input and output membership were not too narrow but not too wide (gaussian functions proved best).

Aggregating the intervals before fuzzification creates simple smooth membership function that is computationally easier to handle than what could result if aggregating membership functions. Moreover, these simpler smooth functions are more clearly interpretable, and provide a better data set for training and utilizing fuzzy systems.

Another potential aggregation method includes weighting SME evaluations based on experience or their assuredness in each evaluations (after providing an interval and some level of certainty in that answer). Other methods might seek to derive a skew from the more complex means to aggregate fuzzy sets into a single smooth function. A detailed exploration of these methods and their effects might be undertaken, but

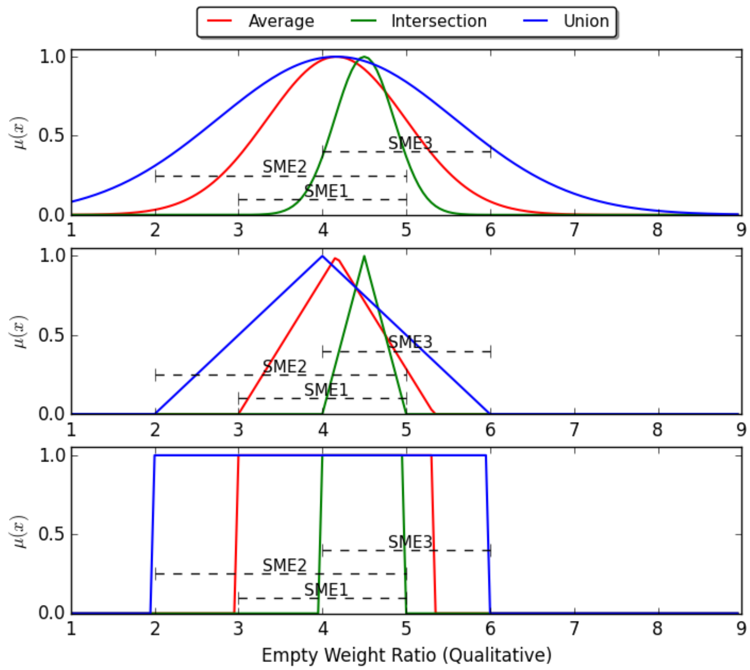


Figure 39: Means of Fuzzifying Expert Evaluations for 3 Example Evaluations

is outside the scope of this research.

3.2.4 Physics-Based Data Generation

In addition to the data gathered from expert evaluation, some physics based data was also required to explore the creation of modeling simple design tools in the framework. During this concept architecture exploration stage of design, some simple first-order tools are often used to get a basic idea of system performance. Here it was necessary to utilize tools that could evaluate all of the architectures evaluated. In the course of developing these tools it is often necessary to use a number of empirical parameters or use educated assumptions to generate needed model parameters. Because the generated data would be used to create fuzzy models (much in the same way regressions like response surface methods are used as surrogate models in modern conceptual design) much of the uncertainty behind these empirical parameters and assumptions could be taken into account.

To explore the use of training fuzzy systems with physics based data, two physics-based tools are used, simple momentum theory to evaluate hover figure of merit, and the more complex R_F method for system gross weight, installed power, and maximum speed. In order to understand the uncertainty inherent in using these methods at this stage in design (R_F method can also be used with higher fidelity approaches), the parameters not identified as being architecturally driven, as well as empirical parameters, are varied using Monte Carlo methods to produce a sample for each data point rather than singular crisp data. The sample of points was then used to create a range similar to the expert evaluations, sometimes through fitting a distribution to the results. A fuzzy membership function could then be created from the range representing a single data point.

In selecting the momentum theory approach to Figure of Merit, the decision was made to eliminate the vertical lift options of thrust from the propulsor and drive system functional aspects. Hover Figure of Merit does not usually apply to direct thrust lift systems, and momentum theory does not provide an approach that accounts or direct thrust. Moreover, direct thrust lift provides almost no chance of being as efficient as the system is intended to be. Only the three disk propulsor and drive options (rotor, propeller, and ducted fan) were carried forward for analysis in the final framework and following activities.

Figure of Merit

To generate data for a Figure of Merit system, simple momentum theory was used [63]. While momentum theory makes a lot assumptions for simplicity sake, it is often used to model power required for rotors in hover and forward flight and propellors in axial flight during conceptual design, as it does not concern itself with the details of blade shape. Momentum theory models the rotor as an actuator disk, adding energy and momentum to a incompressible, steady, inviscid, and irrotational flow through the disk. The volume around the disk and in the wake is modeled uniformly and in just one dimension. An expression for hover figure of merit by momentum theory is shown in Equation 56, based on the non-dimensional parameters in Equations 53 and 55.

$$C_T = \frac{T}{\rho(\pi R^2)(\Omega R)^2} \tag{53}$$

$$C_P = \frac{P}{\rho(\pi R^2)(\Omega R)^3} \tag{54}$$

$$C_P = \kappa C_T \sqrt{\frac{C_T}{2}} + \sigma \frac{C_{D0}}{8} \tag{55}$$

$$FM = \eta_d \left(\frac{\kappa C_T \sqrt{\frac{C_T}{2}}}{C_P} \right) \tag{56}$$

In order to generate data points, the main parameters listed in Table 13 were randomly selected, then the "noise" parameters were randomly varied (uniformly) to create a sample of 500 data points. The resulting minimum and maximum of the sample were then recorded as the possible range of Figure of Merit values possible for that particular set of main parameters. This was repeated to create 1000 random points that could be used for training of a fuzzy model.

Gross Weight, Installed Power, Maximum Airspeed

In order to get an idea of the potential sizes of a vehicle a particular architecture offered, the R_F method was utilized [102]. R_F method is a flexible, straightforward fuel balancing method, were vehicle gross weight is varied and the resulting weight fraction of fuel required for a sizing mission is compared to the fuel weight fraction available. When the aircraft is large enough to carry enough fuel to complete the

Table 13: Parameter Ranges: Figure of Merit Data

Main Parameters	Minimum	Maximum
Disk Loading (lb/sqft)	1	150
Download Factor (%GWT)	0.0	0.3
Solidity (-)	0.0	0.4
Drive Efficiency (%)	0.5	1.0
"Noise" Parameters	Minimum	Maximum
Gross Weight (lbs)	10,000	12,000
Tip Speed (ft/sec)	650	850
κ (-)	1.08	1.32
C_{d0} (-)	0.005	0.01

mission, a feasible minimum gross weight has been found. The algorithm also often incorporates the ability to calculate the size of the vehicle power plant necessary to perform the mission and/or meet some series of engine sizing conditions. The R_F method is flexible, and can be used at very low fidelities with many assumptions, or as part of a high fidelity design tool with detailed performance and weight calculations. For this research a simpler approach was used, assuming a fixed empty weight fraction, ϕ, and using first-order rotorcraft and fixed wing performance calculations.

Simple calculations are used to estimate power required for an aircraft in hover and forward flight based on a general tilting, compound, or other configuration (essentially a fixed wing in forward flight). The core of the code used is based on calculating power required in hover and forward flight for a number of potential concepts, using the process outlined in Figure 40 [63, 95]. These simple algorithms are harnessed to determine the gross weight at which a concept can carry the fuel necessary to carry out a sizing mission. The installed power (or thrust) is also determined as the maximum power (or thrust) needed to perform a number of engine sizing conditions and/or complete the mission.

The simple R_F algorithm was validated through comparison to existing sizing data, mostly generated by an advanced sizing, design, and analysis tool: NASA

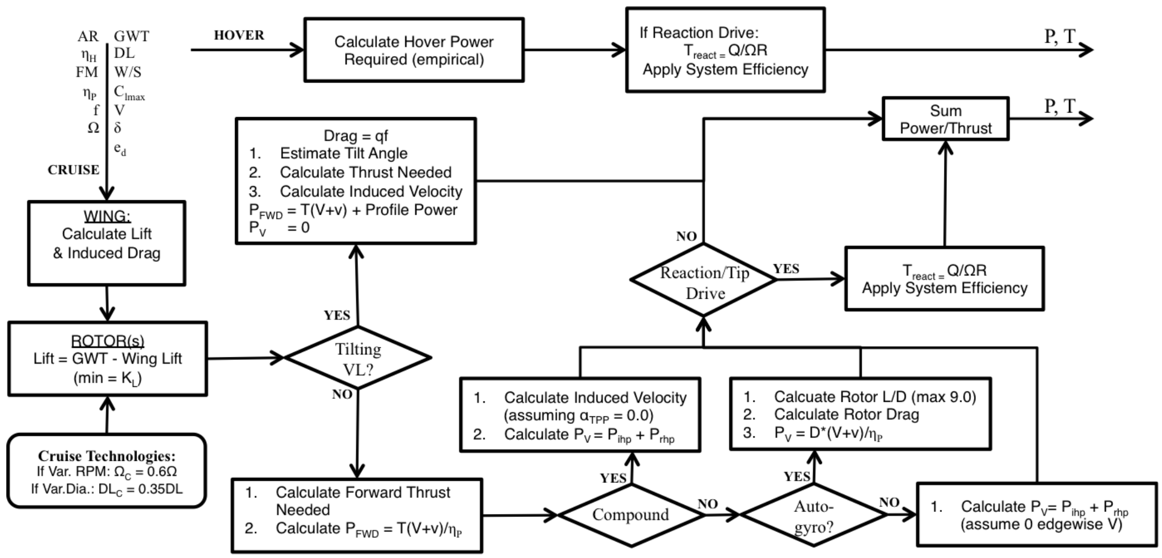


Figure 40: Expanded R_F Core Algorithm

Design and Analysis of Rotorcraft (NDARC). Very little literature exists with comprehensive data on the sizing of high speed VTOL aircraft, but several papers have been published outlining the design of advanced rotorcraft using NDARC. Thus, the algorithm was compared to a number of advanced tilt-rotor [52], several high-speed compound rotorcraft [53, 101]. Additionally, variable diameter tilt-rotor [112], and reaction drive stopped rotor/wing [112] sizing results from a non-NDARC source were used as validation for reaction drive systems. The R_F algorithm was only needed to provide first-order estimates of vehicle gross weight, power installed, and maximum flight speed, not match the exact answers of a very detailed program like NDARC that uses hundreds or thousands of design and empirical parameters and provides comprehensive analysis of rotorcraft. The R_F algorithm was also compared to a simple sensitivity method proposed in Reference [95] for the sizing of advanced VTOL aircraft using only 5-10 aircraft parameters and assumed a priori characteristics. A comparison of the three sizing methods is illustrated in Figure 41. Overall, the R_F algorithm provides a good first order estimate of weight, power, and maximum speed across a wide range of potential aircraft.

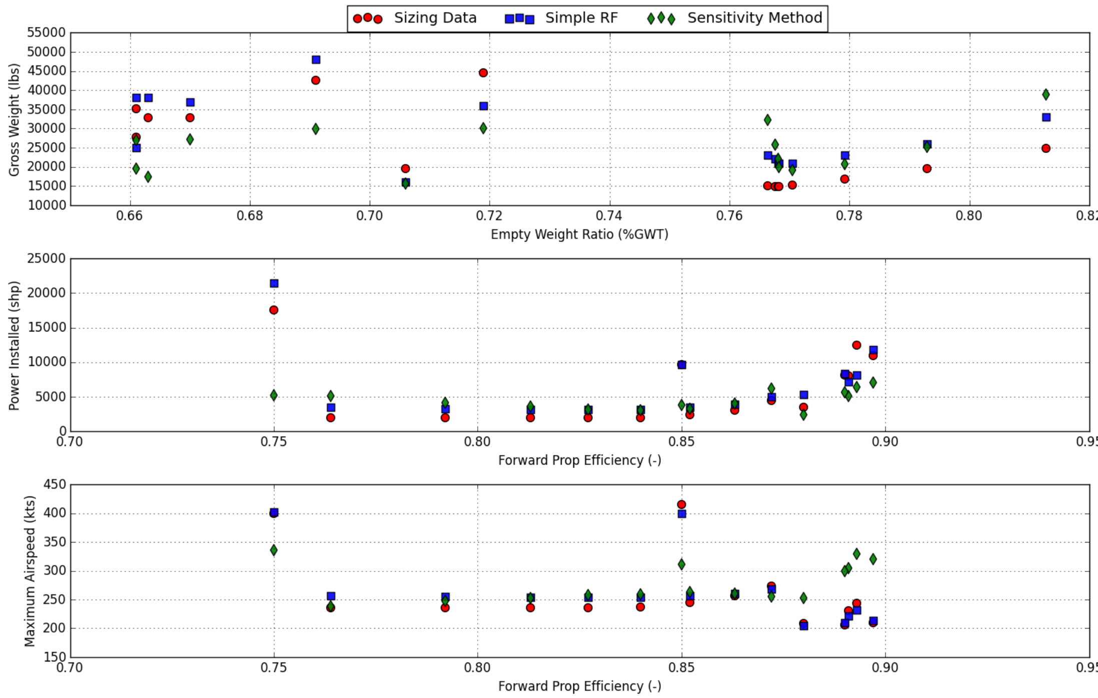


Figure 41: R_F Method Validation

To evaluate the potential concepts for the research example problem on common terms, the simple sizing mission and engine sizing conditions shown in Table 14 were utilized. A payload of 2240 lbs payload used in the sizing mission. This is 15% of 12,000 lbs (the minimum payload ratio required at the maximum allowable gross weight) plus two 220 lbs pilots.

In addition to the design gross weight and installed power, the aircraft's maximum speed was also calculated as this was a design metric in the original problem. However, because sizing the aircraft to be capable of 300 knots meant that in most cases the maximum speed at that condition was only 300 knots. This meant the aircraft that required a larger amount of power/thrust to hover were rewarded, as they used that power margin to achieve higher speeds in forward flight. If no maximum airspeed sizing condition was used, this effect would be further exacerbated as very efficient hovering concepts might only be capable of very low forward flight speeds based solely on their smaller engines.

In order to counter this dilemma, instead of evaluating architectures on maximum airspeed, they were evaluated on the required size of the installed engine. Aircraft that use less power to reach the required 300kts have smaller and lighter propulsion systems, critical to meeting the difficult goal of a 0.60 empty weight ratio. These systems had a better potential of reaching higher speeds by increasing the engine size, while still meeting empty weight goals. This way aircraft with very large engines (and higher cruise speeds) due to poor hover performance were not rewarded for their failure in hover efficiency. In order to compare installed engine size in a standardized fashion, system are all judged based on required installed power. The reaction drive systems still had their (relatively poor) hover efficiency accounted for, and forward flight power calculations included the efficiency of each propulsion type as necessary. Results for turbofan and turbojet systems were just presented in power based on the assumption that $V_{exit} = V_{freestream}$ (an optimistic estimate of thrusting system efficiency).

Similar to Figure of Merit, data for the R_F algorithm was generated for model fitting by iterating on the main design parameters while randomly varying a series of parameters that were empirical or not dependent on architecture at this early conceptual stage. The main and "noise" parameters are listed in Table 15, along with the ranges used. 250 random iterations were completed at each combination of main parameters (also randomized uniformly throughout the design space). To account for the occasional extreme outlier or un-converged solution, the resulting data for each combination was used to calculate a sample mean and standard deviation, and the ranges used for modeling each output parameter were $[\bar{x} - s, \bar{x} + s]$. This approach was used to reduce the amount of time necessary to calculate larger samples for R_F data.

Figure 42 shows the output data modeled as membership functions for each of the data points used to train the system to help illustrate the distribution of the

Table 14: RF Baseline Mission

Sizing Mission			
Segment	Time	Dist	Condition
Hover	5 min		SLS OGE
Cruise		300 nm	V_{BR} @ 14K ft ISA
Dash		50 nm	V_H @ 14K ft ISA
Hover	2 min		SLS OGE
Reserve	20 min		V_{BE} @ 14K ft ISA
Engine Sizing Conditions			
Hover	4000 ft, 95 deg F OGE		
Dash	300 kts 10K ft ISA		

output data for each training point. Some show a great deal of uncertainty, while other designs are more robust with respect to the "noise" parameters. Figure 43 shows representative histograms of the data gathered using the R_F method algorithm to illustrate the relative difficulty of the mission with respect to the constraints on vehicle size ($\leq 12,000 lbs$). In later design phases, gross weight could be tweaked by adjusting the mission range, but it would be desired to keep it at some reasonable value. It should be noted, that modeling the outputs as normal distributions results in the possibility of airspeeds less than 300 knots. Although individual designs were sized to be capable of 300 knots, if the mean of a sample is near 300 and has a sample standard deviation high enough, the distribution can be modeled below 300.

Table 15: Parameter Ranges: RF Method Data

Main Parameters	Minimum	Maximum
Empty Weight Ratio - ϕ (%GWT)	0.5	0.85
Disk Loading (lb/sqft)	1	150
Wing Loading (lb/sqft)	15	300
Hover Drive Efficiency - eta_H (%)	0.5	1.0
Propulsive Efficiency - eta_P (%)	0.6	1.0
Hover Figure of Merit (-)	0.4	1.0
Download Factor (%GWT)	0.0	0.3
SFC - SLS, MCP (lb/hp/hr)	0.35	0.75
General Configuration	Tilting VL, Compound, Other	
"Noise" Parameters	**Minimum**	**Maximum**
Aspect Ratio (-)	2.0	8.0
Tip Speed (ft/sec)	650	850
κ (-)	1.08	1.32
C_{Lmax} (-)	0.9	1.3
$RotorLift - Cruise$ (%)	0.0	0.25
Drag Multiplier (%sqft@GWT)	0.6	1.15

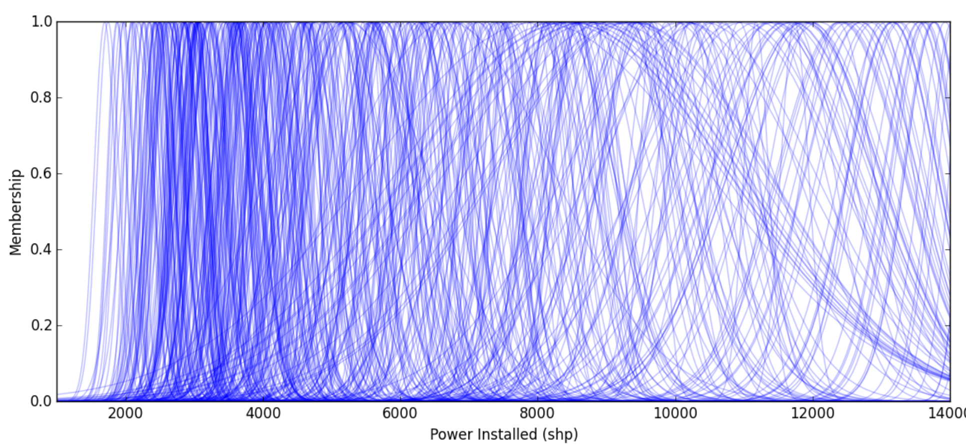


Figure 42: Output Membership Functions for Installed Power Training Data from R_F Algorithm

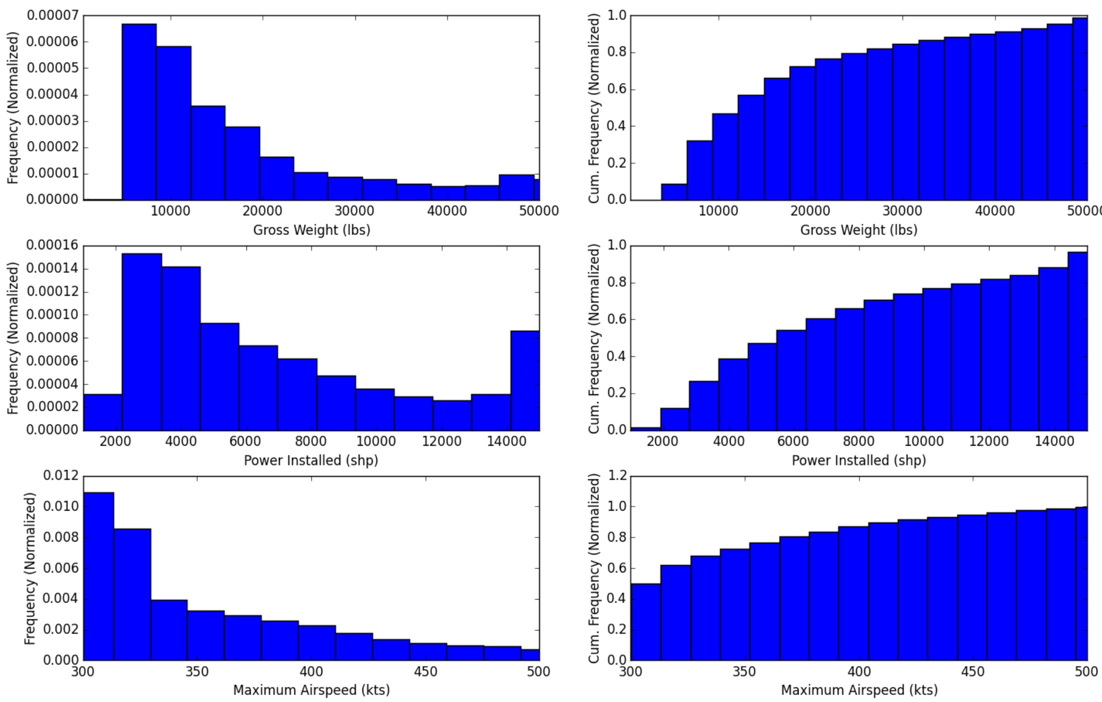


Figure 43: Generated Data from R_F Algorithm

3.2.5 Discussion

This portion of the research attempted to answer Research Question 3, which asked "How should expert data be elicited to build the Fuzzy Expert System(s)?". While the elicitation phase of this research did not proceed as planned due to lack of participation, some important lessons were learned. Because of the nature of problem of building models based on this data, the data elicitation evolved slightly throughout the course of the research, and some of the important lessons came from building the desired fuzzy models with the data as well as the process of gathering the data.

The interval method for elicitation proved to work very well, being relatively easy to explain and providing data flexible enough to be modeled several different ways. Adding the historical data to the architecturally dependent inputs was determined to be necessary during the elicitation process and not anticipated in the conjectures for this research question. While historical data was not readily available for each identified input, it provided some necessary context for generating the rest of the intervals when done before the expert provided intervals when provided alongside the survey. It was also found necessary to use the historical data to provide baselines for the qualitative data. For instance, the standard tilt-rotor configuration was set as the baseline for vertical lift system empty weight inputs, thus providing a point of reference for the scale. If an option was deemed to result in a heavier system than it's counterpart in the tilt-rotor it could receive a worse score. It also became obvious that combining expert ranges with historical data was best accomplished on a careful case by case basis. The best method to make decision on the input range to use proved to be a case by case examination of the aircraft identified for each functional aspect, how they might relate to the mission at hand, the relationship of that functional aspect to others, and the expert data.

Correlations between the inputs identified and criteria investigated were conjectured to provide a means to construct simple rule-based fuzzy systems. As discussed in Section 3.3.1, these correlations turned out to be inadequate to produce the required systems, and were dropped from the survey. They did not provide critically important information about how interactions between input parameters effect the output criteria. This conjecture turned out to be an inadequate means of collecting data.

Lastly, it was hypothesized that expert data sets to train NFSs can be generated by evaluating a set synthesized alternatives. During the process of attempting to gather evaluations of synthesized systems for training expert systems with data, several more lessons were learned. It was evident fairly early on that the expert(s) would have to iterate through the evaluations several times to refine the scores. As new alternatives were continually presented, the user gains a better understanding of the architectural combinations available, and has realizations about the performance of various subsystems. To achieve consistent data, it is necessary to iterate back through previous evaluations, reviewing them and altering where necessary. The online tool created for generating and evaluating alternatives proved to work fairly well, but some improvements were recommended. Instead of evaluating a single alternative at a time, it was suggested that previous evaluations of several related alternatives might be presented simultaneously with new ones, and the user be permitted to change all evaluations. This might help create more consistent data in less time. The synthesized evaluations were originally conjectured to be a means to train NFS, but the NFS utilized were later found to provide a poor model of expert data in Section 3.3.8. However, the data turned out to be necessary to train more complex rule-based systems. So while the conjecture that evaluation of synthesized alternatives could be used to train NFS was found to provide non-ideal results, the data was useful. However, the best way to perform this elicitation task could use some refinement and future work. In all the hypothesized approach worked well enough, and showed a great deal of promise for improvement. There is no reason to believe the hypothesis should be rejected. More discussion of how this data was used is continued throughout Section 3.3.

Physics-based data proved relatively easy to generate with the right models. The most difficult part of this process was validating the codes before generating data. The fitting of the R_F data later in this research indicates that very complicated codes with more inputs might require orders of magnitude more training data points to fit properly. Generating this data is not as much of a problem as the time the fitting process takes to iterative over the data set. In these cases it might be worthwhile to try and minimize the number of inputs used, and let more parameters be varied as noise.

Additionally, while serious participation in the surveys was not sought, several available subject matter experts were asked to take the time to provide some feedback and impressions of the surveys used. One suggestion was to ask the expert(s) to decompose the effects of the system into the individual effects of subsystem during the pre-selected alternative evaluations, then ask them to compare the combined effects they provided with an overall evaluation. It is thought this might provide more context for the following phases of the survey. Preliminary feedback from the survey also was responsible for the development of baseline systems/functional aspects for the qualitative parameters. The other primary suggestion provided was to provide more context for the survey by giving a baseline or example mission for the system. These suggestions provided valuable input for developing the survey/elicitation process, and could be further analyzed in future work related to this research.

3.3 Fuzzy System Models

The next step in the research process was to develop fuzzy systems to serve as models for the various system attributes of interest, supplanting expert evaluation or simplifying physics based tools. This is research was conducted to answer Research Question

2, and produce models for the larger framework in Research Question 4. Again, it is important to note that the development of these systems and the elicitation of the data used to drive them are closely linked.

The purpose of this effort is to demonstrate several ways how fuzzy systems can be used to model an expert evaluation or first-order tool evaluation of a concept with respect to an attribute of interest while capturing the uncertainty therein. In following with the hypotheses outlined for Research Question 2, the first experiments in trying to answer this question explore the use of Fuzzy Rule Based System (FRBS) to logically model an expert's evaluation of an alternative with respect to some criteria. Simple systems with expert defined rules are first created, before moving onto the second hypothesis, where rule-based system are trained with data, both from expert evaluation and physics-based tools. Because the inputs to these systems will be fuzzy, representing uncertainty in design parameters inherent during concept selection, the systems are designed to use fuzzy inputs, drawn from elicited input ranges. Similarly, the outputs to the system are to be kept as fuzzy membership functions, to represent the uncertainty in the outputs. Various means will be explored in the next section to create the best systems based on that uncertainty. Lastly, NFS are utilized in accordance with the third hypothesis. A couple means of implementing an NFS are explored, and then the most promising method is used to train fuzzy models for both expert evaluations and physics-based data.

3.3.1 Fuzzy Rule Based System (FRBS)

The FRBS implemented as expert systems within the framework are exclusively Mamdani systems, using (max-min) composition, as outlined in Section 2.2. These systems can provide both the desired fuzzy outputs and add a defuzzification step to the end to produce a single crisp output. A generic FRBS class was developed in Python to support the framework as discussed in Section 3.3.1. The class would later be evolved into an OpenMDAO component for integration into a larger framework.

Defining architectures for each of the FRBSs was a critical step in their development. The overall system is required to evaluate each concept with respect to a number of criteria. Each criteria is handled by an individual MISO system, independently structured and trained. The appropriate inputs had to be selected and then generated from the input data provided in reference to the available morphological options.

Two methods were explored for generating FRBSs. The first was a manual method, where experts familiar with the problem were asked to first evaluate the problem, determine the appropriate inputs to assess the given criteria. The experts then generated both input and output membership functions, as well as a list of rules relating the inputs to the outputs, based on their experience and intuition. It was attempted to create rule bases based on expert elected correlations of inputs to outputs, as hypothesized earlier. But this proved to be a flawed method, as the correlations did not provide the necessary relationships between input parameters. A Mamdani style inference system was used in these simple systems to allow the expert to evaluate relationships through simple IF-THEN statements. This method proved to work adequately for criteria with only a few (2-3) inputs and relatively simple, straightforward relationships to generate an output. An example is illustrated for system L/D ratio in Section 3.3.3.

However, the desired architectures for many of the systems desired rely on larger numbers of inputs, and even at a simplistic 3 membership functions for each input, a system with 5 inputs would require the decision maker to address $3^5 = 243$) possible rules. Even with careful management and partitioning of rules to reduce the final rule base size, the results proved to be inconsistent and often span most or all of the possibilities in the output universe. This was not entirely unexpected, as it was considered likely that systems trained based on "truth" data sets might provide superior performance.

There have been a number of proposed approaches for learning in FRBS with data sets, as well as a number of means to tune or optimize these systems [2]. Several means were considered before deciding on the method outlined below. One important consideration in both the trained and the expert defined FRBSs was tradeoffs between accuracy and interpretability. Adding inputs to a given system, adding membership functions to each input domain, or adding more and more rules can all increase the accuracy of a fuzzy system. But with this comes the cost of system complexity. As illustrated in Figure 44, each of these decisions move the system in a tradeoff between this accuracy and complexity, where complex systems are more difficult to understand and interpret [28]. Sometimes a more complex system is necessary to model the desired system attribute, but in general the desire was to have simpler systems with easy to understand logic. It is also imagined that in a normal application, most of the system attributes of interest will be capable of being modeled by simple systems (especially non-performance attributes, like reliability), while only a few will require more complicated training. After all, the general purpose behind this research is to provide lower fidelity results, just good enough to provide context for the problem and screen for the best alternatives. In this example problem, however, performance attributes were purposefully selected to demonstrate a range of potential approaches.

Basic FRBS Training and Optimization Procedure

To accurately generate more complicated FRBSs, a training and optimization algorithm was developed. Of the most common training algorithms, a relatively straightforward approach was used [127]. The approach uses each provided data point to create a rule (or rules) based which membership functions in the consequent (input space) and antecedent (output space) the data point belongs to. The algorithm then selects the rules for each antecedent combination that are most applicable (have the

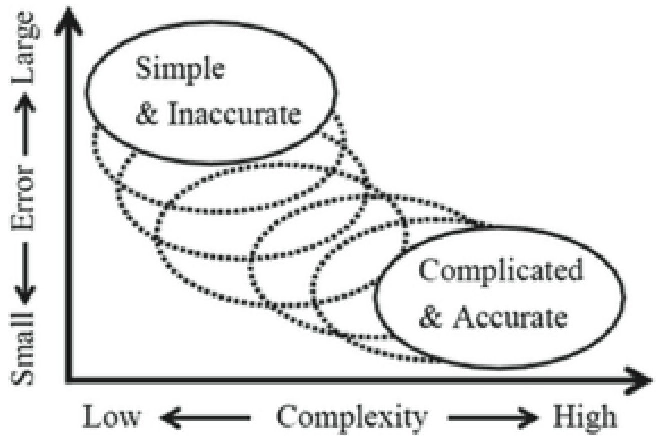


Figure 44: Accuracy vs. Complexity in a FRBS (from [28])

highest combined input and output firing strengths) among all those created. Alternate versions of this algorithm allow for multiple rules to be created with two output membership functions for the same combinations of inputs, or determine the best rules in different manners [2], but the method described below provided the most promising results. This training algorithm was then wrapped with a optimizer to optimize the input and output membership functions to create a more accurate system [2, 28, 81]. The procedure used is outlined below, and example is shown in Section 3.3.4.

1 - Generate Baseline System: First a baseline FRBS is generated by dividing the domain of each input into some number of regions defined by fuzzy membership functions $(\mu_1(x_i), \mu_2(x_i), ..., \mu_m(x_i), \;\text{ for } x_1, ...x_n)$. The output space is then treated similarly $(\mu_1(y), \mu_2(y), ..., \mu_m(y))$. Various shapes and numbers of membership functions can be used based on the nature of the inputs and outputs. The baseline system begins with an empty rule base.

2 - Generate Rules from Training Data: For each input-output pair in the training data set, $(x_1^{(i)}, x_2^{(i)}, ...x_n^{(i)} : y^{(i)})$, the input membership functions that correspond to the maximum degree of fulfillment, or firing strength for each input are determined, such that for each input membership function selected $\mu_j(x_i^{(1)}) > \mu_k(x_i^{(1)}), \; \forall\, k \neq j$.

These membership functions then constitute the antecedent for a rule corresponding to the data point used. In the same manner, the consequent is determined by finding the output membership function with the highest degree of fulfillment for the data point output, such that $\mu_j(y^{(1)}) > \mu_k(y^{(1)}),\ \forall\, k \neq j$. This procedure is repeated for each data point given generating a single rule based on the most relevant membership functions for each input and output value (or fuzzy set for fuzzy inputs and outputs).

3 - Determine Rule Base: While rules are being generated, each is assigned a strength based on the combined fulfillment of the input and output membership functions in the form $S(Rule^{(1)}) = \mu(x_1^{(1)})\,\mu(x_2^{(1)})\,...\,\mu(x_n^{(1)})\,\mu(y^{(1)})$. Each generated rule is added to the rule base. Where a group of generated rules have the same antecedent, but different consequents, the average strength of each generated consequent among the generated rules is calculated, and the consequent with the highest average strength is added to the rule base.

4 - Optimize System Membership Functions: Once the system rule base was defined, a workable FRBS exists, but in order to improve the accuracy of the system, the original input and output membership functions are adjusted to fit the data best. This is called system optimization or sometimes system tuning. This step actually wraps steps 2-3 in some optimization method, so at each iteration the membership functions are adjusted, then new rules are generated and a rule base selected. The system's error against some set of data is calculated (see Section 3.3.2), and the optimization continues as required. The details of developing an optimization process are discussed further in Section 3.3.4.

In order to facilitate the generation and training of FRBSs for this research, a custom Python package was developed, which is described in more detail in Section . The specific results for the training and optimization of several example FRBSs performed for this research are discussed in following Sections, 3.3.3 through 3.3.5.

3.3.2 Fuzzy Error Functions

One important consideration for developing fuzzy systems for the larger framework, was how to measure the error between the system and the training data. The data elicited from experts was fuzzy in nature, with possible ranges for both inputs and outputs. However, the data built from simple physical models had both crisp inputs and outputs. The fuzzy systems used could provide both fuzzy and crisp outputs by a simple toggling the addition of a defuzzification step performed on the system output. This provided some help in comparing system outputs in error calculation, but still meant that an algorithm was needed to compare fuzzy system outputs to fuzzy truth values.

Several measures of fuzzy error were considered, but two common methods proved to be the most useful defined. The first is an integrated root mean squared error (RMSE), as defined in Equation 57, [100]. This error approximates root mean squared error of the membership differences across the domain.

$$RMSE(A,B) = \sqrt{\frac{\int_{y_{\min}}^{y_{\max}} \left(\mu_A(y) - \mu_B(y)\right)^2 dy}{y_{\max} - y_{\min}}} \tag{57}$$

A distance error based on α-cuts was also explored [75]. This measurement has the added benefit of being directional, using the Hausdorff metric (Equation 59) for determining directionality in intervals, and applying it to each α-cut comparison. The distance measure weights the distance between α-cuts with α and accounts for non-normal fuzzy sets by using the supremum of all the cuts where α-cuts exist for both sets. The distance measure is defined in Equations 58 and 59 .

$$\begin{gathered} err_{AC}(A,B) = \frac{\sum_{\alpha \in [0,\lambda]} y_\alpha \, h(A_\alpha, B_\alpha)}{\sum_{\alpha \in [0,\lambda]} y_\alpha} \quad \text{where} \\ \lambda = sup\{\alpha \in [0,1] \, : \, A_\alpha \neq \; \vee B_\alpha \neq \} \end{gathered} \tag{58}$$

$$h(A_\alpha, B_\alpha) = \begin{cases} B_\alpha^L - A_\alpha^L & if \left|B_\alpha^L - A_\alpha^L\right| > \left|B_\alpha^R - A_\alpha^R\right| \\ B_\alpha^L - A_\alpha^L & \text{otherwise} \end{cases} \tag{59}$$

Both the RMSE and α-cut method were tested trying to match several types of membership functions by feeding their parameters into a gradient descent optimizer and using the given error. The methods proved to be quick to calculate and accurate in comparing membership functions. However, because the RMSE error method scaled the squared sum of differences by the domain range, as fuzzy numbers diverged only small variations were created in the RMSE error. For this reason, the α-cut method was used to train and tune a more accurate system. This is illustrated by an example of error values for the two methods is illustrated in Figure 45.

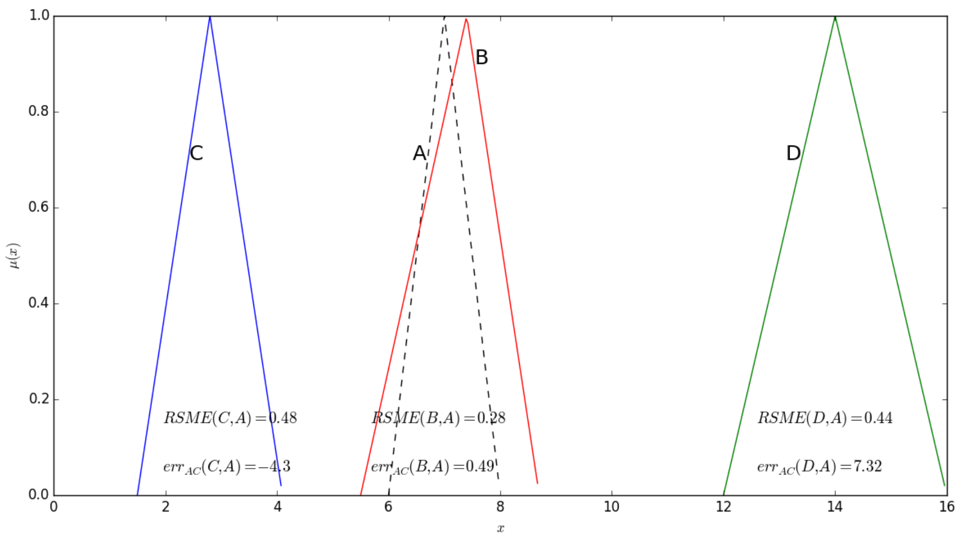


Figure 45: Examples of Fuzzy Error Methods for Various Membership Functions

System error across some set of truth data is assessed using a Mean Squared Error (MSE) measure as the average square of each output fuzzy membership function's error, defined in Equation 60 for m data points.

$$MSE = \frac{1}{m}\sum_{i=0}^{m} err(A_i, B_i)_p{}^2 \quad \text{for } i = 1, ..., m \tag{60}$$

It should be noted that due to the nature of defining fuzzy and system error in this manner, the fuzzy MSE defined here should not be directly related to a statistical MSE measure. System fuzzy MSE measures here are used to compare the fit of one system against another and not to other statistical data models.

3.3.3 Building a Simple FRBS

The first step in exploring the application of FRBSs was the generation of a rule-based expert system to evaluate L/D ratio through simple expert defined rules. Two parameters were considered to define a potential architecture's overall L/D, the L/D ratio of the lifting surface(s) (the wing system), and the estimated drag of the system in cruise condition.

Using the functional architecture defined by the morphological matrix, the first FRBS input was the lifting system L/D ratio, calculated as the intersection of the wing system and the vertical lift system technology L/D parameters. The second input was the relative system flat plate drag, calculated as the intersection of each relevant functional aspect's qualitative flat plate drag area (from the input survey: VL System Type, VL System Technology, FWD System Type, Wing System). These inputs were defined by 3 and 5 triangular membership functions each. The output system L/D was defined in the FRBS by 5 trapezoidal membership functions. This combination of input and output membership function types and sizes was found to provide the best differentiation and accuracy of outputs, while leaving a manageable but accurate number of rules to define. A representation of the overall expert system architecture, showing the flow of data from the morphological matrix to the system, is shown in Figure 46.

The relationships between inputs and outputs were captured using 13 rules (from a total possible 15 combinations of input membership functions), which are outlined in a fuzzy control language file in Appendix C. An example of the inference mechanism is shown in Figure 47, showing the system's calculation of L/D ratio for Alternative 1, the baseline Advanced Tilt Rotor. Here the firing strength of each rule is shown applied to the consequent membership functions, and then the final aggregated output membership function is illustrated. The output for the system was left fuzzy (no defuzzifier was used) to allow the system to capture the uncertainty inherent in the model. A number of means exist to more simply capture the uncertainty in the final fuzzy function and display it to the designer. These means are discussed in the implementation of the final framework.

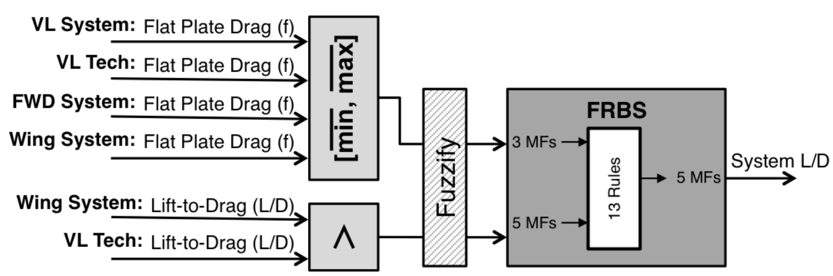


Figure 46: L/D Expert System Architecture

There is no straightforward means to validate this type of simple FRBS system, as it is a direct construction of expert's evaluation of the relationships between the architecture possibilities and projected system L/D ratio. However, to illustrate the capability of the system, and to compare it's performance to traditional means of expert architecture evaluation, the system was exercised for the 10 pre-selected alternatives, evaluated in the benchmark. The resulting system outputs are compared to each alternative's evaluation from the benchmark for L/D in Figure 47. To more simply visualize and understand the system performance, α-cuts of the system's output were taken at the maximum membership value, and shown along-side the ranges experts provided for the system evaluations, along with a percentage of overlap for the two ranges.

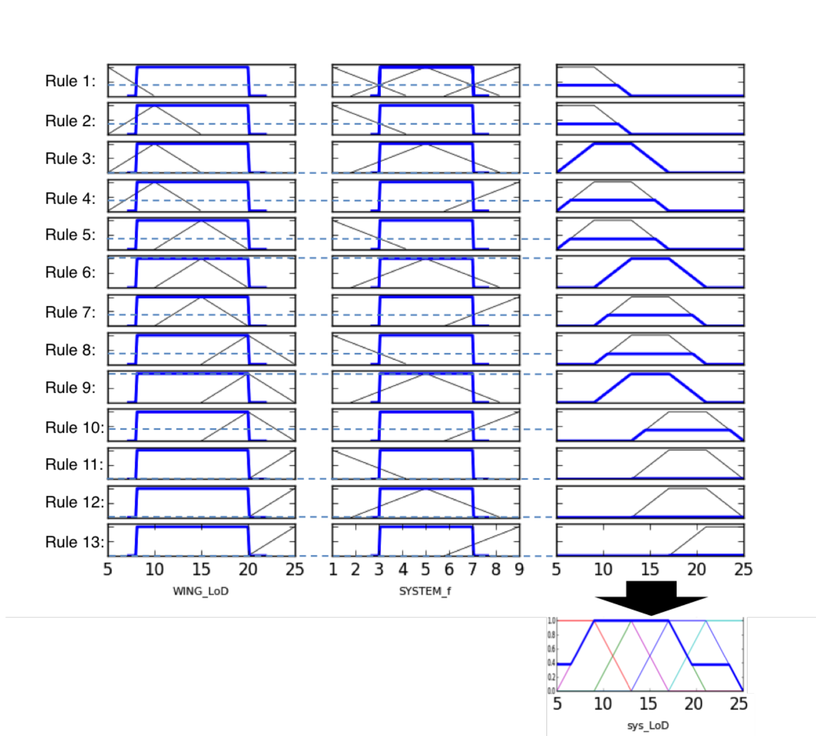


Figure 47: Inference Mechanics of the Example L/D FRBS for a Specific Example (Alternative 1: Variable RPM Tilt-Rotor)

The system matches expert evaluation fairly, though it generally is less certain about it's answers than the experts. The Stopped Rotor configuration has the worst match at only a 60% commonality to the expert range. This is probably not unexpected considering the very unusual configuration. With more expert evaluations, more effort could be made to tune the system membership functions and rules to match them. However if the effort is made to collect a reasonable number of expert evaluations for a given system characteristic, a more mathematical approach to creating and tuning the system is appropriate. This approach is discussed in the next section

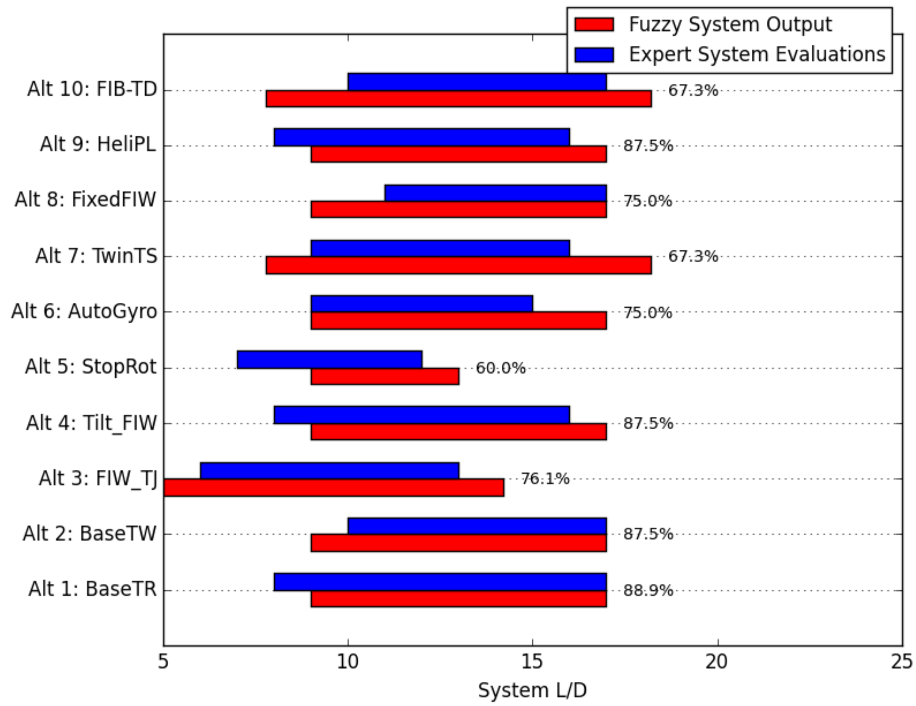


Figure 48: Performance of L/D System for Baseline Alternatives (α-cut at $\max(\mu)$), with % Overlap Shown.

3.3.4 Creation and Training of FRBSs from Expert Elicited Data

The process used to understand how usable FRBSs were created and trained is outlined in this section through the development of an expert system to evaluate system empty weight (ϕ). The training data for this system consisted of interval evaluations of compatible architectures as generated by the online tool described in Section 3.2.2. As outlined previously, these evaluations are on a qualitative 1 - 9 (worst - best) scale. Each of the previously described system design steps for training an FRBS - input selection, membership function selection, and system optimization - are inextricably linked, with decisions made in one affecting the results of the others. Due to the computational time necessary to optimize the larger FRBSs, and the large number of possible combinations of input parameters and membership functions, the following experiments were developed based on some initial exercises into the best way to generate a system. Prior to membership function optimization/tuning, all antecedent and consequent membership functions were baselined as evenly distributed over their respective domains. For gaussian functions, the mean of the first and last function were placed on the minimum and maximum of the domain respectively, while the baseline standard deviation was calculated based on the number of membership functions used (n_i) as $STD = \frac{1}{2n_i}(max_X - min_X)$. For triangular and trapezoidal membership functions, the functions were also distributed evenly with the mid-point of the the first and last function set on the minimum and maximum. The distance between stepwise vertices of these functions were each $x_i = \frac{2}{n_i m_i}(max_X - min_X)$, where m_i is 3 for triangular functions, and 4 for trapezoidal functions.

Training Data Evaluation

In addition to exploring the inputs and membership functions used in the FRBS, it was necessary to look at the amount of data needed to adequately train a system. Again, selecting the number of data points was inextricably linked to the inputs selected and membership functions used. However some idea of how much data was required was necessary early in the process to gather the correct amount of data, as elicitation was time consuming. After some initial investigation into the best inputs to use and the best membership functions for an accurate system, a simple experiment was done to estimate if enough data had been collected. A baseline FRBS system was created with 12 inputs (based on early trials in input membership function selection), with 7 gaussian input membership functions each, and 9 gaussian output membership functions.

This system was trained using steps 1-3 from Section 3.3.1 10 times with a random sample of N data points. The number of data points, N, was increased steadily and the error recorded for each sampling. The error was calculated with respect to a consistent set of 150 validation data points. The results of this experiment are shown

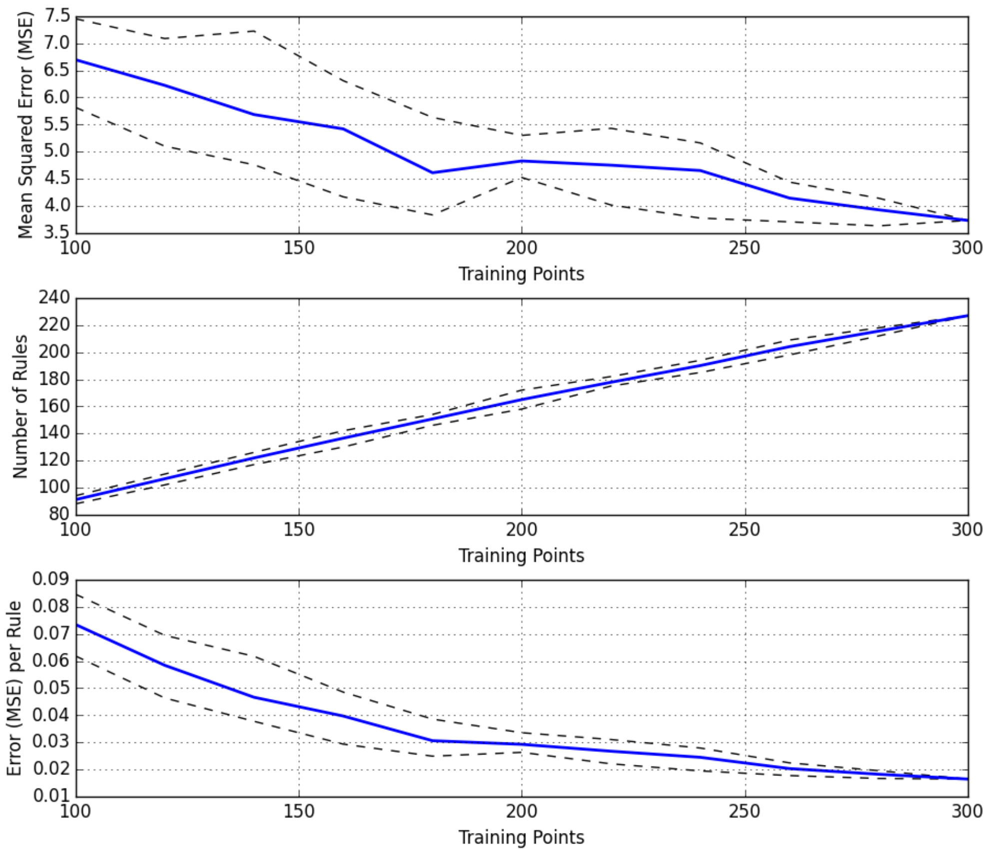


Figure 49: Effects of Training Data Set Size on Baseline FRBS for ϕ (Min, Mean, Max)

in Figure 49, with a the minimum, mean, and max illustrated at each point shown.

The MSE decreases steadily for the most part as the number of rules found in the data increases. The third subplot is revealing however, as the amount of error reduction achieved relative to the number of rules added decreases non-linearly. After approximately 250 training points, the slope of this plot indicates less the MSE decreases less than 0.005 for each rule added. (0.1% of total MSE). At this point it was deemed that the expense of collecting more data and the accuracy added was not worth the time required, the computational expense of training a larger system. For these reasons, 300 training data points was deemed sufficient, and experiments in system training continued.

Input Selection

The first step in actually creating the system was to identify a set of input parameters that would result in an accurate system while still being manageable. Because the number of possible rules in a FRBS ($N_R = \prod_{i=1}^{m} (\#\, of\, MFs)_i$ for inputs $1, ..., m$) increases exponentially with each input, limiting the number of inputs is critical to managing the complexity of the system as well as its computational cost including training. While specific inputs were added for each criterion as a part of elicitation process, various experts might consciously or subconsciously consider a number of potential system parameters in their evaluations. For this reason all inputs were considered for system parameters at first.

Very little literature exists to support this specific case of selecting inputs for an FRBS system in a reasonably simple fashion. Input selection is often accomplished through fuzzy clustering based, identifying inputs and membership functions by finding clusters of data that fit together with respect to various parameters or criteria. However, the limited discrete nature of the input data from expert evaluation, and the continuous nature of the input domains lent itself to a more straightforward method for input selection. A methodology for input selection based on tracking system error through a simple addition and removal process was investigated, but the easiest method to select an input set was found to be optimization. For this purpose the research used a differential evolution algorithm already available in the Python modules the FRBS was built with [109]. This simple metaheuristic for minimizing possibly nonlinear and non-differentiable continuous space functions seeks to improve a population of candidate solutions, moving them around in the available space by combining portions of solutions based on their quality. It has also been shown to work well for discrete and combinatorial problems such as the one here [66]. An implementation of the algorithm is readily available in the SciPy/NumPy software package already in use to model fuzzy systems (see Section 3.3.1).

Based on initial investigation into the best type and numbers of membership functions to use for the FRBS and training data, a baseline FRBS of 5 gaussian membership functions for each input and 7 output gaussian membership functions was utilize in the approach. The membership functions were distributed evenly throughly each input/output range, with the first and last membership function mean assigned to the minimum and maximum respectively. The input and output data were modeled using gaussian fuzzy membership functions, with the range representing 4 standard deviations ($\sigma = 0.25(\max - \min)$). The optimizer was asked to select the combination of inputs that gave the system the lowest MSE across 150 randomly selected but constant validation inputs. At each iteration a rulebase was created based on 250 data points, and then a fuzzy inference system (Mamdani) created from the rulebase and membership functions. Then the system was tested against the selected validation data.

Figure 50 shows the average MSE for tested systems with each input included and not included. The bars outlined in black indicate the final 15 inputs selected, which are also listed in Table 16. The system with these 15 inputs had a MSE of 2.548, and was the third best of those explored. The best system had an MSE of 2.538, and had 16 inputs. The selected input combination was the least number of inputs in the top 10. Additional inputs meant additional rules, and additional computation and training time. While the system calculation time was on the order of milliseconds for each rule (0.003 s/rule), when this is multiplied by the number of extra rules and the number of times the system runs, the time was significant in the sense of a larger framework of models.

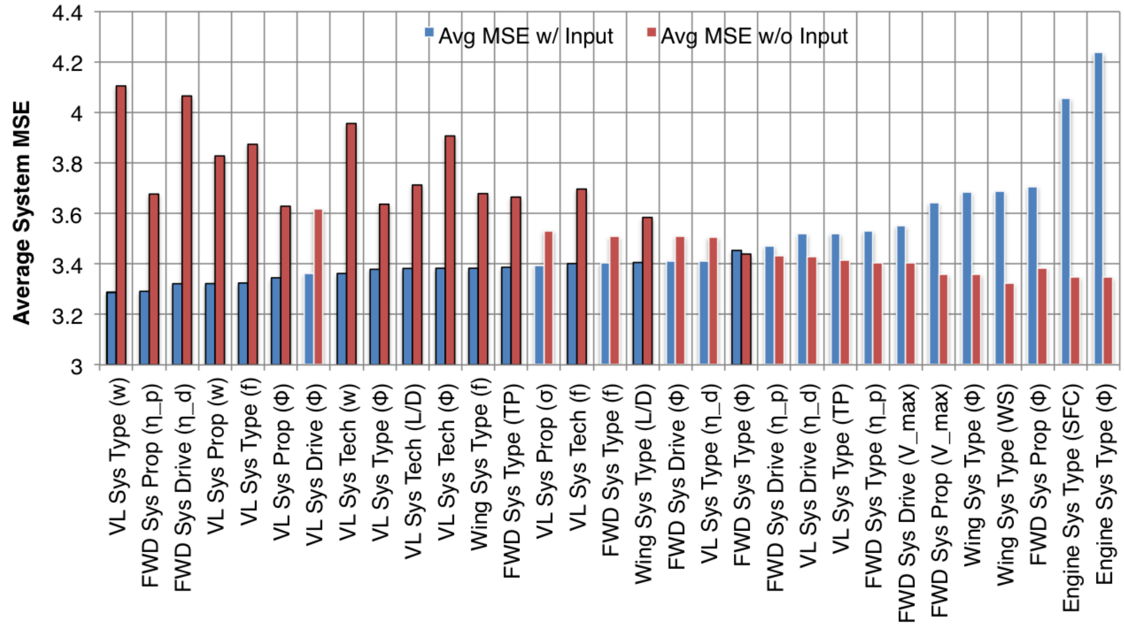


Figure 50: Inputs Effects on System Accuracy (from optimization)

Table 16: Final Inputs for Empty Weight FRBS System

Function	Input	Function	Input
VL System	ϕ	FWD Propulsor	η_p
	w	FWD System	ϕ
	f		T/P
VL Propulsor	ϕ	FWD Drive	η_d
	w	Wing Type	L/D
VL Technology	ϕ		W/S
	w		
	L/D		
	f		

Membership Function Selection

Once the input parameters had been determined, the next step was to explore the use of different types of membership functions for the system and data, and different numbers of membership functions in the system. This was accomplished once again using steps 1-3 of Section 3.3.1, repeating the training for different combinations of system membership functions and data membership functions. A training set (rule generation) of 250 data points was used, with 150 points used for validation. This provided some independence of the validation data, while reducing the time required to validate the system against all 300 points or gather additional data. The results of the experiments are shown in Table 17. While each membership function type represents the uncertainty in a slightly different way, the nature of fuzziness and membership functions lend themselves to some subjective interpretation. With this in mind, the goal was to select the membership function types for the system and data that resulted in an accurate representation of the data.

It is apparent that overall the gaussian and triangular membership functions significantly outperform the trapezoidal functions for both system and data functions. The best combination of functions is consistently shown to be gaussian functions for both the system and data. Moreover, it seems that in this application increasing the number of membership functions in the consequent proved to add more accuracy than adding membership functions to the antecedent. This trend indicates that nuances in the number of input combinations were less important than the ability to more finely granulate the output space. Thus roughly the same number of rules could be more accurate if permitted to map to a more detailed output space.

Table 17: Training MSE for Various System and Data Fuzzy Membership Functions

		Data Functions		
		Gaussian	Triangular	Trapezoidal
System Functions	Gaussian (5 In - 7 Out)	2.266 *(178 rules)*	2.956 *(228 rules)*	4.221 *(260 rules)*
	Triangular (5 In - 7 Out)	3.852 *(195 rules)*	4.114 *(213 rules)*	4.569 *(176 rules)*
	Trapezoidal (5 In - 7 Out)	4.231 *(195 rules)*	4.025 *(213 rules)*	4.610 *(176 rules)*
		Gaussian	Triangular	Trapezoidal
System Functions	Gaussian (5 In - 9 Out)	2.548 *(178 rules)*	2.956 *(228 rules)*	3.674 *(260 rules)*
	Triangular (5 In - 9 Out)	3.489 *(195 rules)*	3.215 *(213 rules)*	3.541 *(176 rules)*
	Trapezoidal (5 In - 9 Out)	4.231 *(195 rules)*	4.024 *(213 rules)*	4.610 *(176 rules)*
		Gaussian	Triangular	Trapezoidal
System Functions	Gaussian (7 In - 7 Out)	3.231 *(244 rules)*	3.108 *(236 rules)*	3.758 *(238 rules)*
	Triangular (7 In - 7 Out)	3.357 *(244 rules)*	2.283 *(236 rules)*	4.139 *(211 rules)*
	Trapezoidal (7 In - 7 Out)	4.034 *(244 rules)*	5.318 *(236 rules)*	4.241 *(211 rules)*
		Gaussian	Triangular	Trapezoidal
System Functions	Gaussian (7 In - 9 Out)	3.342 *(244 rules)*	2.780 *(236 rules)*	2.735 *(238 rules)*
	Triangular (7 In - 9 Out)	2.752 *(244 rules)*	3.829 *(236 rules)*	2.926 *(211 rules)*
	Trapezoidal (7 In - 9 Out)	4.034 *(244 rules)*	5.317 *(236 rules)*	4.242 *(211 rules)*

In addition to accuracy, it was also meaningful to understand the effects of membership function selection on system computational cost. Adding more rules and more membership functions to the database both mean a trained fuzzy system would take longer to provide an output. In general the systems all performed in a range of about 0.002 - 0.004 seconds per rule. For the systems explored, this meant the systems calculated in roughly 0.4 - 0.8 seconds depending on the number of rules trained. These numbers were averaged over 50 random data points. The difference may not seems significant, but with systems for many design criteria it could take significantly more time to explore the available design space.

A general trend shows that using trapezoidal inputs resulted in slightly less computation time for the system (likely due to their simple shape when calculating firing strengths and aggregating outputs). This resulted in a roughly 5-15% decrease in system calculation time, as averaged across 50 random data points. Despite this advantage, the accuracy of trapezoidal functions proved too low for use.

The system selected for optimization consisted of 5 gaussian input membership functions per input and 9 gaussian output functions. Data would be modeled using gaussian functions as well. This system was thought to be a balance between accuracy, interpretability, and computational time. In addition, it provided slightly more output membership functions than the 5 input - 7 output systems for optimization. Gaussian functions also provided the least number of optimization parameters, being defined by just 2 parameters, to 3 for a triangular and 4 for a trapezoidal. The triangular and trapezoidal functions could be made symmetrical, reducing their parameter count by 1, but at the expense of some of their flexibility.

In addition to the raw MSE, several methods of assessing the system's goodness of fit were explored. Figure 51 illustrates several of these methods, which are based on those used commonly in statistical model regression. Figure 51a is generated by taking α-cuts of the data (actual) and model output (predicted) at several α values and plotting the minimums and maximums against one another. It is apparent that at higher α values, the model closely matches the data, but as the values drop, the model is generally more uncertain. This is also apparent in Figure 51b, which shows the centroids of each data point plotted against the model output centroids. The model is more conservative, with model centroids skewed towards central values, indicating increased uncertainty in extreme results. Figure 51c shows the relative fuzzy error (as normalized by the actual data centroids) against the centroids of the actual data. The errors are largest at the lowest rankings (partially a function of the normalization), but also skews towards the center, with higher rankings tending to have negative error. Finally, Figure 51d shows the membership functions of several random data points alongside the model outputs for those points.

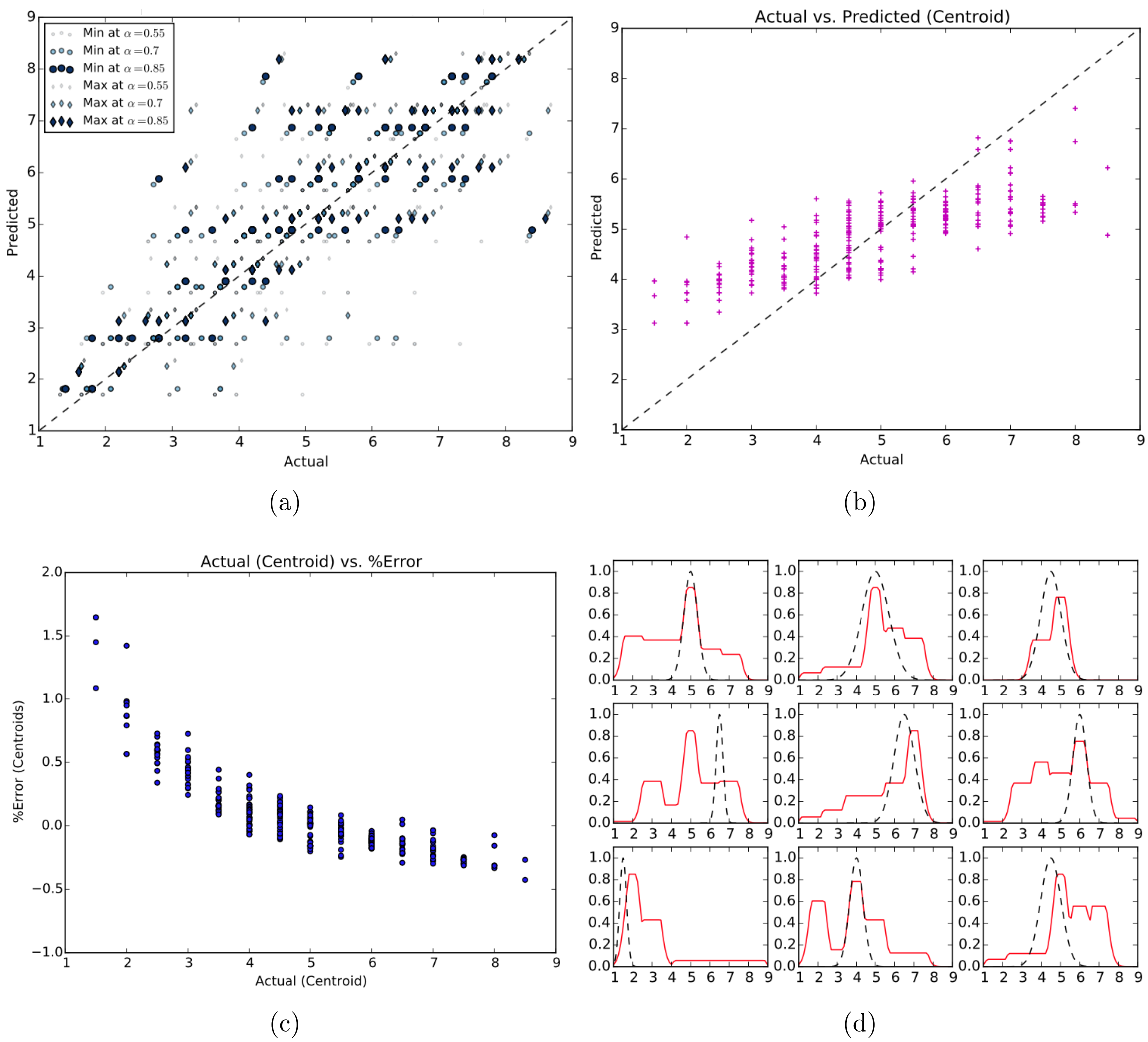


Figure 51: Goodness of Fit Checks for Unoptimized Empty Weight FRBS: a) Actual vs. Predicted for Various α-cuts; b) Actual vs. Predicted (Centroids); c) Actual vs. Relative Error; d) Sample Output Predicted vs. Actual

System Optimization

With an idea of what membership functions should be used and the effects of different database sizes, the final step was to optimize the FRBS to the data. This process is also often called tuning, and involves some combination of training (and possibly re-training) and shaping and shifting the membership functions to provide the best system design [2, 28, 81].

The optimization methods explored required a parameterization of the location and shape of input and output membership functions to systematically adjust their shape in the system. An example of the parameterization used for gaussian and triangular membership functions is shown in Figure 52. The parameter set a_i, b_i, c_i defines each parameter in the membership function, and is constrained such that the functions cover the entire domain, overlap one another some minimum portion of the domain, and are of a reasonable width. Figure 53 illustrates an example of the baseline gaussian membership functions used, as well as the membership functions defined at their minimum and maximum possible parameters.

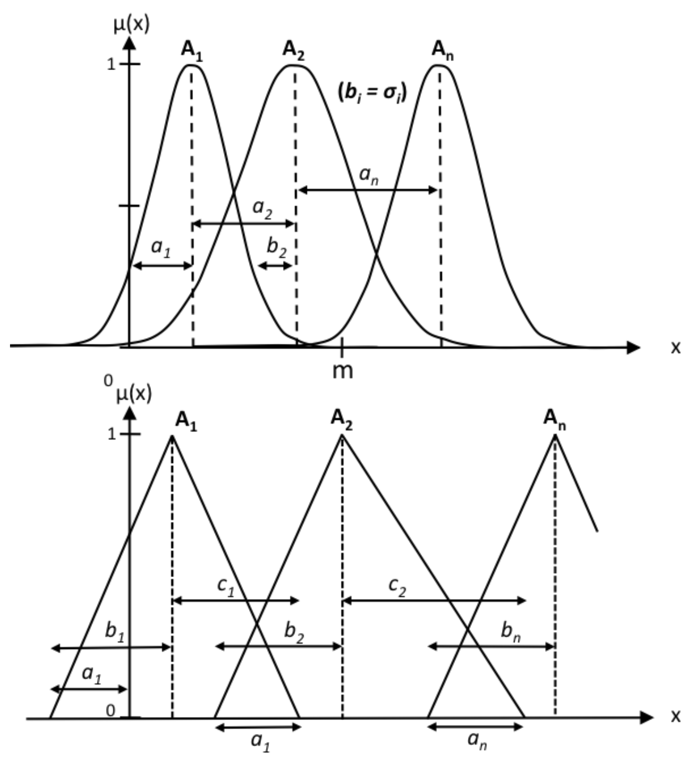


Figure 52: Parameterization of Gaussian and Triangular Membership Functions for FRBS Optimization

Three methods were briefly explored for FRBS optimization, a simple genetic algorithm, the differential evolution algorithm, and a gradient based method, limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS-B). L-BFGS-B is a version of the BFGS quasi-newton method that uses a limited amount of memory and has been adjusted to use simple box constraints, popular in machine learning [142]. The

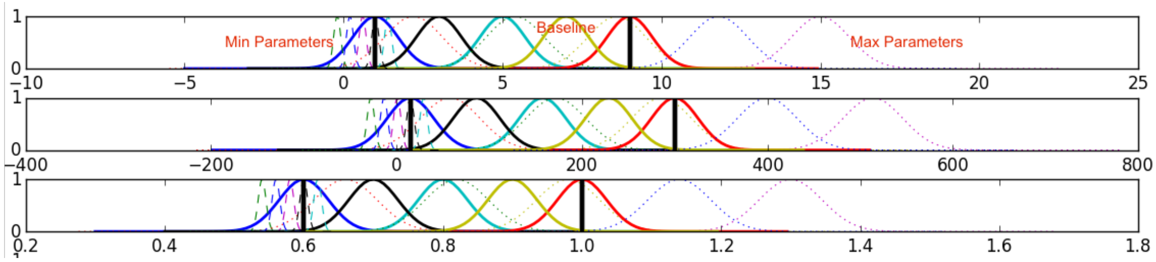


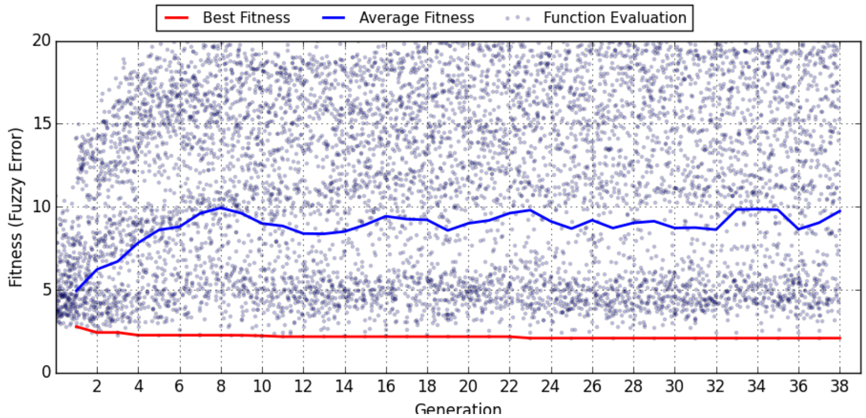
Figure 53: Example of Bounds on Membership Functions for FRBS Optimization

systems were again each trained with the same 250 data points and validated against the same 150 as previously, with their fitness or objective function defined as the fuzzy system MSE across the validation points.

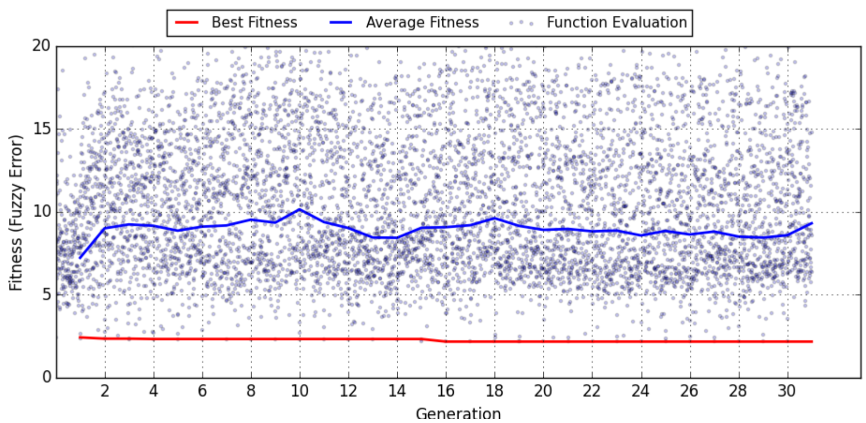
The L-BFGS-B algorithm immediately proved to require too many function (system) evaluations to calculate the Hessian to work in a timely manner. L-BFGS-B has a rough computational cost of just $O(mn)$ evaluations, where m is the maximum number of variable metric corrections used to define the limited memory matrix and typically $m \in [5, 30]$. With 168 parameters ($2 \times 5\,MFs \times 15\,inputs + 2 \times 9MFs = 168\,parameters$), this meant 800+ evaluations of the system per iteration of the algorithm.

The genetic and differential evolution algorithms both proved to work fairly well in reaching some optimum set of parameters, although a global optimum was not guaranteed. With populations of between 150-250, each algorithm typically took between 15-40 generations to arrive at a converged solution. Because of it's slightly quicker and better performance in selecting inputs, the genetic algorithm was selected for use in optimizing the FRBS system. Figure 54 illustrates the optimization process of the selected Empty Weight fuzzy system model with differential evolution and a genetic algorithm. The convergence criteria for the genetic algorithm checks the number of consecutive generations without a new best candidate (15 generations were required).

The final optimized FRBS was validated against the full 300 points and had a



(a)



(b)

Figure 54: Optimization of Empty Weight Ratio FRBS with a) Differential Evolution b) Genetic Algorithm

fuzzy MSE of 2.190 (compared to 2.548 for the unoptimized system). The system had also increased to 223 rules from the 177 rules in the unoptimized baseline. The tuned membership functions are visualized in Figure 55. The same goodness of fit checks were generated for the optimized system, and are shown in Figure 56. While far from a perfect fit, the checks illustrate a significant improvement over the baseline trained system. The full system, as defined by its Fuzzy Control Language file, is shown in Appendix C.

For engineers and scientists taught in the methods of statistical regression, the fit appears poor, especially when compared to closely related method of using response surface methods to model physics based multi-disciplinary design and analysis tools.

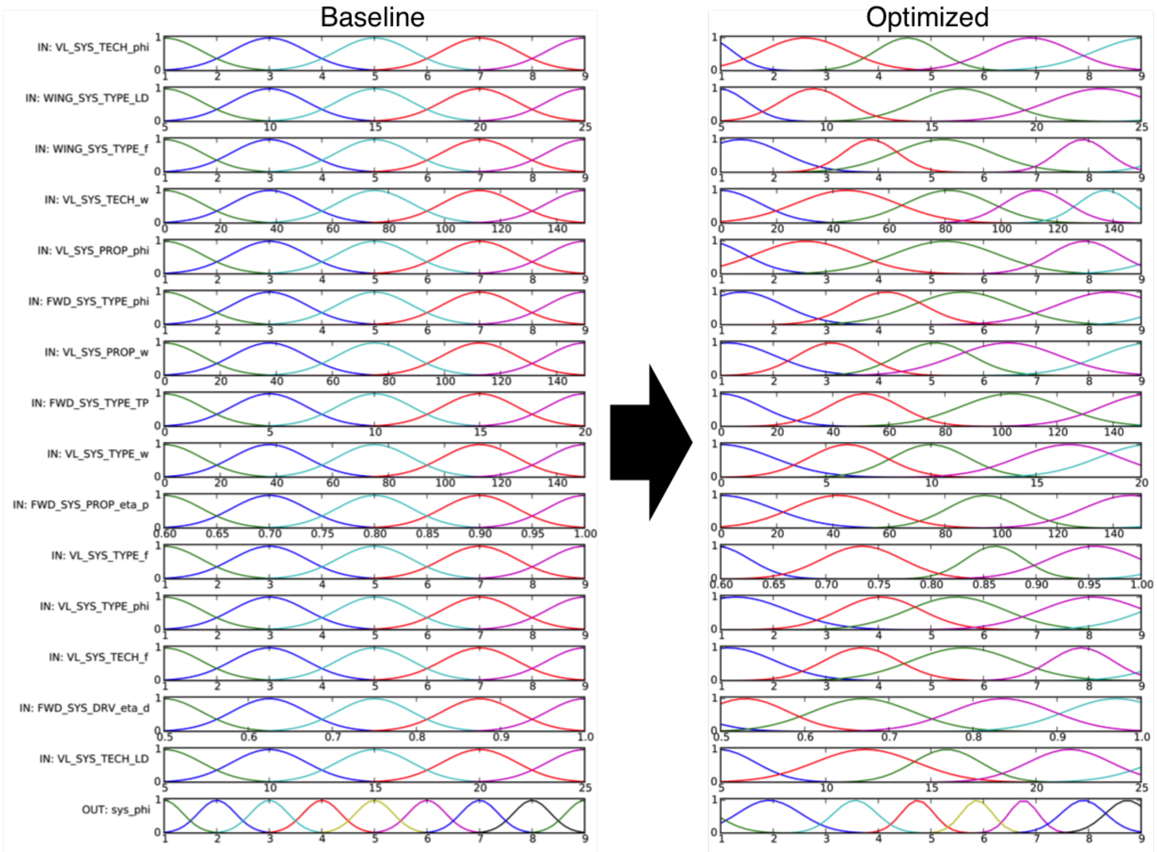


Figure 55: Optimization of Empty Weight Ratio FRBS: Membership Functions

However it is important to consider that this exercise is distinctly different, and the purpose to create a consistent means of interpreting (often inconsistent) expert data, and allow for the quick generation of an expert-based evaluation of criterion performance in a very uncertain environment. This rule-based system illustrates a systematic, moderately interpretable, and logical means of estimating the qualitative empty weight of system based on it's architectural makeup, where it would be far too costly to constantly consult a real team of experts over a large ($\geq$ 1000) number of alternatives.

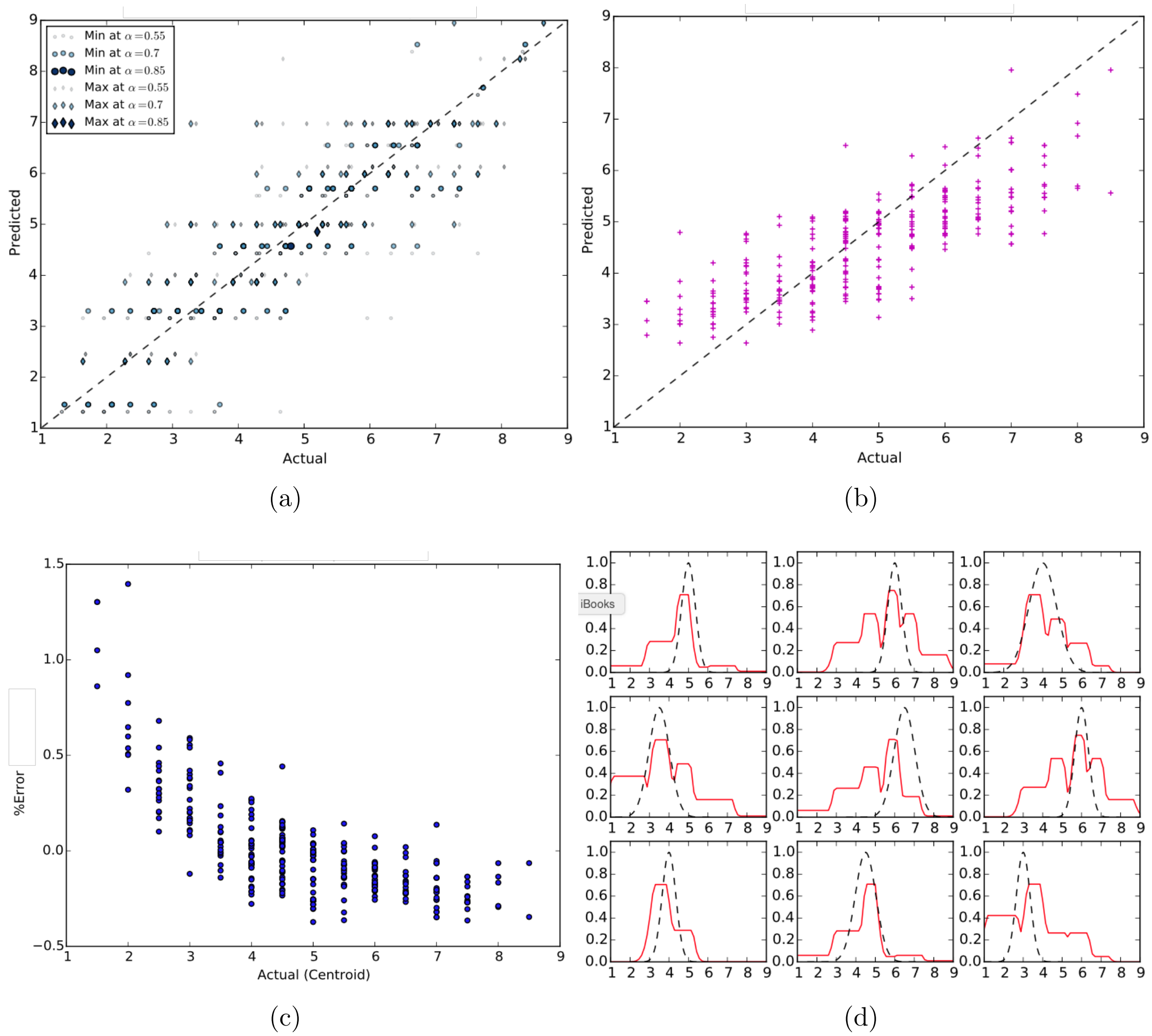


Figure 56: Goodness of Fit Checks for Optimized Empty Weight FRBS: a) Actual vs. Predicted for Various α-cuts; b) Actual vs. Predicted (Centroids); c) Actual vs. Relative Error; d) Sample Output Predicted vs. Actual

3.3.5 Creation and Training of FRBSs from Physics-Based Data

In addition to creating a FRBS to model expert elected data, an experiment was performed to attempt to model physics-based data. The data generated for Figure of Merit was used to understand how well a FRBS might model physics-based data. The lessons learned from modeling expert elicited data for empty weight ratio were applied to the modeling of an FRBS system for Figure of Merit.

The inputs of the system were already mostly known from the generation of the data. Disk loading for the vertical lift system was calculated as the average minimum and maximum input values for disk loading for each of the vertical lift system type, propulsor, and technology aspects of the system. The input data were modeled as gaussian input functions, and a system with 5 gaussian membership functions per input and 9 gaussian membership functions for the output was created. The system architecture is illustrated in Figure 57.

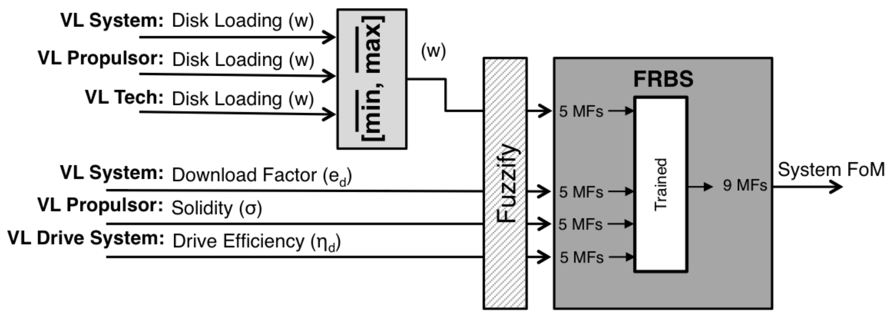


Figure 57: System Architecture: FRBS Figure of Merit

The system was trained with 400 data points and optimized using an additional 100 points for validation. The final system has 223 (out of 625 possible) rules, and had an overall fuzzy MSE of 0.0044, or a fuzzy RSME of 0.067. The goodness of fit tests were applied to the resulting optimized system as shown in Figure 58. With more consistent data the FRBS generation procedure results in a much better fit, but is limited due to the discrete nature of the available membership functions in the output domain.

The Figure of Merit data could easily be fitted using a response surface equation with very good accuracy. However the FRBS approach here provides nearly instantaneous output of a full membership functions, while nested Monte Carlo or some other sampling method would have to utilized to produce a full distribution for the outputs of a particular system given both distributions on the noise and input paramaters in a statistical approach. The FRBS also fits with the rest of the framework providing

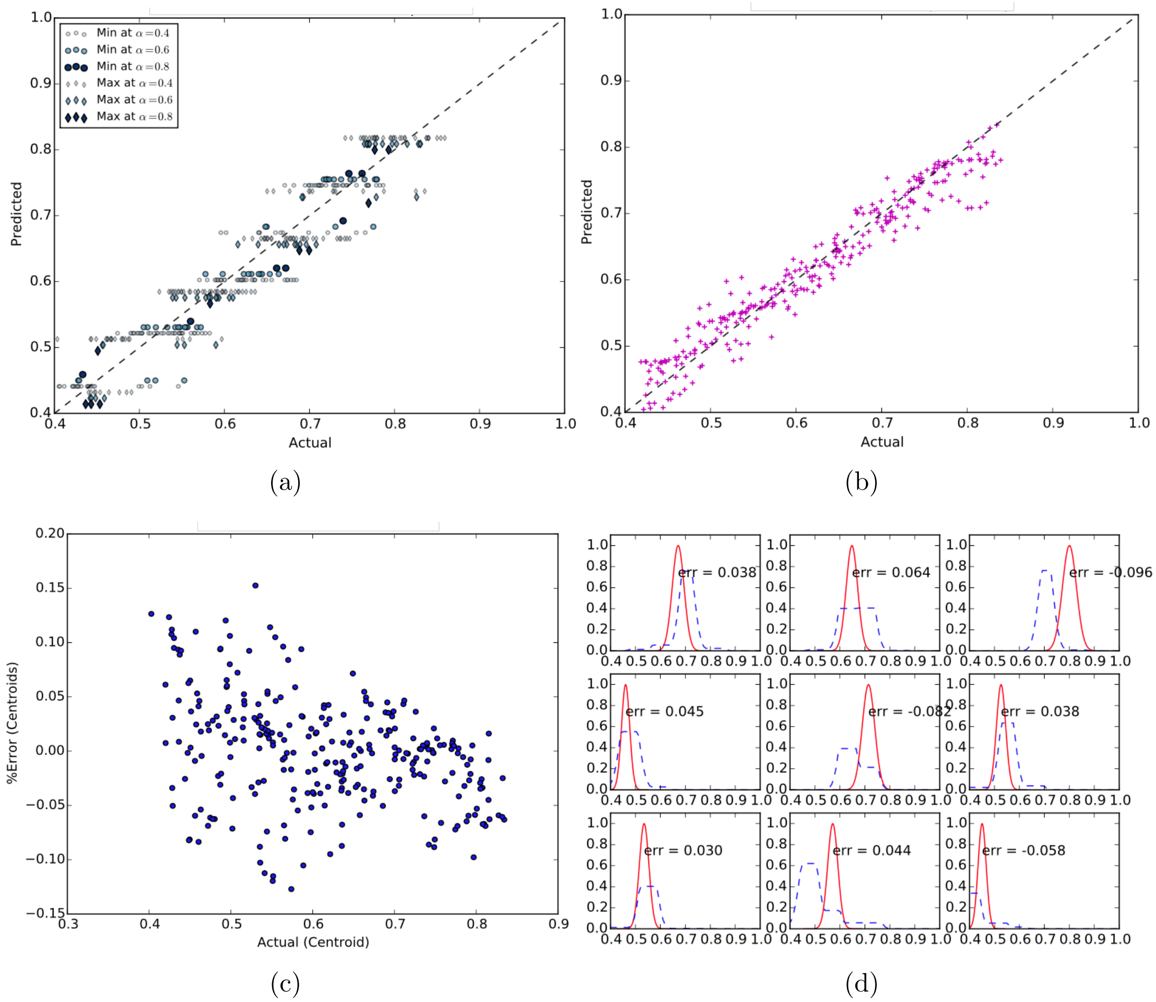


Figure 58: Goodness of Fit Checks for Optimized Figure of Merit FRBS: a) Actual vs. Predicted for Various α-cuts; b) Actual vs. Predicted (Centroids); c) Actual vs. Relative Error; d) Sample Output Predicted vs. Actual

consistency in approach. At this stage in the design, when a rough, lower fidelity approach is feasible, an FRBS model of physical data is a quick, appropriate approach to modeling a system attribute under uncertainty. The Fuzzy Control Language file for the resulting system is shown in Appendix C.

3.3.6 Neuro-Fuzzy Systems (NFS)

The number of individual approaches to neuro-fuzzy systems (NFS) proposed in literature are staggering, and the terminology is quite varied. Approaches include neural networks with fuzzy neurons or parameters, networks that mimic fuzzy systems but allow for neural network like training, crisp networks that represent fuzzification, fuzzy parameters, or fuzzy inference systems, and more [1, 21, 67]. NFSs are particularly popular for applications that require controllers for complex but "fuzzy" systems, where expert knowledge can be used to guide the controller. However, the systems explored for this research were those proposed to model expert opinions, and capable of accepting fuzzy inputs and producing fuzzy outputs. Once again, reducing the uncertainty implicit in the nature of the concept selection problem to a single crisp evaluation number was not desired.

The purpose of this research is not to determine a definitive technique for the use of NFSs in the larger concept exploration framework, but rather to show how they might useful in the context of the larger purpose of modeling vague and uncertain expert and physical data. For this reason several NFS approaches were explored, several examples of which are shown in Figure 59. Based on this preliminary research, two architectures were selected for further exploration, the NEFPROX and Discrete Fuzzy Expert System (DFES) systems.

NEFPROX System

The NEFPROX is a NFS based on rule-based Mamdani systems proposed by Nauck and Kruse for function approximation [79]. The NEFPROX system is a simple 3-layer generic fuzzy perceptron, with neurons arranged in "input", "rule", and "output" layers. The first layer acts a pass through, gathering the appropriate inputs for the system and passing them to each rule. The rule layer of the system has an activation function that approximates determining the minimum firing strength for that rule. The input weights to the rule neurons are the appropriate membership functions from each corresponding input to that rule. The output neuron (a single neuron for each output) then has input weights corresponding to the output membership

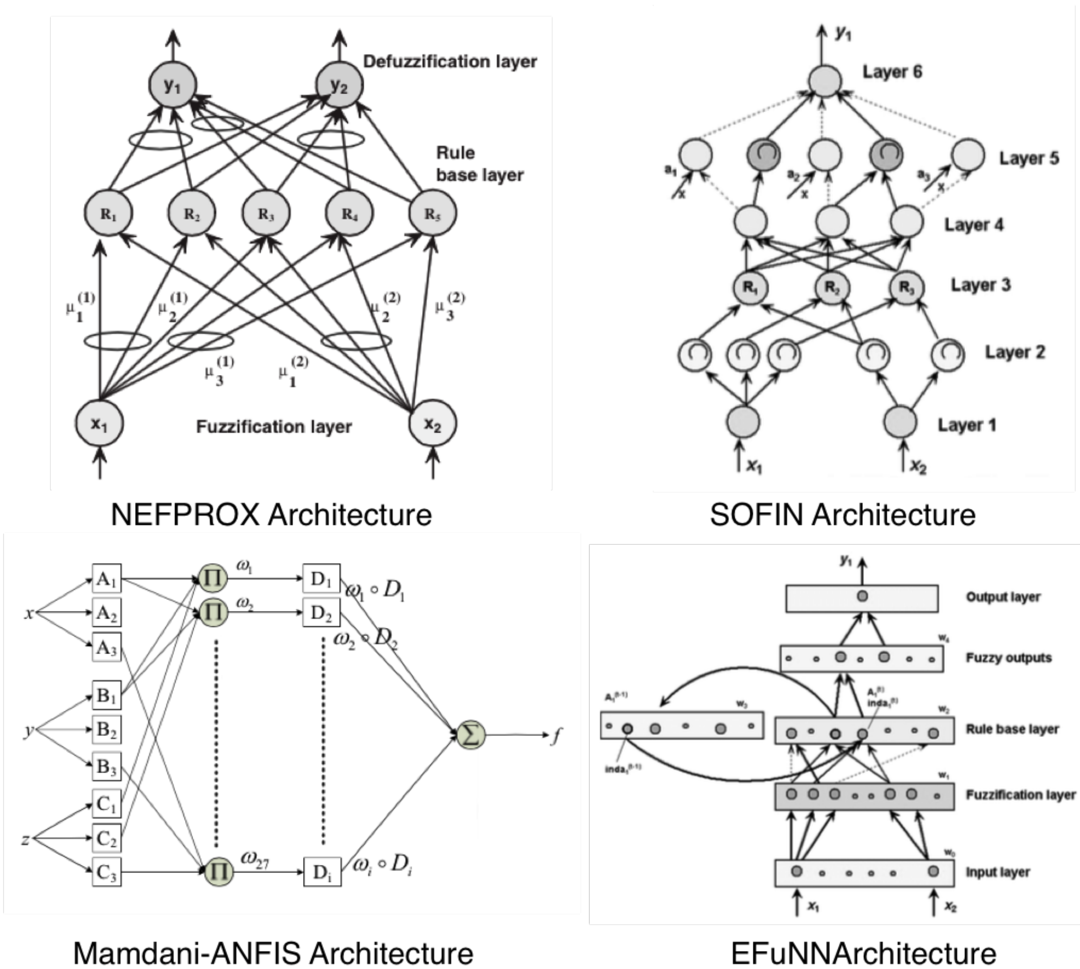


Figure 59: Examples of Neuro Fuzzy Architectures: NEFPROX [79], SOFIN [1], M-ANFIS [21], EFuNN [1]

functions associated with each rule. This neuron performs implication associated with each rule and aggregates the outputs into a single fuzzy membership function. The output can then be defuzzified as necessary. Each of the membership function weights are grouped, so that during training changes to one weight affect all the weights that correspond to the same membership function so the system can continue to be essentially interpretable as a FRBS. In this way the system attempts to balance interpretability and accuracy. A full mathematical description of the system can be found in [79], while an example of NEFPROX architecture can be seen in Figure 60.

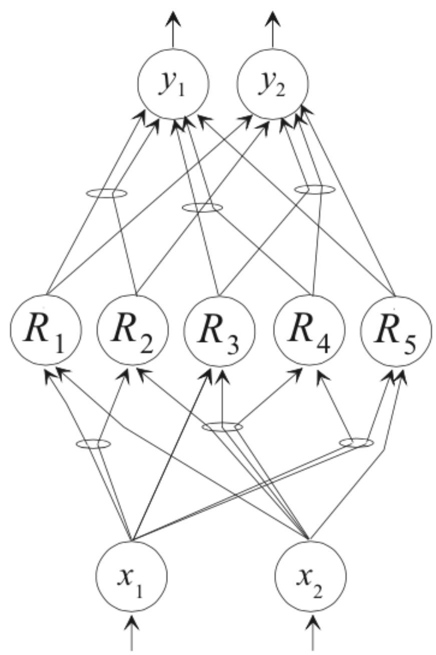


Figure 60: Example of NEFPROX Architecture [79]

The NEFPROX system was trained through the use of the algorithm described in [79]. This takes place in two phases. In the first phase, "structure learning", the system uses the training data to determine the appropriate rule base in much the manner that the FRBS training was accomplished. Each data point is reduced to a rule based on the input and output membership functions with the strongest degree of memberships. To determine the rule base, if two rules have the same antecedent, the rule with the output membership function that has the highest mean degree of membership across the data is used. Once the rules are determined, and the rule and output neurons and their connection weights have been updated, the next phase of learning takes place. During "parameter learning", the connection weights in the form of fuzzy membership functions are adjusted through gradient-based backpropagation. Every time a membership function for one rule or output neuron is adjusted, all the other membership functions associated with that input or output and linguistic term are adjusted as well so they remain the same. After each pass through all of the training data, a separate set of truth data was checked for validation in an attempt to prevent over-training. The cumulative system error against the validation set is checked and then the entire training and validation process repeated.

Implementation of the NEFPROX system resulted in several primary issues. The first being that the system is not designed to be operated with fuzzy inputs and outputs. While modification of the system to produce fuzzy outputs is simple, the back propagation algorithm is contingent on being able to calculate a direct error between the output and truth data. Using a more complex fuzzy error measure is possible, but the error measure must be directional (i.e. indicate a positive/negative direction as well as the error magnitude), and adds significant mathematical complexity to the backpropagation algorithm. This complexity turned out to be a significant barrier in terms of the computational time necessary to train the system. Moreover, the nature of the system meant that it's capability was virtually identical to a Mamdani FRBS (albeit providing a means for training). This meant that no significant gains in accuracy would be seen between a NEFPROX and a FRBS system, other than those made by differences in training.

Discrete Fuzzy Expert System (DFES)

After early trials of the NEFPROX system failed to produce meaningful results, another NFS type that might fit the problem description was sought. Reviewing additional literature, a simple NFS type was identified that was based entirely on flexible fuzzy inputs and outputs. A Discrete Fuzzy Expert System (DFES) is a NFS that uses crisp neural networks to approximate discretized fuzzy membership functions representing a fuzzy expert system [48, 43]. In this relatively simple system, the entirety of the fuzzy inference system is replaced by a neural network which represents a continuous mapping between the fuzzy inputs and fuzzy outputs. The simple, feedforward neural network approximates a fuzzy inference system where all rules fire simultaneously, without thresholding or uncertainty, for each given new input set.

The entire domain on some interval, $[a_i, b_i]$ for each input (i) is approximated across some n discretized points, $[x_{i,1}, ...x_{i,n}]$, such that $a_i \leq x_{i,j} \leq b_i$ for $j \in 1, ..., n$. The neural network then has input nodes corresponding to each discretized point, $x_{i,j}$, plus one bias input node, $x_{i,(j+1)} = 1$, for each input. Allowing for some appropriate number of hidden nodes, the output nodes of the system correspond to a similar discretization of the domain for the output (more than one output can be used, but is not in this case), $[y_1, ...y_m]$, such that $a \leq y_j \leq b$ for $j \in 1, ..., m$. Given new input data, each input node assumes the degree of membership of its corresponding input value, $\mu(x_{i,j})$. Once the neural network has completed feed forward, the output node values represent the degree of membership of each corresponding discretized output value, $\mu(y_j)$. In this way, the inputs and outputs of the neural network are continuous on $[0, 1]$. Figure 61 illustrates a simple version of the system architecture with two inputs and one output.

One notable issue with DFES systems was occasionally seen if the granularity of any of the input domains was too low. If singleton or significantly dense functions are

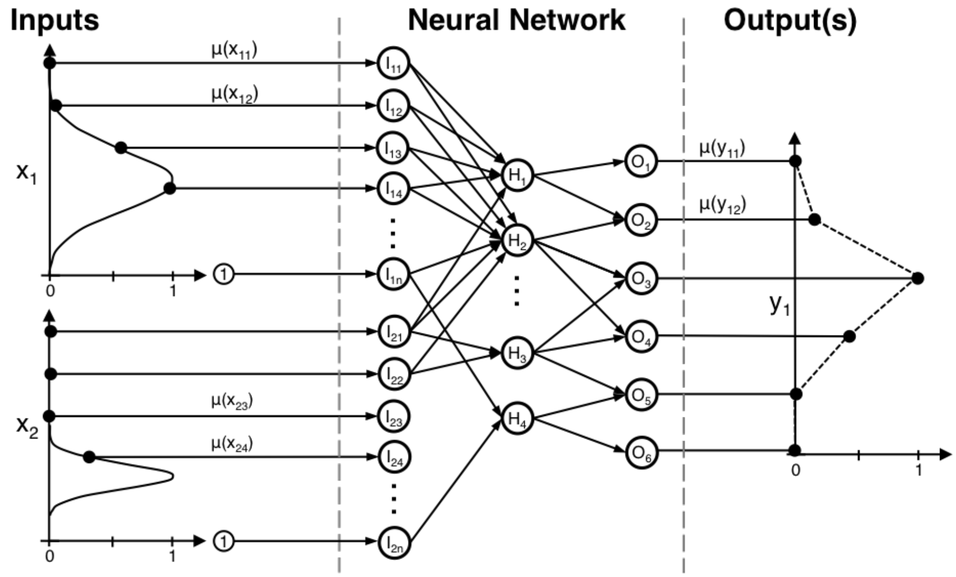


Figure 61: Discrete Fuzzy Expert System (DFES) Representation [48]

used to train the system, it is possible that occasionally no input neurons would fire at a significant value. A simple catch was added in the DFES feed forward function to check if no input neurons had fired at a value above some threshold ($\mu(x) = 0.4$), then the maximum input membership function value was determined and assigned to the input node nearest to the value in the input domain that the maximum membership was found at. Conversely, if the granularity of the input space was too high, and there were a number of inputs, the required neural network could become unmanageably large, resulting in very high training times. Thus, it was necessary to strike a balance in the input and output discretization.

Any relevant method can be selected to train the neural network, but for this research a simple back propagation algorithm was utilized. To train the DFES for a given data set is used with both fuzzy inputs and fuzzy outputs, in this case representing expert evaluations or generated, fuzzified data for both the inputs and outputs. Both data based on set linguistic terms and/or raw data can be used in training. Some portion of data is deemed "holdback" for validation, to prevent over-training. The full training algorithm used follows the following steps:

1 - Randomize Training Data: The training data set order is randomized to ensure a more uniform fit.

2 - Train with Backpropagation Data: At each data point, the system is run using the fuzzy inputs, and the error is calculated at each discrete point (node) in the output space. The error is then back propagated through the system, adjusting each neuron's weight to better fit the data.

3 - Test System: The system error against the holdback (validation) data is checked by calculating the root mean squared error (RMSE) of the output nodes of each data point and summing them.

4 - Convergence Check: The standard deviation of the total error (training + validation) is calculated over the last 10 iterations through the data. If it is not low enough, the process repeats from Step 1. If this falls below some threshold the training is stopped. The system with the lowest validation error is selected as the final system.

The effectiveness of this training method is discussed further below as a part of the development of specific DFES systems for Hover Figure of Merit and Empty Weight Ratio. An example of the error progressions can be seen in Figure 63, with each iteration being one trip through steps 1-4.

3.3.7 Development of a DFES System for Physics-Based Data

The primary criterion used here to illustrate the development of a DFES is Figure of Merit, with training data generated using Momentum Theory as outlined in Section 3.2.4. Later the training of a DFES to expert elicited data for Empty Weight Ratio is also discussed. The training process for a DFES is much more straight forward than that of a FRBS, with less real options to understand. The shape of the membership functions used to model input and output data, must be determined, as well as a reasonable granularity of the input and output spaces and number of hidden nodes.

The amount of training data used turned out to be less critical of a factor, as the use of holdback data prevented the overtraining of the driving neural network. Moreover, training data could be quickly produced with the simple algorithm used, and training was relatively quick for smaller DFES systems, even for larger training data sets (> 400 points).

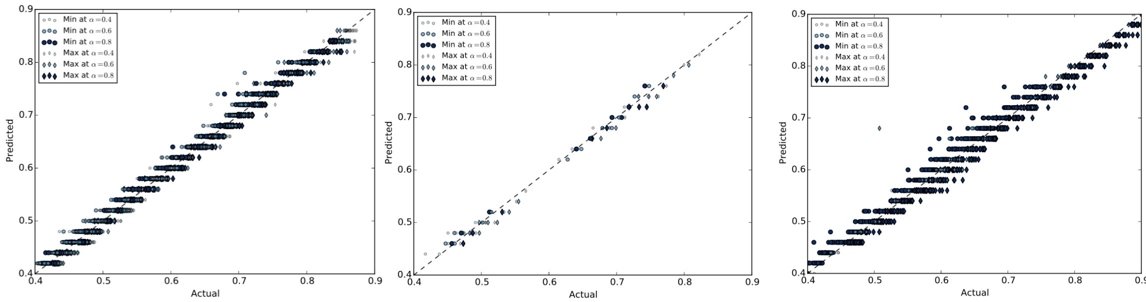
To compare the effects of modeling the data with various membership functions type, a preliminary system was created with an input and output granularity of 30 nodes per variable and 100 hidden nodes. This system was trained, modeling the data as gaussian, triangular, and trapezoidal membership functions. Because the inputs for the Figure of Merit training data are crisp values, the inputs are modeled as very narrow fuzzy gaussian numbers (to encourage triggering a input node) while the output data was modeled with the desired function type.

Each system was validated against a separate 500 data points, with the results shown in Table 18. Here the mean discrete RMSE across all the data points is shown, as well as the mean squared fuzzy error (calculated using the α-cut based distance measure previously used). Figure 62 also shows actual versus predicted plots at 3 α-cuts for each data type.

Table 18: Comparing Data Modeling Function Types for DFES Figure of Merit Systems

Output Data Function	Total Discrete RMSE	Fuzzy MSE	Avg. Run Time (s/Node)
Gaussian	0.04578	0.000132	4.033e-05 (0.010s)
Triangular	0.30744	0.026351	4.844e-05 (0.012s)
Trapezoidal	0.08503	0.000316	5.031e-05 (0.012s)

While both the gaussian and trapezoidal function types do a fairly good job of modeling the data, the gaussian membership function proves to be best for creating an accurate system. Similarly to the FRBS, accuracy of modeling results was deemed to be the most important factor for data modeling, as each function was interpretable in its own way. Using the gaussian functions for data modeling, the next experiment



(a) (b) (c)

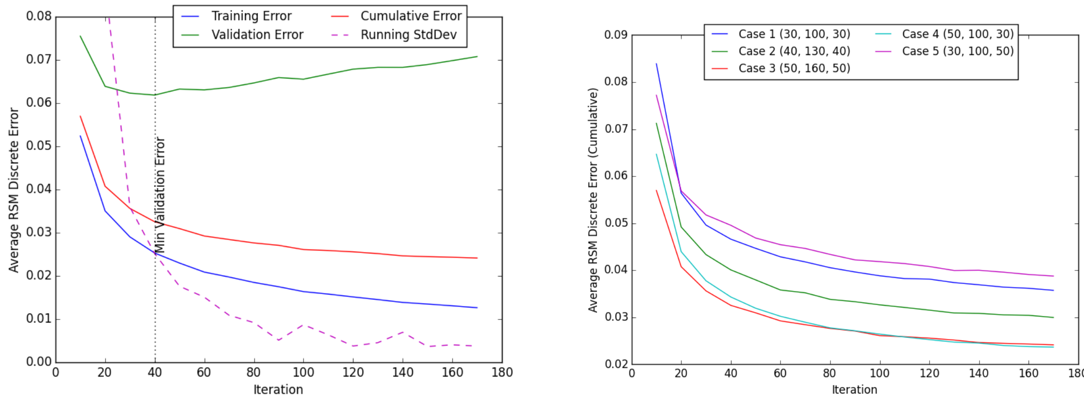
Figure 62: Actual vs. Predicted for Various α-cuts (30 nodes/input, 100 hidden nodes, 30 nodes/output): a) Gaussian Data; b) Triangular Data; c) Trapezoidal Data

run varied the number of input and output nodes to understand how discretizing the systems affected its accuracy and run time. Figure 63 illustrates the training progression of the selected system (50-100-50), as well as the total error during the training progression of each system. Each iteration is one pass through steps 1-4 of the outlined training process.

Table 19: Training Results for DFES Figure of Merit Systems

Nodes/ Input	Hidden Nodes	Nodes/ Output	Mean Discrete RMSE	Fuzzy MSE	Avg. Run Time (s/Node)
30	100	30	0.04578	0.000132	4.033e-05 (0.010s)
30	100	40	0.04918	0.000171	4.234e-05 (0.011s)
30	120	50	0.04588	0.000168	4.704e-05 (0.014s)
40	130	40	0.03517	0.000099	5.171e-05 (0.017s)
50	160	30	0.03189	0.000161	6.073e-05 (0.024s)
50	160	50	0.03147	0.000058	6.336e-05 (0.026s)

It is apparent from the results that the increases of input granularity better serve the system discrete accuracy than the increases in output granularity. However to make signifiant gains in reducing the fuzzy error of the outputs, additional output granularity is required. Compared to the time scales of the FRBS systems (0.5-1.0s), the effects on execution time of adding addition nodes for increased accuracy is negligible. For this reason, the system with 50 nodes per parameter in both inputs



(a)

(b)

Figure 63: Training of a DFES for Figure of Merit: a) Example System Training Progress; b) Cumulative Error During Training for all Cases (Nodes/Input, Hidden Nodes, Nodes/Output)

an output was selected for the final system.

In Section 3.3.5, a FRBS system was created to model Figure of Merit, and the final optimized system had a fuzzy MSE of 0.0044. The same system fuzzy MSE for the DFES is 0.000058, showing a fit that is magnitudes of order better. To further compare the fit to the FRBS system for Hover Figure of Merit, the same goodness of fit checks were used for this DFES system, and are shown in Figure 65. It is immediately clear that the DFES provides a much more accurate model of the data. With no discrete set of membership functions to combine for the output, the DFES tracks the continuous nature of the data much more smoothly across the criterion space.

3.3.8 Development of a DFES System for Expert Elicited Data

With the increase in accuracy provided by the DFES for Figure of Merit data, a DFES model of Empty Weight Ratio was also explored. A system was created with 30 nodes/input, 200 hidden nodes, and 40 nodes/output. The system was trained with the same 15 inputs identified through optimization of the FRBS, under the assumption that regardless of the model type that these inputs best reflect the variability in the

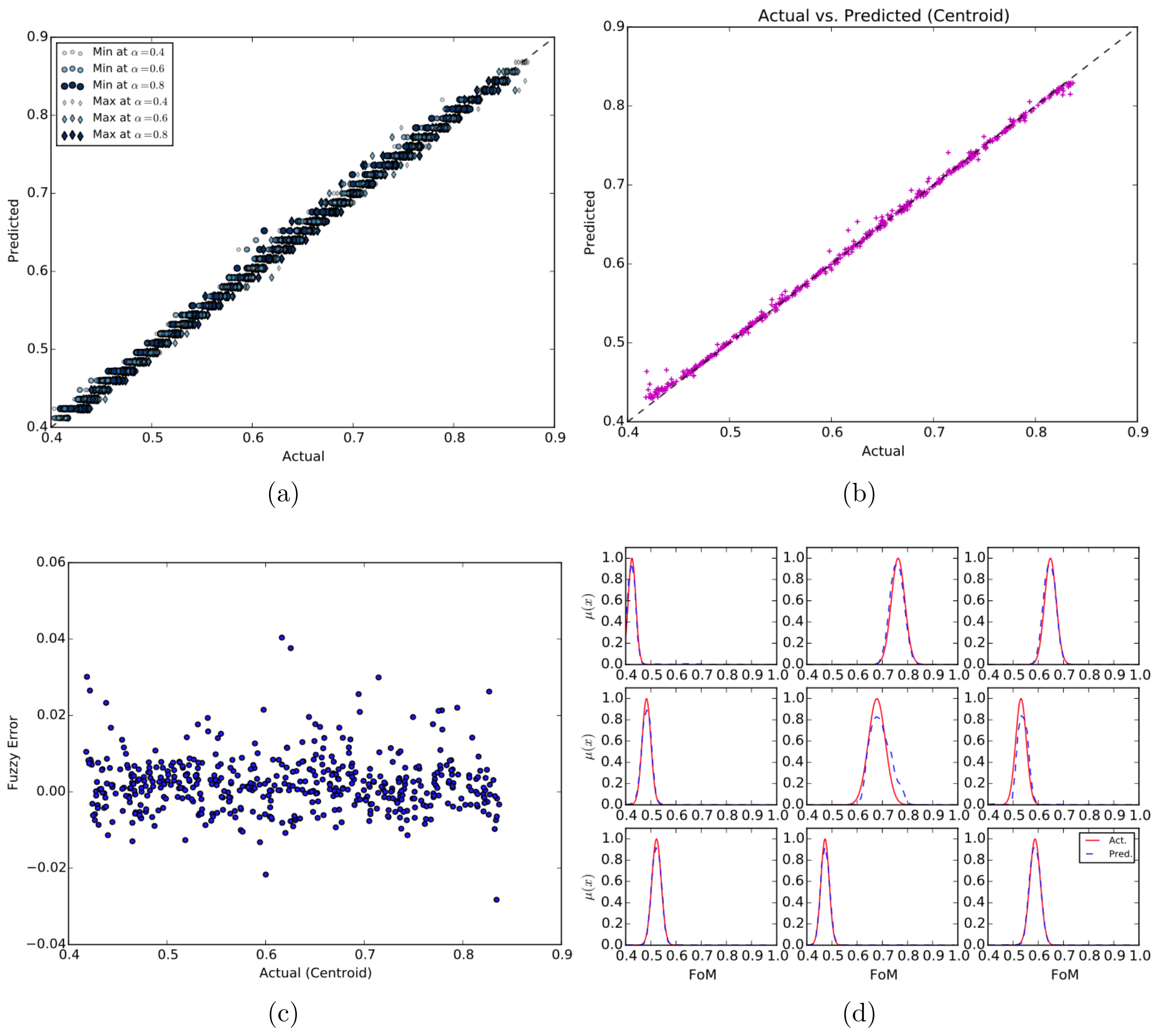


Figure 64: Goodness of Fit Checks for DFES Model of Figure of Merit (50 nodes/input, 160 hidden nodes, 50 nodes/output): a) Actual vs. Predicted for Various α-cuts; b) Actual vs. Predicted (Centroids); c) Actual vs. Fuzzy Error; d) Sample Output Predicted vs. Actual

output, and would be the best input set to create an accurate model. At the least, if the DFES model followed the trend of the Figure of Merit model, moderate gains in accuracy could be found, and the system could be improved from there. The system was trained with the available 300 data points and a 15% holdback fraction. Gaussian functions were used to model the data.

Unfortunately, the DFES proved to perform very poorly in this case. The trained system had a validated fuzzy MSE of 3.627 (significantly worse the 2.1 of the FRBS)

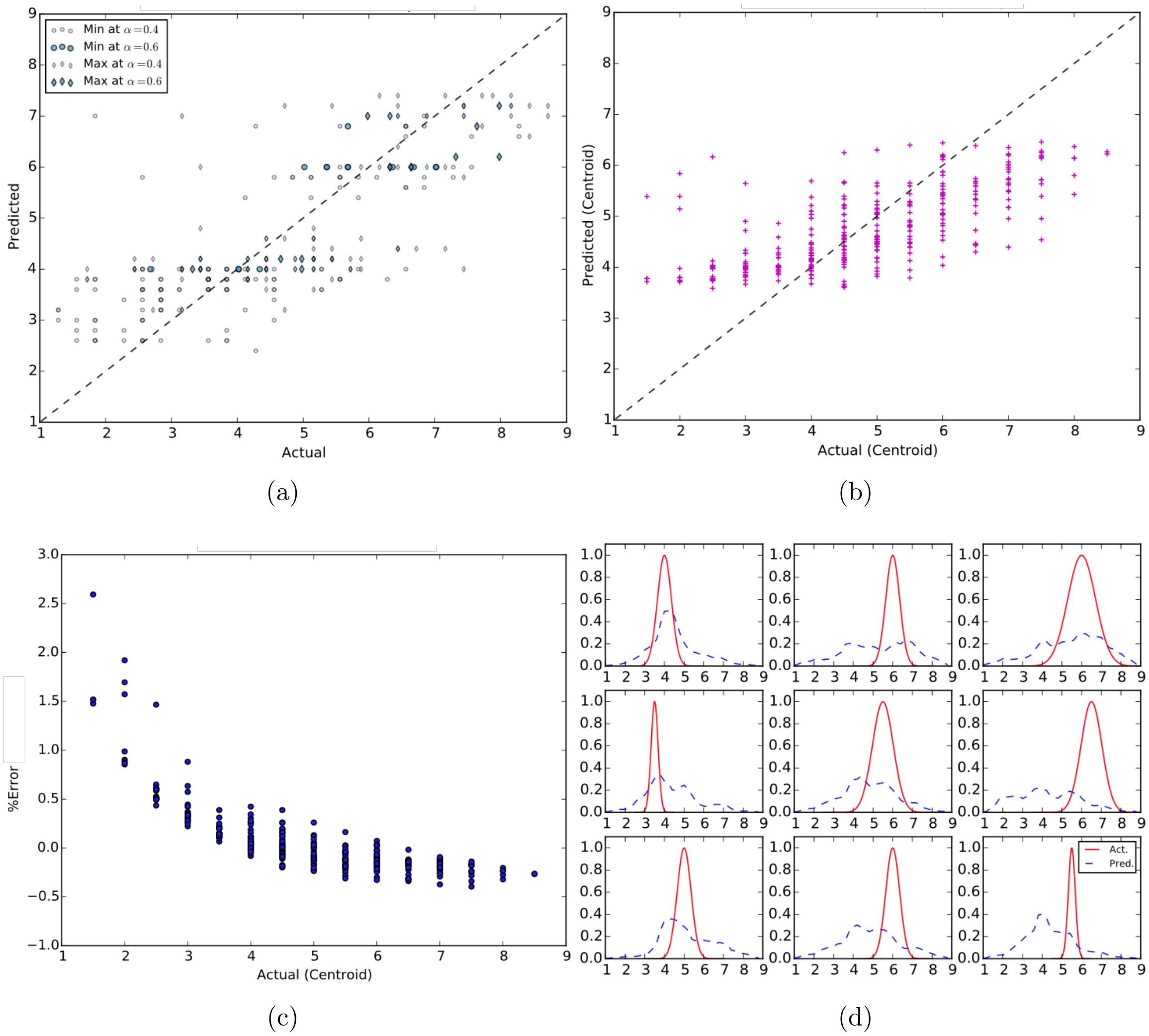


Figure 65: Goodness of Fit Checks for DFES Model of Empty Weight Ratio (30 nodes/input, 200 hidden nodes, 40 nodes/output): a) Actual vs. Predicted for Various α-cuts; b) Actual vs. Predicted (Centroids); c) Actual vs. Fuzzy Error; d) Sample Output Predicted vs. Actual

and a mean discrete error 0.237 across all 300 points. Result of the goodness of fit tests are shown in Figure 65. It is clear that this is not an improvement over the FRBS system. The centering is much more noticeable, and errors are very high. The example points show that each data point results in a membership function that has low values across most of the domain. It is speculated that as the expert data has some inconsistencies (demonstrated by the FRBS), the back propagation training tries to adjust for all the errors, rather than training to the strongest relationships, the results often end up being very uncertain ($\mu < 0.4$) over large ranges of the output space to try and match all the data. These "mushy" results do not result in terrible overall error, but at the same time they do not provide a meaningful model.

3.3.9 Additional Framework Fuzzy Models

In order to be able to model all of the system attributes desired and create the full framework, several additional fuzzy models were created. The development of these systems is not covered in detail in this report, but the lessons from developing the previous systems were applied in an attempt to create reasonably accurate, logical, and fast models.

A simple FRBS system was created to model system propulsive efficiency. This system uses the intersection of the scored ranges of propulsive efficiencies for forward propulsor and forward system type, as well as the forward system drive efficiency. Ten rules were used to generate an output for the system , with the full fuzzy system generation file shown in Appendix C. The resulting system was used to model the 10 baseline alternatives from the benchmark, and the resulting α-cuts at $\max(\mu_X)$ are shown in Figure 66 along with the benchmark expert scores.

The data collected using the R_F method from Section 3.2.4 was used to create three fuzzy DFES models: gross weight, power installed, and maximum airspeed. The same inputs were used for each system and are those listed as main parameters in Table 13. In the final framework, some of these inputs (empty weight ratio, figure of merit, and propulsive efficiency) were drawn from other fuzzy models rather than an input specified by experts based on the selected options. Each of the systems used an input granularity of 40 nodes/input, an output granularity of 50 nodes/input, and 250 hidden nodes. The systems were trained using 600 data points with a 0.1 holdback fraction. The resulting system fit errors are listed in Table 20, and the same four goodness of fit checks were used for each system to confirm. Rather than show

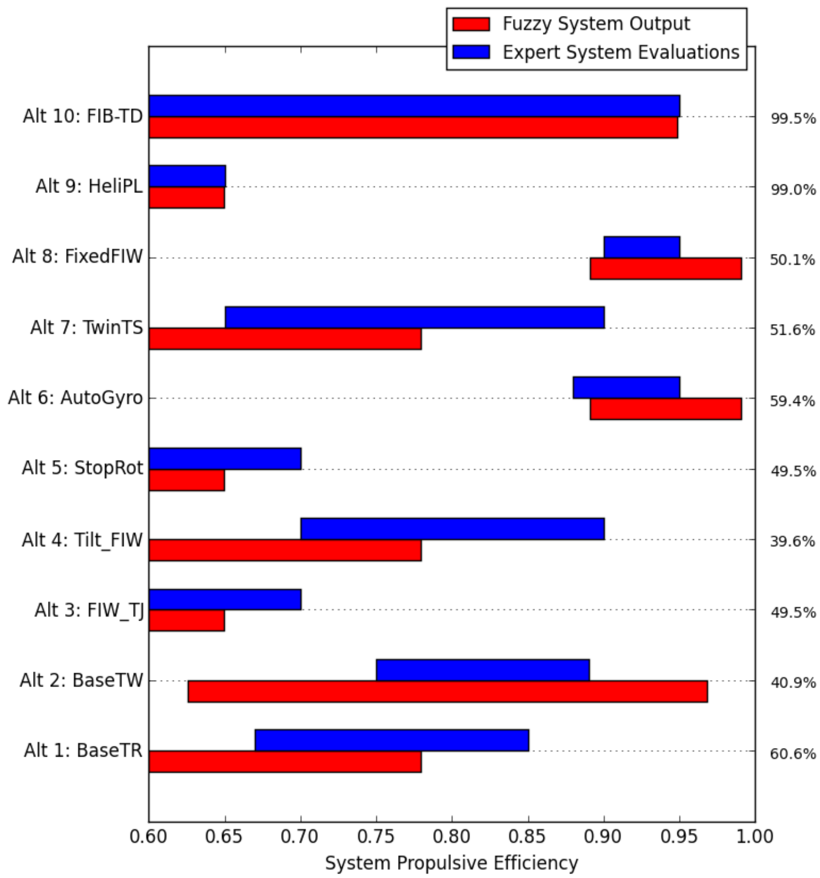
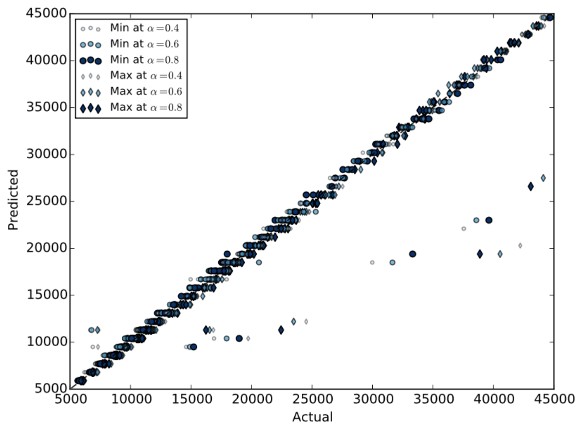


Figure 66: Performance of Propulsive System for Baseline Alternatives (α-cut at $\max(\mu)$), with % Overlap Shown.

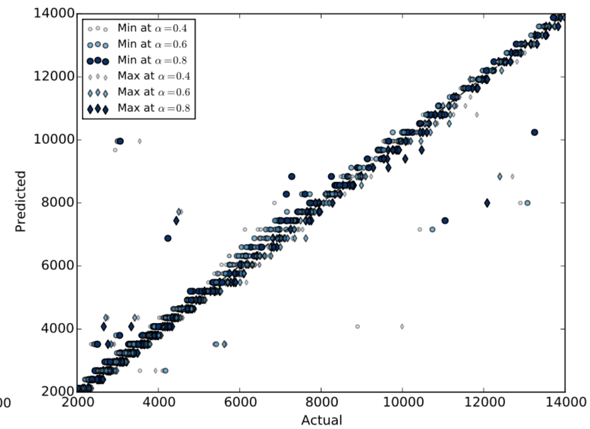
four pages of fit check plots, just the actual versus predicted plots for various α-cuts are shown in Figure 67.

Table 20: Fit Statistics for DFES R_F Systems

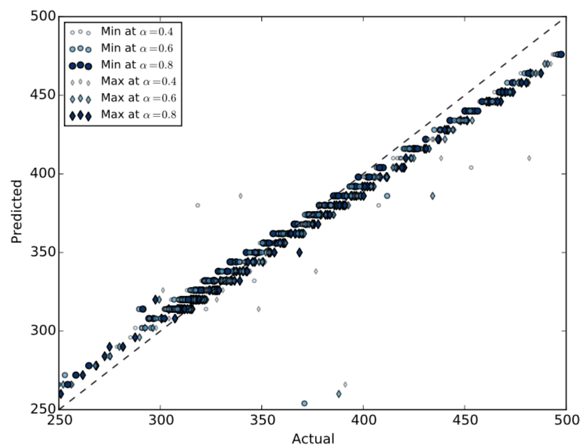
System	Mean Discrete RMSE	Fuzzy MSE
Gross Weight	0.03130	4878.4
Installed Power	0.05095	3147.7
Maximum Airspeed	0.12532	79.7



(a)



(b)



(c)

Figure 67: Actual vs. Predicted for Various α-cuts Fit Checks for DFES Models of R_F tool a) GWT; b) Power Installed; c) Maximum Airspeed

The R_F DFES system fits were deemed to be good enough to continue developing the framework, but are clearly not as good as the fits for Figure of Merit. This is likely due to the large increase in complexity from momentum theory to the R_F

method. The R_F method is producing data that is non-linear, and potentially not continuous, due to the complexity of logic and mixing non-linear models within the algorithm. It is likely that better fits could be obtained through extended analysis of DFES modeling and potentially including much larger models. There is also a consideration to be made with respect to the time spent in making the models versus the accuracy required at this stage in design. It is important to consider when the model provides a close enough result for this type of screening framework. For the purposes of example in this research, and generating a framework that is capable of first-order estimates in the face of a great deal of uncertainty, the models were good enough to continue on to the next phases.

3.3.10 Discussion

The example fuzzy systems developed here were designed to serve essentially as surrogate models for expert opinion and physics-based data in a larger framework for architecture evaluation and identification. These experiments attempt to answer Research Question 2: "*How can Fuzzy System(s) be used to evaluate architecture alternatives early in conceptual design?*". The results are positive but mixed, and also reflect the answers to Research Question 3: "*How should expert data be elicited to build the Fuzzy Expert System(s)?*" The research attempted to identify a means to create fuzzy expert models based on simple expert rules (Lift-to-Drag Ratio), as well as expert evaluations (Empty Weight Ratio), and first-order physics-based models (Figure of Merit).

FRBS were first explored as expert systems in their most conventional sense, with a manageable number of inputs and a rule base defined by experts to model their decision making process. While no true validation of the system is possible without completing conceptual design of some number of alternatives, Figure 48 indicated that the model results were consistent with the benchmark evaluations performed.

The simplicity of the system also means that a simple validation of the rules and membership functions can be a measure of system performance, and that results are easily interpretable.

As discussed at the end of Section 3.3.4, the FRBS models provided a systematic means of representing expert elicited data, logically estimating the qualitative empty weight of a system. Though the results of the fit were not perfect, this is most likely a result of inconsistencies in the expert data, or effects considered by the expert that were too nuanced to be represented in the available inputs. The means to create a more accurate fuzzy system model could involve more complex combinations and higher numbers of membership functions. However it is much more likely that the best means to a more accurate model is to improve the elicitation process of scoring alternatives. By taking additional time to feed the elicited criteria ranges given back to the expert, and provide context by allowing more than one scoring at a time, it is believed that a more consistent data set of evaluations would result. It is also important to note that empty weight is a notoriously difficult criteria to calculate accurately in conceptual design. Figure 68 show several examples of actual versus error plots for parametric empty weight models from NDARC based on actual aircraft data [52]. These models indicate errors of over 25%, and other parametric models in [52] have indicated errors of up to 50% of their actual value. While a different type of model, these results provide some context for the FRBS expert model results.

The neuro-fuzzy DFESs also proved to be a very accurate model of physics-based data sets, much more so than the FRBSs. Not only this, but they were significantly simpler to construct and train, as well as executed more quickly. In contrast, the DFES model of expert elected data turned out not to be useful. Trying to adjust weights to satisfy all the data drove the system to provide ambiguous outputs.

In addition to validation of the systems, each of the fuzzy system models generated were also used to generate scores for the benchmark alternatives from Section

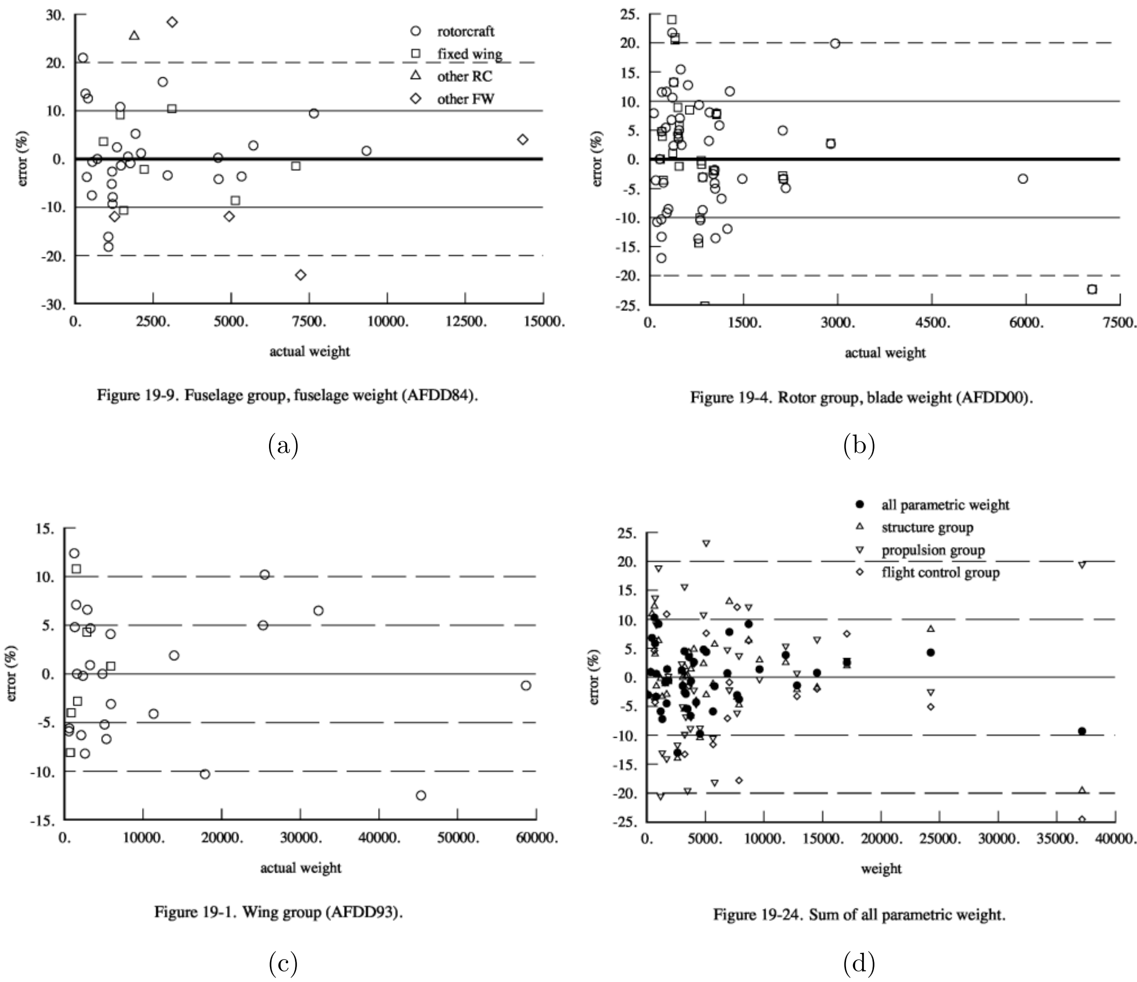


Figure 68: Examples of Validation of Parametric Weight Models for NDARC [52]

3.1.2, to compare to the scores given for those holistic alternatives. FRBS models were used for Empty Weight Ratio and Lift/Drag Ratio, while the DFES model was used for Figure of Merit. The results are shown in Figures 69, 70, and 71. The models compare very favorably to the benchmark alternatives in general. The empty weight results show close matches of the resulting functions for most all of the alternatives, with exceptions for Alternative 5 (Stopped Rotor Turbofan) and possibly Alternative 8 (Fixed Prop Fan-in-Wing Blended Wing Body). The fuzzy model identifies the stopped rotor as having a much better qualitative empty weight ratio than the benchmark score. However the result from the fuzzy model could be more accurate

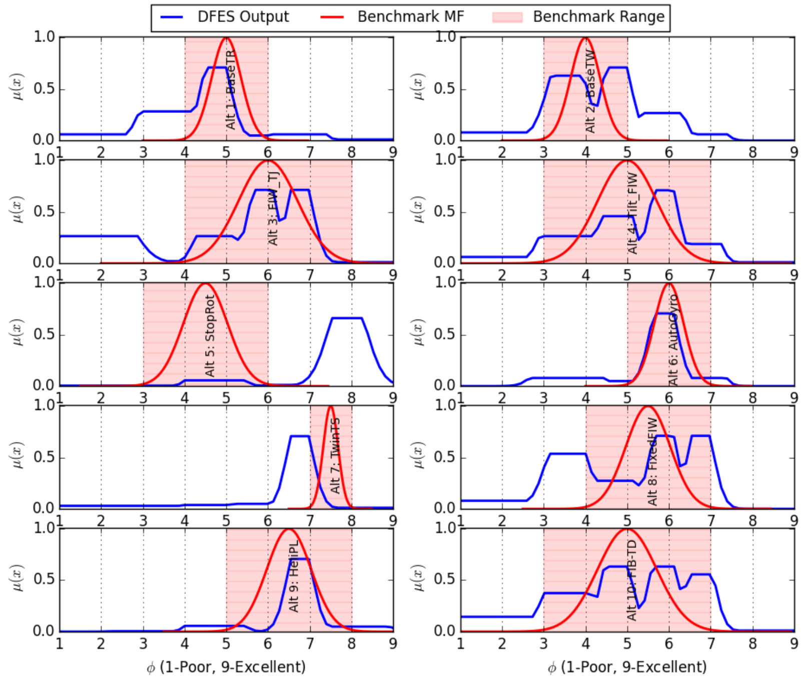


Figure 69: Comparison of Empty Weight Ratio FRBS System Model to Benchmark Scores

for the Stopped Rotor, with a light weight reaction drive system as well as the rotor and wing not accounting for separate weights. The system could be very light, but the complexity and powerful engines needed to account for the poor reaction drive system efficiency could increase its weight. The fuzzy model also indicates a much larger uncertainty about the Fan-in-Wing Blended Wing Body, although it is scattered mostly symmetrically around the benchmark evaluation.

Overall the hover figure of merit fuzzy model provides a close match for each of the benchmark alternatives. This may speak more to the accuracy of the benchmark scores, as the fuzzy model is based on Momentum Theory and only the inputs used are partially expert driven (partially historical data). In general, the fuzzy model indicates more uncertainty in the results through wider fuzzy output functions, and is slightly more optimistic in a number of cases. The system may not account particularly well for tilting systems where hover efficiency may need to be compromised to allow for a better propulsive efficiency in forward flight from the same propulsor.

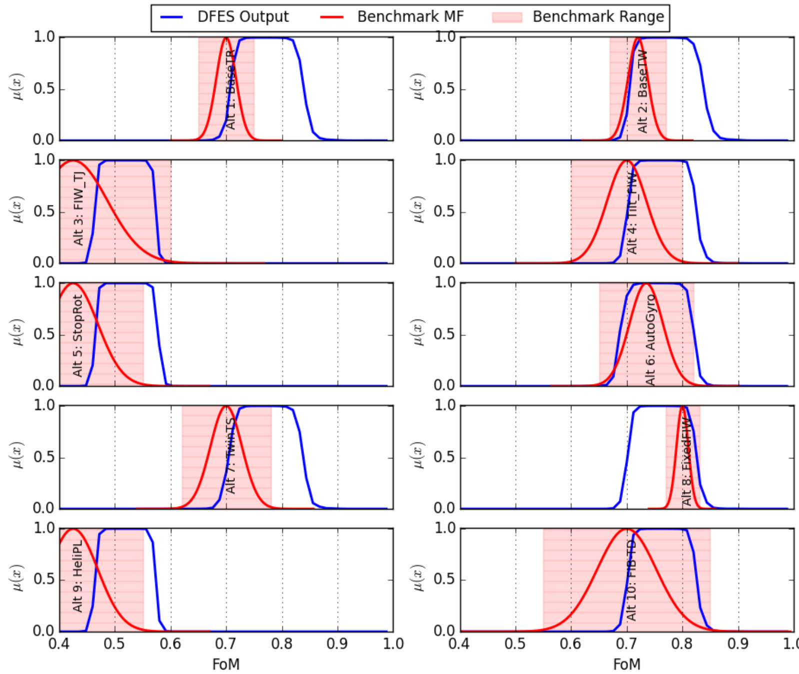


Figure 70: Comparison of Hover Figure of Merit DFES System Model to Benchmark Scores

It also may not account for losses in Figure of Merit due to tilting non-vertical lift propulsors, which will likely not be as efficient as the primary vertical lift mechanism, decreasing overall system efficiency. These issues could be accounted for in future iterations of the research by incorporating the primary DFES in a larger rule-based system to account for contingencies.

The Lift/Drag ratio fuzzy model as already been compared to the benchmark scores in Figure 48, but only with respect to one dimensional intervals. The comparison of the full membership functions functions indicate that the fuzzy model is a little more uncertain than the benchmark scores, but this is likely to be expected. A less conservative approach to the fuzzy model could be undertaken as well and the rulebase and membership functions adjusted to that respect.

The additional systems created in Section 3.3.9 were also used to generate comparisons to the scores for benchmark alternatives. These are shown in Figure 72 and Figure 73. The system propulsive efficiency functions match relatively well, with the

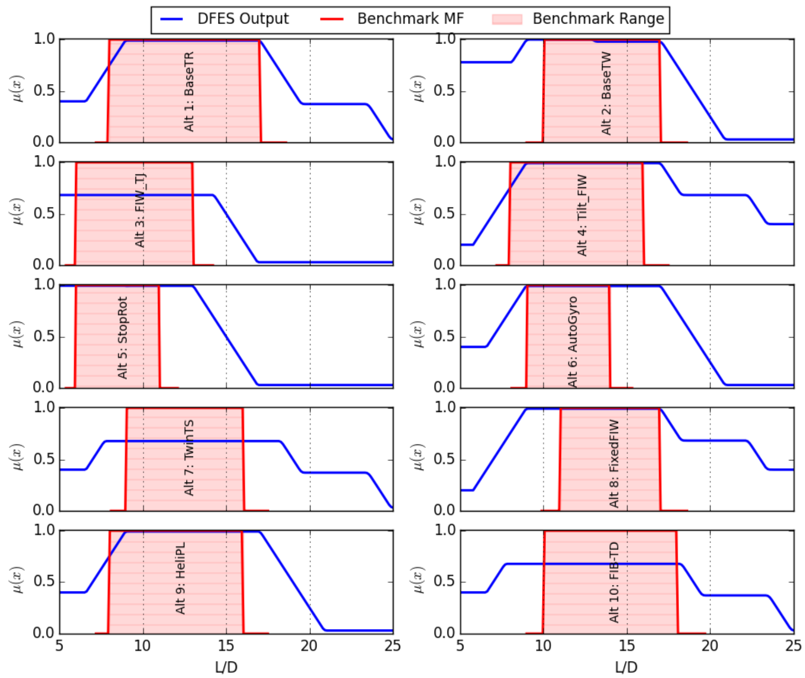


Figure 71: Comparison of L/D FRBS System Model to Benchmark Scores

fuzzy system models often indicating a bit more uncertainty in the results, which is to be unexpected. The maximum airspeed comparison is much more difficult to make, however. The fuzzy models of maximum airspeed are based on engines being sized for a maximum airspeed of 300 knots, resulting in the benchmark alternatives all showing fuzzy system results close to 300 knots. If each architecture were being developed individually, the maximum airspeed to size the aircraft and engines for would likely be improved carefully to a reasonable range for that system that did not affect other design goals. A general assumption that was necessary to make when scoring the synthesized alternatives in the benchmark was that this process would be completed later in design. This was necessary to differentiate the performance of these systems in the benchmark concept selection processes. In the data used to create the fuzzy model, higher airspeeds here are indicative of larger installed engines due to poor hover performance. It was for this reason that power installed was used as a design objective in the final framework rather than maximum airspeed.

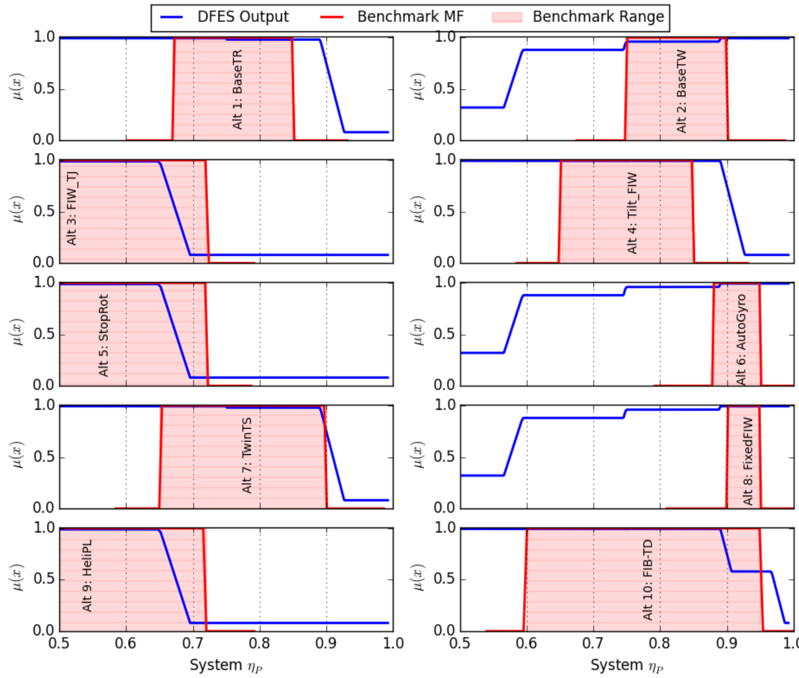


Figure 72: Comparison of Propulsive Efficiency FRBS System Model to Benchmark Scores

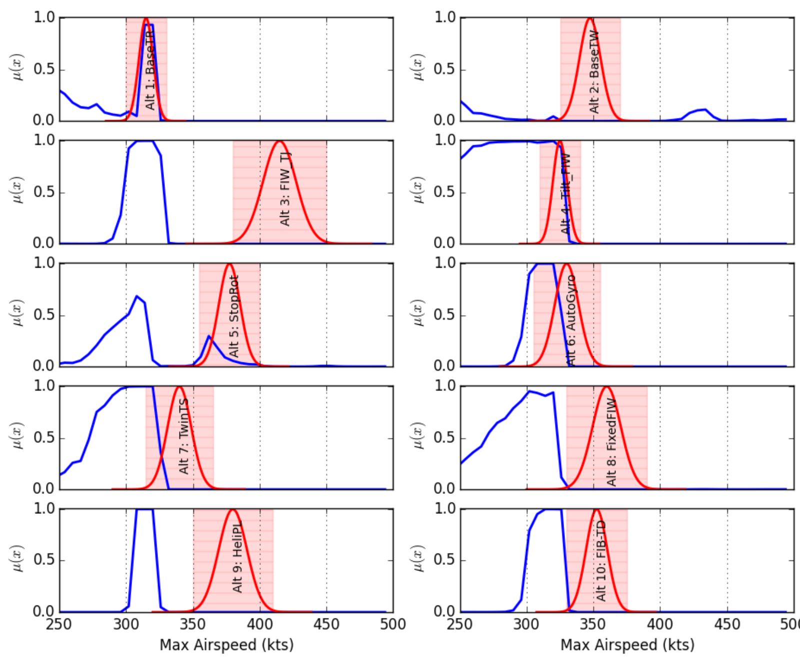


Figure 73: Comparison of Maximum Airspeed DFES System Model to Benchmark Scores

With these results in mind, the answer to Research Question 2 appears vary with the exact nature of the expert evaluation or data modeling required. Different types of fuzzy systems and approaches to building them provide the best means to model different criteria for a given problem. However fuzzy systems do appear to provide a relatively quick means to model the given criteria, and with reasonable enough accuracy at this early stage of the design process to identify architectures of interest.

It was hypothesized that rule-based fuzzy systems could provide a means to quickly and simply model performance for some attributes with a manageable set of expert defined rules. No reason was found to believe this is not true for attributes that can be estimated based on just a few inputs. Several systems were created for 2 inputs, and it is likely that systems with 3-4 inputs are feasible if the number of membership functions in the antecedent (inputs) are kept reasonable and the rule generation process is carefully managed. But for attributes that are based on more than a few inputs, the number of rules required to create a meaningful system becomes too large to be easily defined. But, for these systems, a training method using expert elected data was successful in producing logical systems to model more complicated attributes.

It was also hypothesized that these rule-based inference systems could be trained to provide fuzzy models of physics-based data. This was shown to work adequately for the Figure of Merit data, but at the same time it was found that NFS provided a much better, more accurate, means to model the physics-based data sets. With a limited number of fuzzy output membership functions, the FRBS proved to be somewhat coarse in modeling the continuous nature of the data. While the result was acceptable, it was not nearly as accurate as the NFS approach.

In the third hypothesis, it was thought that NFS would be easier to train than rule-based systems, as well as more accurate. As discussed, the NFS systems certainly proved to be more accurate when working with physics-based data, but it was not shown to be more accurate for the expert based data. With data that was not perfectly consistent, the DFES approach used adjusted its neural network to try and minimize the error for each data point, resulting in outputs that just had low membership values for large sections of the output domain. It is possible, that future research might find other NFS approaches would work better for this type of noisy, potentially inconsistent, expert data. In comparing the DFES approach with the FRBS approach for the same data, the neural-fuzzy system was certainly easier to setup and train. Decisions about how to structure the system were easier to make, and the training process for DFES systems was a quite faster than optimizing a FRBS for the same data. It is likely that for each application of the fuzzy system modeling approach, the best system will vary based on a number of factors, even some not foreseen here.

With a fuzzy model created for each of the desired design objectives (as well as models for gross weight and maximum airspeed), the research effort was continued, moving to the final phase. Each of the systems/models here was developed individually to predict the performance of a given architectural alternative with respect to a single attribute or objective. In the next phase, these pieces were combined to achieve a comprehensive picture of potential alternatives.

3.4 Modeling Systems and Framework Implementation

Once each of the fuzzy systems for evaluating alternatives had been trained and validated as necessary, it was combined into a single framework for design space exploration to address Research Objective 2 and Research Question 4. Using the fuzzy systems generated, a model was constructed in OpenMDAO to combine the individual fuzzy system models into a single framework. The framework serves as flexible multidisciplinary model combining the systems created and capable of evaluating selected architectures as desired based on their selected morphological matrix options. The framework structure is illustrated in Figure 74, and described as follows.

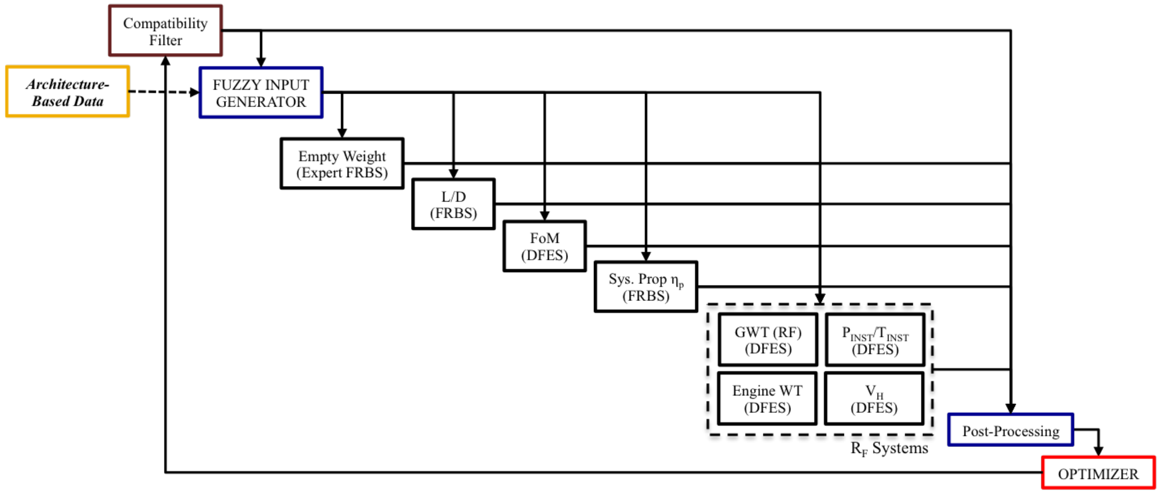


Figure 74: Full Evaluation Framework

Given a set of selected options from the morphological matrix (Table 7), a compatibility filter evaluates the options to ensure they are each compatible, and returns the number of incompatible options. If there is any incompatible combinations, a passthrough flag is used to skip all the individual models and return null values for any responses. Otherwise the options selected are passed to a fuzzy input generator that uses the architecturally dependent input data (outlined in Section 3.2.1) to generate fuzzy membership functions representing the necessary inputs for each individual fuzzy model. Once calculated, the system empty weight ratio, system Figure of Merit, and system (forward) propulsive efficiency are fed forward along with other necessary inputs into the R_F based systems (gross weight, installed power, maximum airspeed).

Before empty weight ratio and specific fuel consumption are passed to these systems, they are converted to the quantitative values required to drive the systems. While this qualitative to quantitative conversion is not ideal, it's necessary to meaningfully calculate the criteria that come from the R_F based systems. It was thought to be better to perform the qualitative to quantitative conversion externally, flexibly, and visibly in the framework, rather than transparently inside the R_F algorithm. The qualitative measures, having been evaluated on a $[1-9]$ scale, were linearly interpolated onto a quantitative scale using the ranges $[0.95, 0.50]$ for empty weight ratio, and $[1.05, 0.45]$ lb/hp/hr for specific fuel consumption.

Although not used in the final optimizations, the systems for gross weight and maximum airspeed were kept in the framework and calculated for each concept as meaningful criteria to track for potential architectures. It was also deemed worthwhile to show the performance of these systems, as many design problems are critically limited and optimized for gross weight and power installed.

Once each fuzzy model executes in the framework, the fuzzy criteria outputs are passed to a post-processing module. This module was used to convert the fuzzy outputs into meaningful crisp values or metrics for the optimizer. For the purposes of this research, the same five primary design criteria were used in the framework as the benchmarking and fuzzy MADM problems, except for maximum airspeed, which was replaced by power installed as discussed in Section 3.2.4.

3.4.1 Identification of Pareto Optimal Architectures

With a framework in place, the next step in the research was to determine some feasible means to utilize it and try and answer Research Question 4. Here the research sought to provide some feasible MCDM options to explore and understand the modeled design space, and provide guidance in selecting the best architectures for more detailed conceptual analysis and design. It is acknowledged that there exists some great number of possible means to utilize the framework and identify potential candidates, and it is likely that different applications may require different approaches and methods not explored here to answer these questions. The specific drive of this research, however, was to be able to use some MODM method to synthesize some number of promising alternative architectures. As hypothesized, this is attempted through the use of *a posteriori* multi-criteria optimization to seek out a Pareto frontier of optimal solutions. The results can then be analyzed to make decisions about how to proceed in system development with respect to the particular problem at hand, whether it be adding decision maker preferences to the objectives or other approaches. First, the optimization process is addressed.

Following the hypothesis for Research Question 4, the primary means researched for exploring the design space for potential architecture candidates researched was an evolutionary algorithm. The Nondominated Sorting Genetic Algorithm (NSGA) II [32], developed by Deb in the early 2000's, has become a popular means of addressing problems with multiple criteria to develop a large set of Pareto-optimal solutions, and was the optimization method used for finding potential concept architectures. More complicated single and multiple objective genetic algorithms have been proposed that utilize fuzzy objectives values [92], but a simpler approach was selected. Handling the various shapes and non-normalized fuzzy membership functions in the framework responses could be difficult and overly complicated.

Nondominated Sorting Genetic Algorithm II

The Nondominated Sorting Genetic Algorithm II is an evolutionary algorithm with the objective of seeking a diverse set of candidate solutions to a multi-criteria optimization problem, rather than seeking a single best solution [32]. Rather than combine all of the objectives into a single fitness function, the candidate population is sorted into a hierarchy of fronts based on Pareto dominance. A Pareto frontier is identified, and then a second front that dominate all but the first front, and a third front and so on, until all the candidates are accounted for. Thus each candidate is ranked, x_{rank} according to its Pareto dominance. Candidates on a given front are distinguished further by their crowding-distance, x_{dist}, a measure estimating the distance of a solution from its nearest neighbors on the same front. This is accomplished by averaging the side length of a cuboid formed by those nearest neighbors, as illustrated in Figure 75.

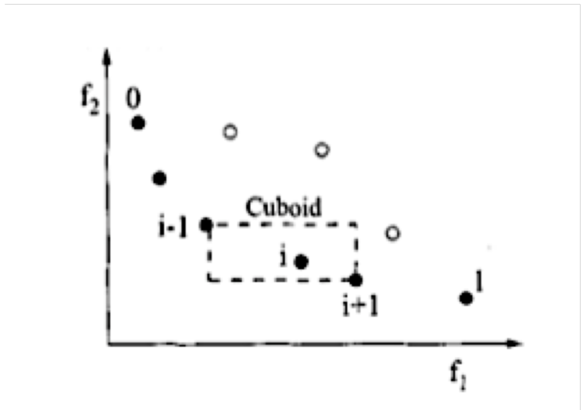


Figure 75: Crowding Distance Calculation [32]

To compare candidates and guide the selection process, a crowded-comparison operator, $\prec_n$, is used. Between two candidates with different non-domination ranks, the lower rank is preferred, and where ranks are equal, the candidate in a less crowded region is preferred. This pushes the algorithm towards a set of solutions spread out in the criteria/objective space , along the Pareto frontier. One primary change was made in the original NSGA-II algorithm. In order to eliminate duplicate solutions in the candidate population, each time the new population is selected, duplicate candidates are not permitted in the selection based on their binary genotype [26]. This is done to retain as much diversity as possible on each frontier. The drawback to this addition is that the population takes more generations to accelerate towards the Pareto frontier, as duplicate solutions on the Pareto frontier in the earliest generations (when few good and compatible candidates are available) would provide higher chances of promising candidates being selected for crossover. The primary loop of the algorithm is outlined in Table 21.

In addition to the original crowding distance operator other means to maintain a diverse solution space were also researched and explored. A number of distance operators for multi-objective optimization have been proposed and shown to work for

Table 21: NSGA-II Algorithm (from [32])

$R_t = P_t \bigcup Q_t$	combine parent and offspring population
$F =$ fast-non-dominated-sort(R_r)	$F = (F_1, F_2, ...)$, all non dominated fronts of R_t
$P_{t+1} =$ and $i = 1$	initialize next parent population
until $\lvert P_{t+1}\rvert + \lvert F_i\rvert \leq N$	until the parent population is filled (without duplicates)
crowding-distance-assigment(F_i)	calculated crowding-distance in F_i
$P_{t+1} = P_{t+1} \bigcup F_i$	add ith non dominated front to the parent population
$i = i + 1$	check the next front
Sort$(F_i, \prec_n)$	sort in descending order using $\prec_n$
$x \prec_n y$ if $(x_{rank} < y_{rank})$ or	
$(x_{rank} = j_{rank}$ and $x_{dist} > y_{dist})$	
$P_{t+1} = P_{t+1} \bigcup F_i[1 : (N - \lvert P_{t+1}\rvert)]$	choose the first $(N - \lvert P_{t+1}\rvert)$
$Q_{t+1} =$ make-new-pop(P_{t+1})	use selection/crossover/mutation to create new offspring (Q_{t+1})
$t = t + 1$	increment the generation counter

maintaining a diverse solution space in various specific applications[36, 125]. Mostly commonly measures are created for the objective or fitness space based on objective distance, similar to the crowding distance operator, but other measures are often used for the input (phenotype) and genetic (genotype) spaces. The purpose of optimizing the framework in the greater sense of the system development is to identify and analyze as many pareto optimal solutions as possible to allow the designer to make decisions about what architectures would be ideal to move forward with into a more detailed conceptual design. For this purpose it is desirable to maintain as much diversity as possible among candidates in the input space, to provide as many alternatives as possible. A hamming distance separation measure was created, by calculating the average hamming distance of each candidate to all the candidates in the same frontier. Using this measure in place of the crowding distance measure is explored in the application of optimization to the framework.

Application of NSGA-II Optimization

An NSGA-II optimizer was integrated into the framework to identify the best potential architectural alternatives. In order to use the optimization technique properly, the output (criteria performance) membership functions needed to be defuzzified into crisp values to be handled by the optimizer. The centroid defuzzification method was

Table 22: Optimization Parameters (NSGA-II)

Input Parameters	Objectives
VL System Type ($1 < x < 6$)	Empty Weight Ratio - Φ (centroid) - Maximize
VL System Prop ($1 < x < 3$)	System Hover Figure of Merit (centroid) - Maximize
VL System Drive ($1 < x < 3$)	Lift/Drag Ratio (centroid) - Maximize
VL System Tech ($1 < x < 5$)	Propulsive Efficiency - η_P (centroid) - Maximize
FWD System Prop ($1 < x < 4$)	Installed Power (centroid) - Minimize
FWD System Drive ($1 < x < 3$)	
FWD System Type ($1 < x < 4$)	
Wing System Type ($1 < x < 6$)	
Engine System Type ($1 < x < 4$)	
Population: 108	
Crossover Rate: 0.65	
Mutation Rate: 0.075	
Tournament Size: 3	

used here to weight the outputs to their membership values across the domain. This is done to help reflect the uncertainty in results as represented by the function, as opposed to using a mean of max, min of max, or max of max type defuzzification to try and reflect the most possible value.

As previously discussed, the objectives for the optimization were empty weight ratio (qualitative), system hover figure of merit, lift-to-drag ratio, system propulsive efficiency, and installed power. The input parameters were, of course, the options for each of the 9 system functional aspects. Each option input domain was bounded by the available options from the morphological matrix, and 4 bits were used in a grey coding approach to define each input gene with only the integer portion used (fractional floor). Tournament selection was used to select candidates for crossover, with the rest of the optimization setup shown in Table 22. These parameters were selected based on some trial and error experimentation with the optimization process and were found to perform well.

The optimization was first applied using the crowding distance measurement. No direct convergence criteria was used, but rather the progress was tracked in real time with several metrics (including populations of top frontiers, average crowding distance, and the minimum/maximum/average of each objective on the best frontier and population) and the ability to stop the algorithm if it appeared to converge. Figure 76 shows the progress of the optimization over 50 generations with respect to each objective. It is evident that the optimization identifies barely any improvement after approximately 20 generations.

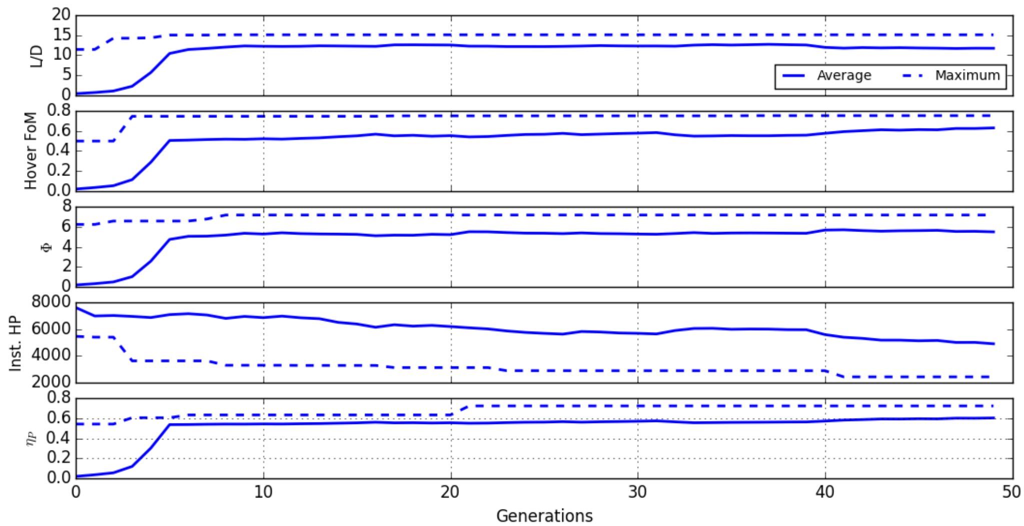


Figure 76: Optimization Progress (Crowding Distance)

The final population in this case represented a single Pareto frontier, which is illustrated in the scatter plot in Figure 77. Representations of Pareto frontiers in dimensions higher than 3 are notoriously difficult, and the non-convex nature of the data makes this frontier no different. One thing noticeable in the output is the banding of the points in the domains of several outputs. This phenomenon has two causes: in the case of the simpler FRBSs, there are a limited number of output membership functions and rules influencing the output and its centroid; and the nature of the inputs for some functional aspects (particularly with the limited size of the morphological matrix) means some systems, particularly those with just a few inputs are seeing a limited number of possible input combinations. These two issues can work separately and together in some circumstances to give the appearance of banding in the results.

It is clear that many of the points on the identified Pareto Frontier identified far exceed some of the DARPA design goals (L/D $\geq$ 10, Figure of Merit $\geq$ 0.75, and all designs are capable of 300 kts), while other points fall far short of some goals. Many designs have projected Figure of Merits below 0.6, L/D values below 10, and empty weight qualitative values are mostly below a 6.0. It was found to be difficult to visualize points that perform well with respect to every dimension, a problem that is addressed in the next section.

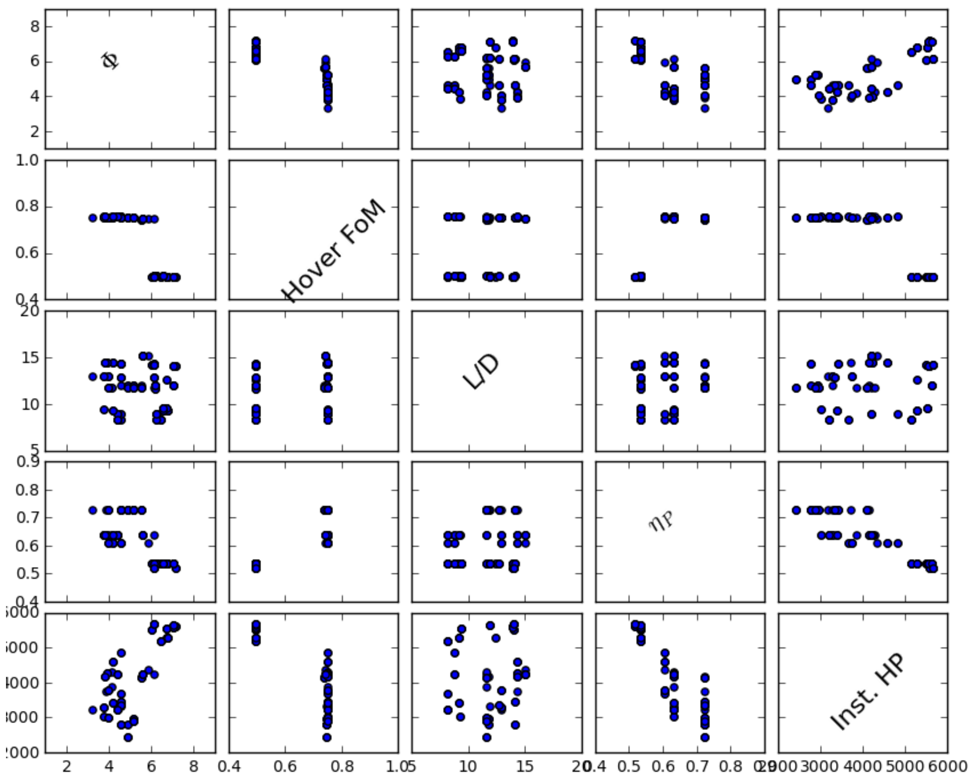


Figure 77: Optimization Results: Scatter Plot of Pareto Frontier (Crowding Distance)

The same optimization was also run using hamming distance in place of crowding distance in the NSGA-II algorithm. The rest of the algorithm remained exactly the same. The average population hamming distance and crowding distance were both tracked over the course of each optimization to understand the diversity the parameter and objective space respectively. Figure 78 compares the diversity of the population for both these spaces while the optimization used both crowding distance and hamming distance measures. The progress of the optimization with respect to each objective is shown when using hamming distance (HD) and compared to the previously shown progress using sorting distance (SD) in Figure 79

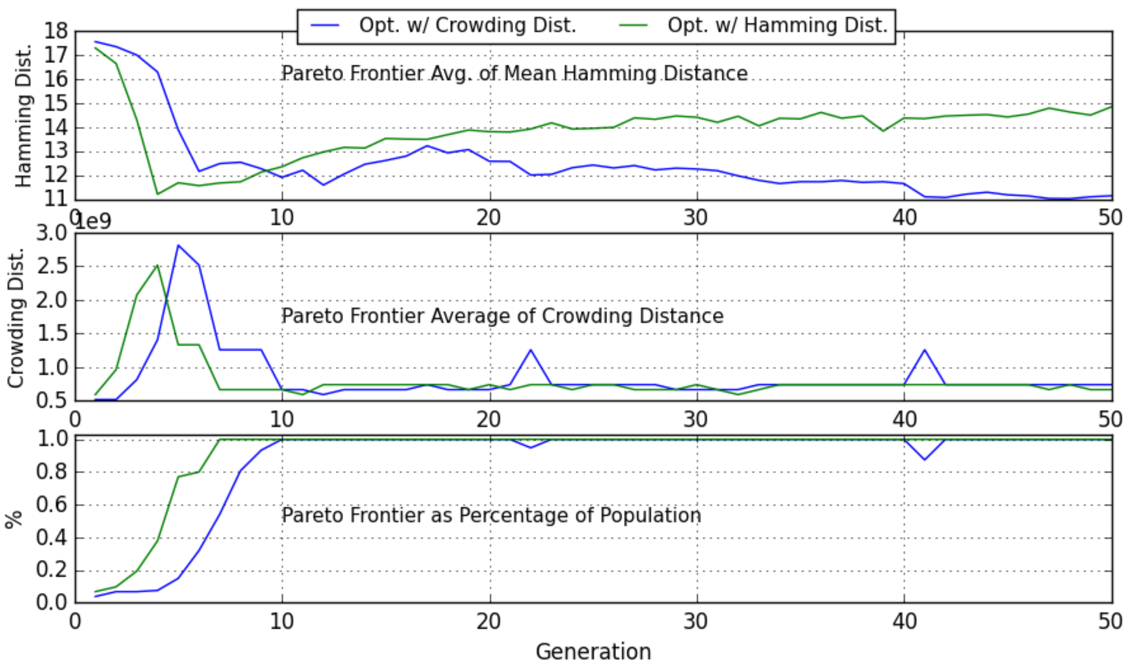


Figure 78: Comparison of Diversity Measures: Crowding Distance vs. Hamming Distance

When using the hamming distance measure the NSGA-II maintained much more diversity in the genotype space (more diverse architectures) as indicated by the the much higher average hamming distance in the population after approximately generation 17. Interestingly, using the hamming distance measure did not significantly affect the diversity of the objective space, with the crowding distances being almost identical from generation to generation when using the two measurements. The performance of the optimization progress is fairly close for each of the two measures. Using hamming distance appears to have slowed the initial progress some, but the converged averages and maximums are very close for each objective except Hover Figure of Merit. The lower average here is likely due to the increased diversity in the parameter space resulting in more low Figure of Merit Designs along the converged

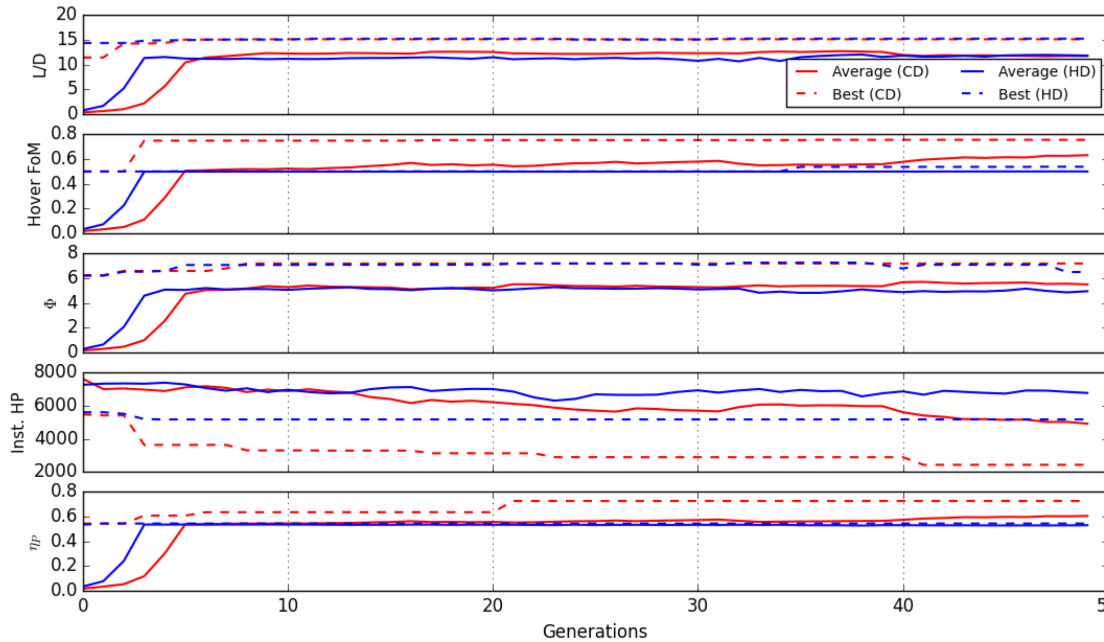


Figure 79: Optimization Progress (Both Distance Measures)

frontier.

As stated when defining the example problem in Section 3.1, the morphological matrix selected was purposefully kept small to allow for complete exploration of the design space. The framework was modified to run every one of nearly 13,000 compatible option combinations from the morphological matrix, and the Pareto Frontier was identified along with the second frontier of solutions only dominated by solutions on the Pareto Frontier.

To measure the optimization's performance, two measurements were used. In the first one, the the fraction of the optimization population that is in the first two non-dominated frontiers (Pareto and P-1) at each generation is measured. Secondly, for each point in the optimized frontier, the minimum Euclidian distance to the true Pareto frontier was found, and the distances across the population were averaged. This average minimum distance to the Pareto frontier shows just how close to the optimums the results were [32]. These measurements are presented for each generation in Figure 80. The crowding distance measure appears to clearly perform better here, with almost 50% of the final population being in the first two non-dominated frontiers. The optimizations also seem to indicate some progress in identification of more of the true optimum frontier. While progress of the maximum and average objective values in the population seem to have converged after 20-30 generations, it appears that identifying true convergence may be more difficult. Future research may be necessary to find the best convergence criteria, as a larger morphological matrix with millions or billions of compatible alternatives would take too long to run in its entirety.

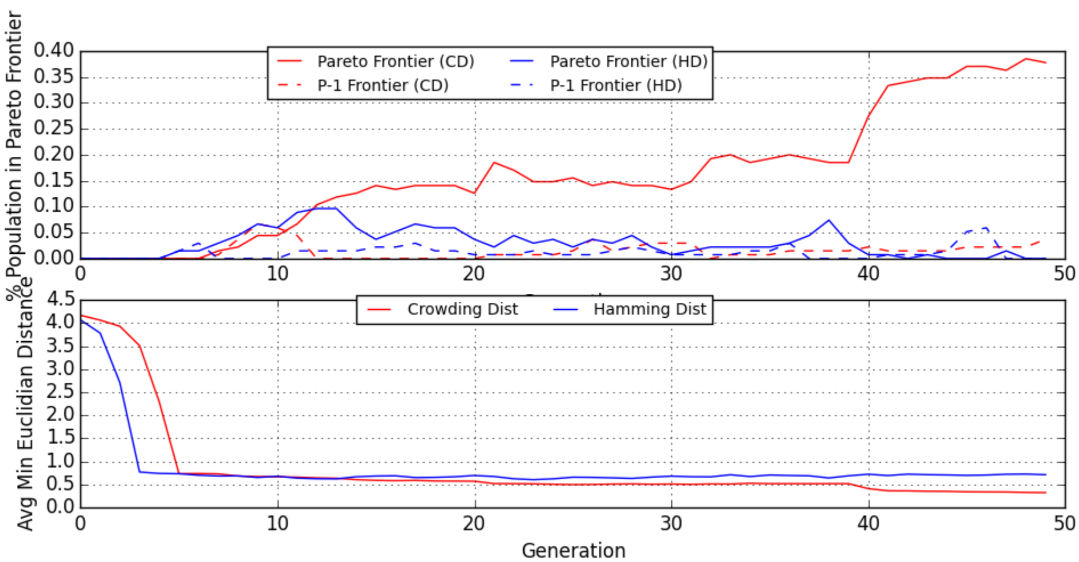


Figure 80: Optimization Progress with Respect to Identifying True Pareto Frontier

Figure 81 shows all of the compatible alternatives, with the true Pareto frontier and the optimization results highlighted. It is immediately noticeable that even the true best alternatives have some centroid values that fall below design goals with respect to some objectives. While there are some best alternatives that the optimization does not identify (particularly with respect to empty weight), the objective ranges for optimization results generally match those of the best alternatives. So while not all of the best alternatives were identified exactly, the optimization did not completely miss exploring all of the objective space.

The optimization process performed relatively well to identify a Pareto frontier

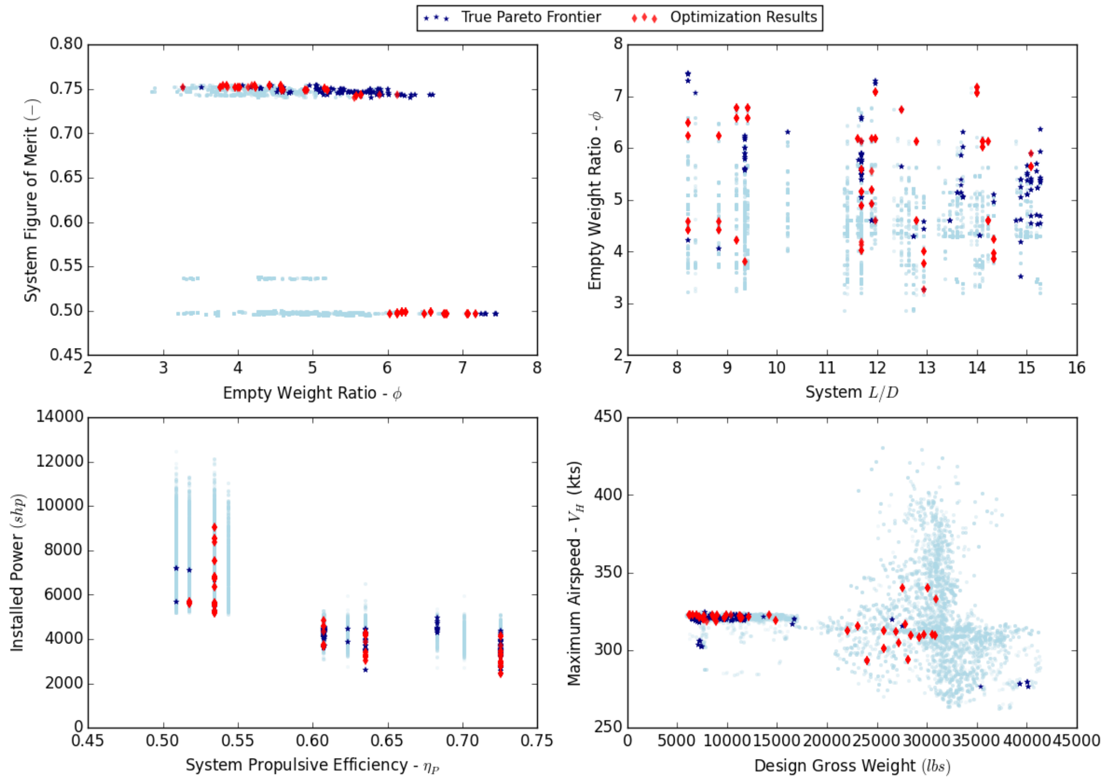


Figure 81: Results (Centroid) for all Compatible Alternatives with Pareto Frontier and Optimization Results

of alternatives using the centroid values, but it is clear that many of the identified alternatives excel beyond the requirements or a reasonable value with respect to several objectives, but perform poorly with respect to others. A more desirable set of solutions would be those that have the best chance of meeting a requirement or some defined goal with respect to all the objectives simultaneously compatible. Several means were explored to this end, and resulted in the development of a metric to identify the possibility of success of alternatives to satisfice all the objectives simultaneously. This is explored in the next section.

3.4.2 Fuzzy Possibility of Success (FPOS) Metric

Some modern means for managing uncertainty in multi-criteria systems design are metrics to assess a conceptual design's likelihood of meeting some set of design criteria simultaneously. One means of doing this is the Joint Probability Decision Making

(JPDM), discussed in Section 1.1.2. JPDM incorporates uncertainty in multi-criteria decisions using both empirical and analytical methods to calculate a design's Probability of Success (POS), a metric defining the probability of meeting all of the design criteria thresholds, as defined in Equation 61 [10, 64].

$$POS = P(Criteria_1 \in Constraint_1, ...Criteria_n \in Constraint_n) \tag{61}$$

The probability of meeting all constraints is often determined empirically using a joint cumulative probability distribution function, shown in Equation 62. Here a_i are sample criterion values, collected through sampling with a method like Monte Carlo Simulation. In this way, the results are not limited by assumptions about input parameter distributions or calculating complicated analytical parameters like correlation coefficients.

$$F(x_1, x_2, ...x_m) = \frac{1}{n}\sum_{i=1}^{n} I(a_{i1} \leq x_1, a_{i2} \leq x_2, ..., a_{in} \leq x_n) \text{ and} \tag{62}$$

$$I(a_{i1} \leq x_1, a_{i2} \leq x_2, ..., a_{in} \leq x_n) = \begin{cases} 1 \text{ for} & (a_{i1}, a_{i2}, ..., a_{im}) = (x_1, x_2, ..., x_m) \\ 0 & otherwise \end{cases}$$

In the same vein as JPDM, a metric was desired for this architecture evaluation framework to capture the relative fuzzy certainty of meeting a goal value or constraint for each of the design criteria, and all of the criteria simultaneously. In order to measure these uncertainties, the framework created does not require sampling, as the output of each system is a full fuzzy membership function. The dominance measure from Section 2.1.2 was utilized to measure the possibility of each criterion meeting the goal using the system's fuzzy output. The possibility that a system's fuzzy output, $\mu_O(x_i)$, will meet a goal, a_i (where x is to be maximized) is defined in Equation 63, where again x^* is defined as $x^* = min(x|\mu_A(x) = 1.0)$. If a criteria should be minimized the equation may be reversed. In this manner, any metric that reaches a 1.0 membership value at a point in the universe greater than the criterion goal, the possibility will be 1.0 (it's maximum), and otherwise the possibility is equal to the membership function. Figure 82 shows an example of calculating FPoS for a single criterion. For n criteria, the Fuzzy Possibility of Success (FPoS) is defined by Equation 64. This definition is derived from the joint possibility distribution outlined by Fuller [44].

$$P(a) = \mu_{\leq O}(a)$$
$$\text{where: } \mu_{\leq O} = \begin{cases} \max(\mu_O(x)) & for \;\; x \leq x^* \\ \mu_O(x) & for \;\; x > x^* \end{cases} \tag{63}$$

$$FPoS = \min_{i=1}^{n} \left(P(a_i)\right) \quad for \; i = 1, ..., n \tag{64}$$

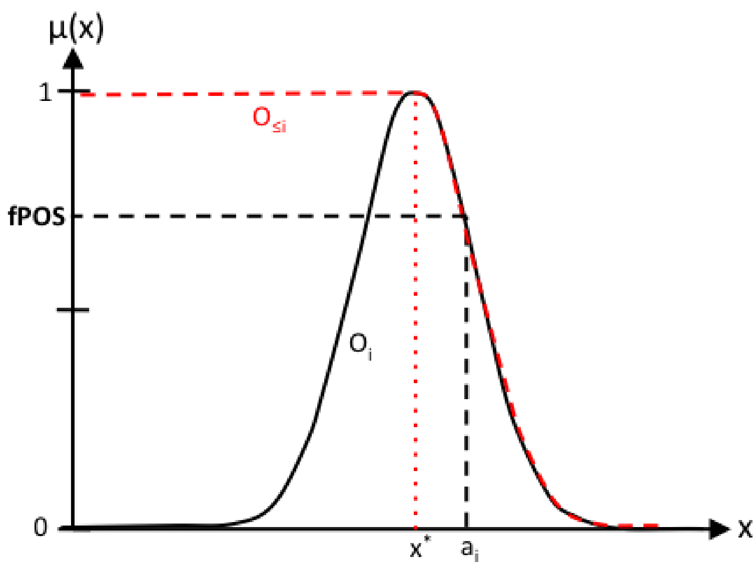


Figure 82: Fuzzy Possibility of Success (FPoS) Metric for a Single Criterion

This FPoS metric provides a measure of how likely an alternative is to meet the given design goal with respect to it's membership function for that system attribute/criterion. A score closer to 1.0 (the maximum value) indicate a more robust alternative, more likely to perform well with respect to the given goal, even given the measured uncertainty. While other metrics were considered, this approach, closely related to the dominance measure previously discussed in Section 2.1.2, was considered to be simple to calculate and easy to translate. The FPoS metric is next used in place of the output centroids in the optimization process, allowing for a search of alternatives with high FPoS values for each objective, indicating a more robust alternative.

Optimization with Fuzzy Possibility of Success (FPoS)

The optimization process was repeated with both diversity measures, this time attempting to maximize the FPoS for each individual metric. Again preferences were not applied here, but if they were applied *a priori*, a simpler optimization method could potentially be used with a single objective of the combined system FPoS, or the results of the NSGA-II could be weighted in decision making. Instead the goal is again to identify a Pareto frontier of best possible solutions for the designer to analyze and help determine alternatives to carry forward in the design process. In order to use the FPoS approach, goal values had to be set for each metric. The DARPA design goals of a Hover System Figure of Merit of 0.75 and an L/D of 10.0 were increased to 0.775 and 12.5 respectively to provide some buffer and reduce the number of alternatives meeting the goals. A qualitative empty weight ratio goal of 6.7 was used as it corresponded to the DARPA goal of $\phi = 0.60$ with the linear interpolation used to quantify empty weight ratio in the framework. The propulsive efficiency goal was set to 0.875 as it was deemed a reasonable value that provided a good limited number results. The results were all desired to exceed these goal values. Installed power was desired to be less than a goal of 3500 horsepower, as this is roughly equivalent to two

1800 horsepower class turboshaft engines, a reasonably available engine system. The optimization parameters used are outlined in Table 23.

Table 23: FPoS Optimization Parameters (NSGA-II)

Input Parameters	Objectives
VL System Type ($1 < x < 6$)	Empty Weight Ratio - Φ FPoS (Goal: ≥ 6.7) (maximize)
VL System Prop ($1 < x < 3$)	System Hover Figure of Merit FPoS (Goal: ≥ 0.775) (maximize)
VL System Drive ($1 < x < 3$)	Lift/Drag Ratio FPoS (Goal: ≥ 12.5) (maximize)
VL System Tech ($1 < x < 5$)	Propulsive Efficiency - eta_P FPoS (Goal: ≥ 0.875) (maximize)
FWD System Prop ($1 < x < 4$)	Installed Power FPoS (Goal: $\leq 3500hp$) (maximize)
FWD System Drive ($1 < x < 3$)	
FWD System Type ($1 < x < 4$)	
Wing System Type ($1 < x < 6$)	
Engine System Type ($1 < x < 4$)	
Population: 135	
Crossover Rate: 0.65	
Mutation Rate: 0.07	
Tournament Size: 3	

Both optimization processes were again repeated for 50 generations. Figure 83 shows the optimization progress for each objective (in this case, FPoS for each design goal). For this optimization NSGA-II using the hamming distance measure and that using the crowding distance measure, appear to give generally similar results, with each optimization performing better on some objectives than others. It appears that the optimizations can relatively easily find a frontier where all candidates have figure of merit and propulsive efficiency FPoSs of 1.0, but vary on performance with respect to the other objectives. Given the stochastic nature of the algorithm, these small differences do not indicate much a difference in the two measurement methods.

Figure 84 shows the diversity in both the parameter and objective space, as previously shown for the centroid optimization. The results are similar to optimization with the centroids, where using the hamming distance measurement results in some small gain in genome diversity, while using the crowding distance resulting in a very small increase in diversity in the objective space.

Using the full data set of compatible runs, the optimizations were again analyzed to see how well they performed in identifying the actual Pareto frontier of FPoS

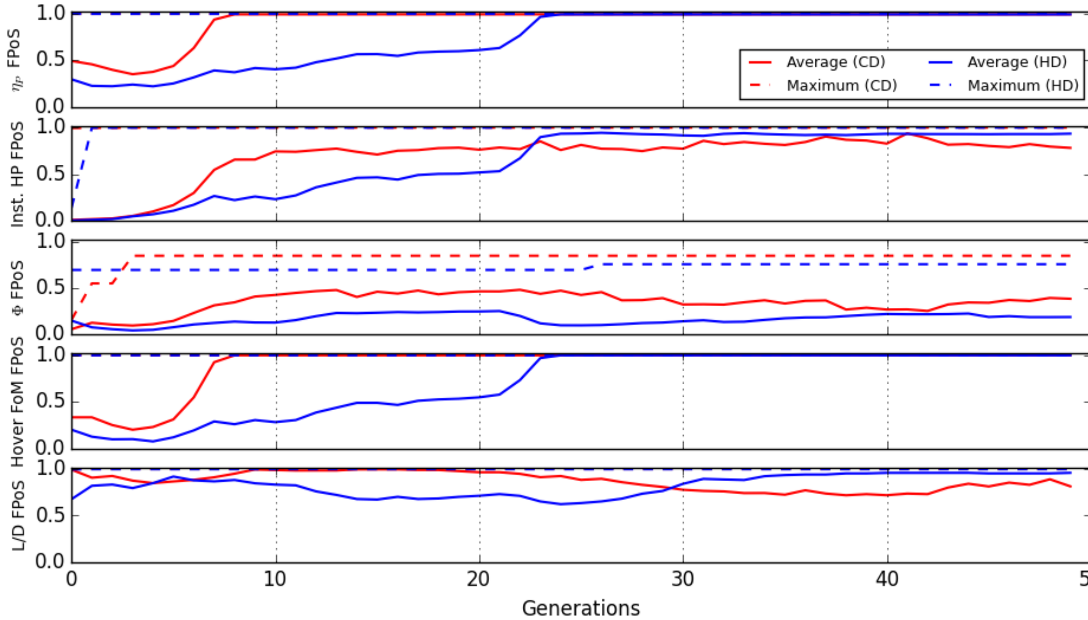


Figure 83: Optimization Progress for FPoS (Both Distance Measures)

objectives. Figure 85 shows this progress of the optimization using each method over 50 generations with respect to the two previously outline measurement methods. For the multi-criteria optimization with FPoS objectives, the hamming distance seemed to perform much better, with most to all of its population in the true Pareto frontier, and a distance to the Pareto frontier as nearly 0.0. One noticeable trend from this Figure and Figure 80, is that using crowding distance seems to result in a much more stable population, while the hamming distance measurement seems to result in a less stable population from generation to generation. This is likely a result of the drive to have a more diverse genome space, with the algorithm more prone to generate (through crossover and mutation) and accept new candidates. While this makes convergence much more difficult to identify, it is a signal that using hamming distance is indeed helping produce a diverse population.

The purpose of using a Fuzzy Possibility of Success metric was to help identify more balanced alternatives that were likely to satisfice design goals for each objective. Figure 81 was recreated again using the models' output centroids, but with the Pareto

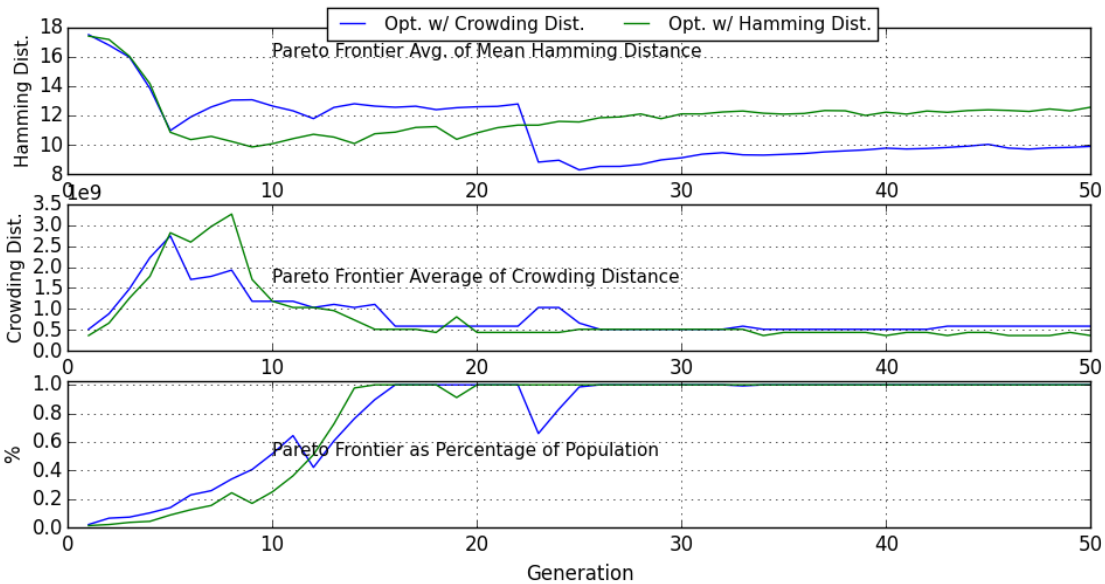


Figure 84: Comparison of Diversity Measures for FPoS Optimization: Crowding Distance vs. Hamming Distance

frontier of the FPoS objectives identified, and is shown as Figure 86, along with the points identified in the latest optimization. The alternatives identified in the Pareto frontiers have shifted away from the areas of poor performance, especially with respect to figure of merit, lift/drag ratio, and propulsive efficiency.

To understand the likelihood of the identified Pareto population meeting each design goal, cumulative histograms are shown in Figure 87 for both the identified Pareto population and the full set of compatible options. Also shown is a histogram of the FPoS for meeting all of the goals simultaneously (the minimum of the objective FPoSs for each alternative). Nominal goals of 12,000 lbs and 350 knots were added for gross weight and maximum airspeed to illustrate their distributions, but are not included in the system FPoS. It appears that meeting the empty weight ratio goal of 6.7 is the most difficult goal to meet, with a maximum FPoS of approximately 0.8. In contrast, the other goals could be met at a FPoS value of 1.0 for 30 to 80% of the total alternatives. The system level FPoS illustrates the difficultly of meeting each of the goals simultaneously. The Pareto population alternatives nearly all have an

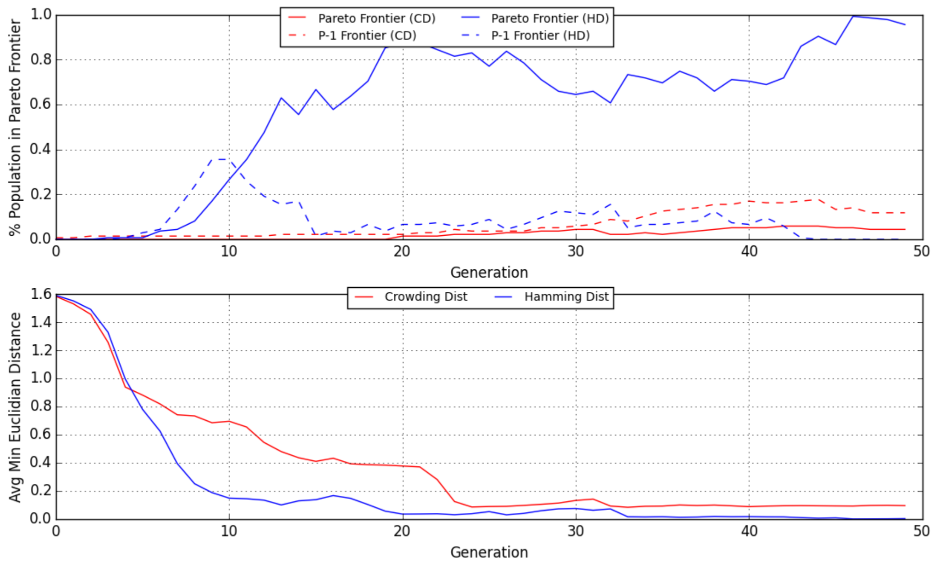


Figure 85: Optimization Progress for FPoS with Respect to Identifying True Pareto Frontier

FPoS value of 1.0 for each of the objectives except for empty weight ratio.

It is evident that the FPoS approach to identifying the best alternatives provides a better means of searching the design space for the best alternatives. The alternatives identified appear to have the best possibility of performing well with respect to all of the objectives simultaneously, and are most likely come closest meeting all of the necessary goals defined in the DARPA solicitation. Though there were tradeoffs involved in using the hamming distance or crowding distance measures, the hamming distance measure seemed to do the best job of identifying the Pareto frontier using the FPoS objectives. For these reasons this final optimization using hamming distance was used moving forward to the final step of decision making.

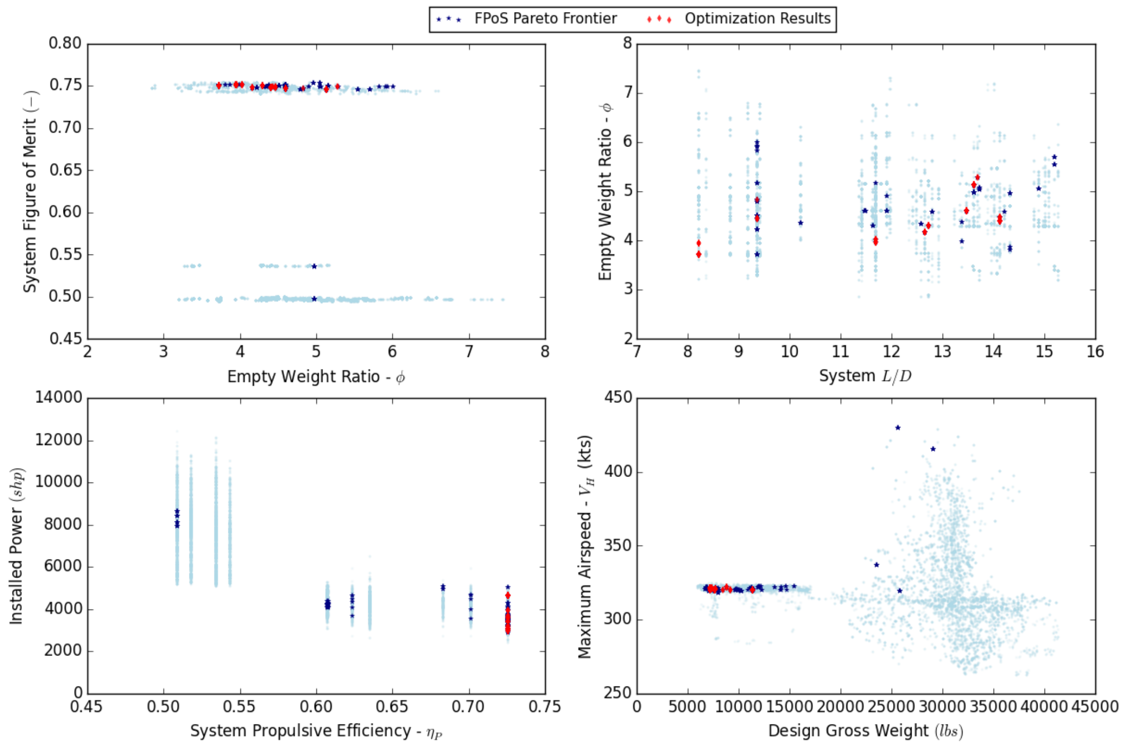


Figure 86: Results (Centroid) for all Compatible Alternatives with FPoS Pareto Frontier and FPoS Optimization Results

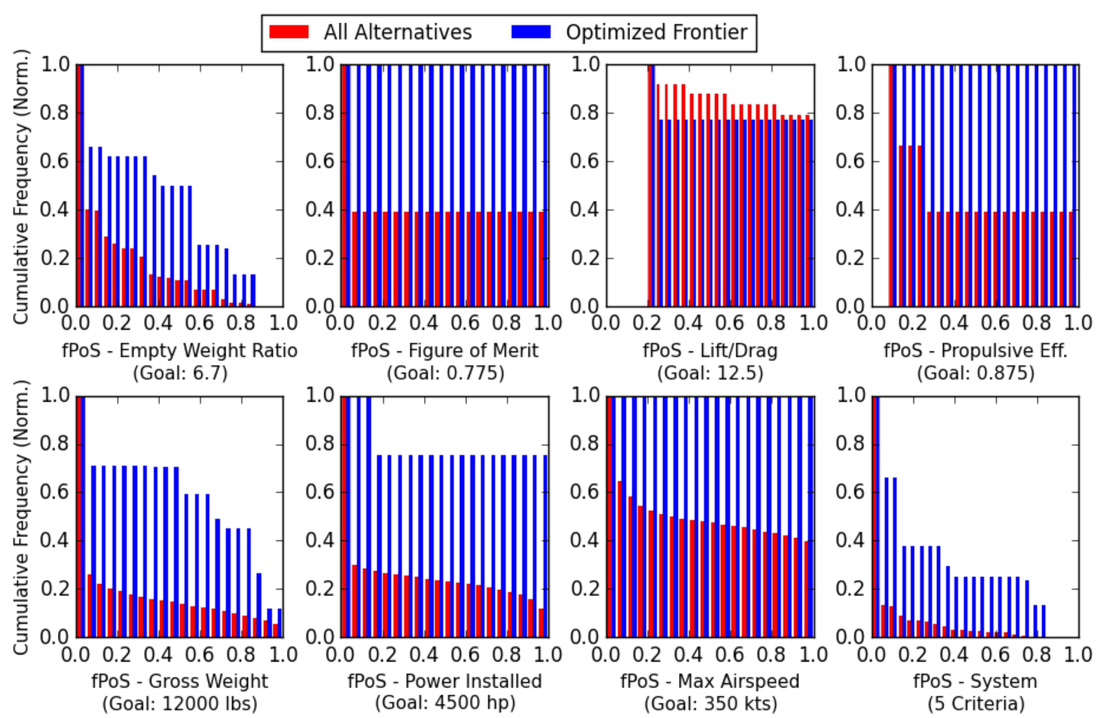


Figure 87: Normalized Cumulative Histograms for FPoS Objectives (Optimization using Hamming Distance)

3.4.3 Decision Making and Implications

Once the optimization had been completed, some means was required to take the results, interpret them, and then use them to make decisions in the framework of the design process. The explored style of *a posteriori* multi-objective optimization is meant to identify a set of Pareto optimal solutions for the decision maker to explore. For the purpose of this research, the decision to make is which architectural alternatives or families make sense to consider for a more rigorous and higher fidelity conceptual design and selection process. The concept behind the framework built is flexible enough that it could be used for other purposes, which is discussed further in Section 3.5, but only this primary purpose is explored here.

Once again there are likely a number of feasible means of using the data and tools available to make a decision. As part of keeping this research effort manageable, just a few ways of exploring the alternatives identified are explored here. The motivation, again, during this step of the process was to understand what the Pareto frontier of alternatives identified said about the available architectural design space and down select to a reasonable number of specific architectures to carry forward in design.

To better understand the results of the optimization several means were explored to visualize these Pareto alternatives in terms of their architectures. Understanding the most common options selected for each functional aspect was as simple as generating a histogram of the options used in the optimized population, but the performance of a given architecture across all the objectives is driven by the combinations of selected options. Figure 88 is a matrix of 2D histograms for each combination of functional system aspects from the alternatives from the optimized population. Each plot in the matrix illustrates the most common combinations of options from the morphological matrix seen in the Pareto frontier.

Reviewing Figure 88, some things can be quickly inferred. Virtually all of the alternatives in the Pareto population have shaft driven vertical lift systems, shaft

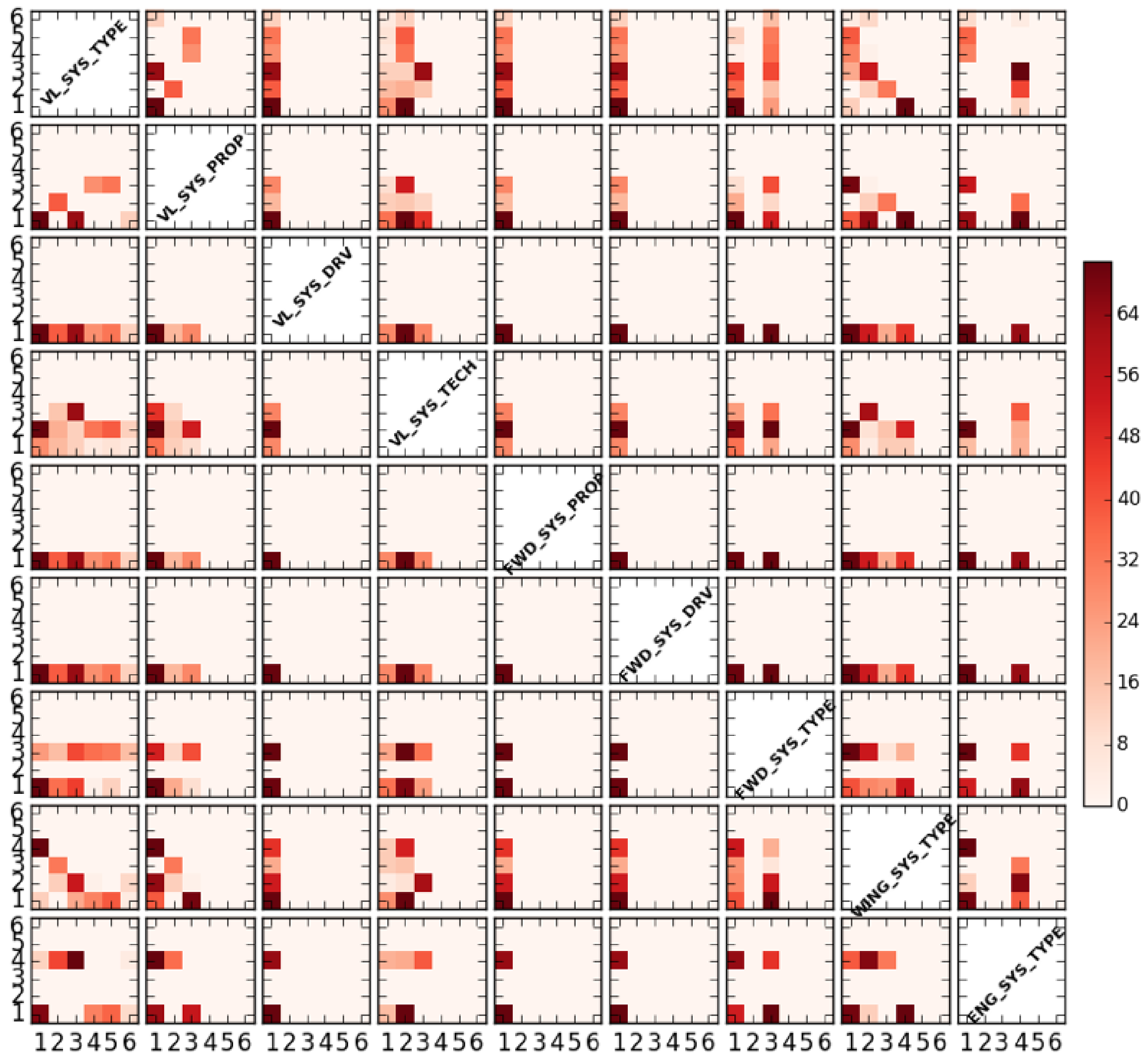


Figure 88: 2D Histograms of Selected Options for Pareto Alternatives (Frequency by Color)

a

driven forward drive systems, and nearly all use propellors as a forward propulsor. There are not very many tilting systems of any type, with the majority of alternatives all having fixed and clutched/disabled systems. Some other notable combinations with high frequencies are slowed rotor compounds for each of the open rotor types, (SMR/Transverse/Tandem with a variable RPM technology), and fan-in-wing/body alternatives with conventional wings an clutched/disabled propellors. Other common combinations can be traced throughout the matrix.

The 2D histogram matrix can provide some meaningful information about how specific combinations of alternatives, but it would be useful to be able to quickly view each of the alternatives on the Pareto frontier in terms of the combinations of their alternatives. Rather than list and review all the alternative combinations, a variation on the parallel plot was used, mapped onto the morphological matrix. Each line represents one combination in the Pareto population, as colored by the overall system FPoS value. This visualization with the results of the optimization is shown in Figure 89. It is quickly seen from this visualization that the best overall alternatives, based on their FPoS scores, seem to come mostly from embedded fan and tailsitter VTOL types. It also seems that many of the open systems are paired with a propellor vertical lift propulsor. This might not make sense at face value, but could be taken to mean that the requirements/mission favor higher disk loadings than are conventionally used on open rotor systems ($\geq 25lbs/ft^2$), and higher solidities (≥ 0.10). Some of the information presented here might not be seen in the multiple histogram plot. There also appears to be just a select few tilting (both VL and non-VL) systems, but their system FPoS values appear to be quite high. These options might not be evident in the histograms. It is apparent that it is important to carefully consider the results in a number of ways before making a decision.

Because one of the primary desires of using the researched approach was to be able to identify promising families of alternatives, some time was taken during the decision making process to review how well combinations of major functional aspects of the system performed. As an example, the alternatives in the Pareto population were sorted into families based on their Vertical Lift System, Vertical Lift Technology, and Wing Type. The FPoS values for each objective and the system were averaged across the resulting families, with the results listed in Table 24. The families can then be reviewed to show how major architectural decisions affect the various objectives. Over all the embedded fan families seem to perform well in terms of empty weight,

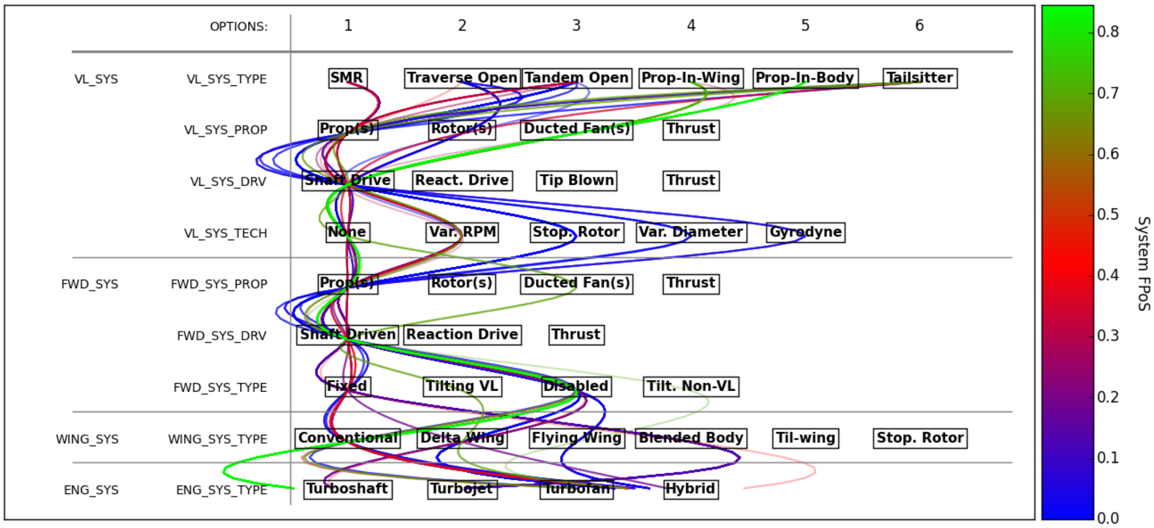


Figure 89: Illustration of Pareto Alternatives on Morphological Matrix

followed closely by those with a single main rotor. Most families have an FPoS value close to 1.0 for figure of merit, lift-to-drag ratio and propulsive efficiency, with just the empty weight ratio and power installed driving the system FPoS. These goals appear to be the easiest to meet in the design space, with empty weight being the most difficult.

Table 24: Identified Architecture Families

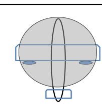
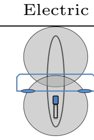
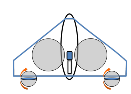
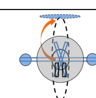
Family	VL System	VL Technology	Wing Type	Average FPoS					
				Φ	FoM	L/D	η_P	P_{inst}	System
1	Single Main	None	Conventional	0.377	0.999	0.990	0.993	0.977	0.377
2	Single Main	None	Blended Body	0.549	0.999	1.000	0.993	0.126	0.126
3	Single Main	Var. RPM	Conventional	0.377	0.999	0.990	0.993	0.977	0.377
4	Single Main	Var. RPM	Blended Body	0.549	0.999	1.000	0.993	0.126	0.126
5	Traverse Sys	None	Flying Wing	0.038	0.999	1.000	0.993	0.974	0.038
6	Traverse Sys	Var. RPM	Flying Wing	0.038	0.999	1.000	0.993	0.974	0.038
7	Traverse Sys	Stop Rotor	Delta Wing	0.001	0.999	0.198	0.993	0.998	0.001
8	Tandem Sys	None	Conventional	0.320	0.999	0.999	0.993	0.998	0.320
9	Tandem Sys	Var. RPM	Conventional	0.320	0.999	0.999	0.993	0.998	0.320
10	Tandem Sys	Stop Rotor	Delta Wing	0.001	0.999	0.198	0.993	0.999	0.001
11	Prop(s) In-Wing	None	Conventional	0.728	0.999	1.000	0.993	0.999	0.728
12	Prop(s) In-Wing	Var. RPM	Conventional	0.728	0.999	1.000	0.993	0.999	0.728
13	Prop(s) In-Wing	Var. RPM	Delta Wing	0.315	0.999	0.992	0.993	0.999	0.315
14	Prop(s) In-Body	None	Conventional	0.846	0.999	1.000	0.993	1.000	0.846
15	Prop(s) In-Body	Var. RPM	Conventional	0.846	0.999	1.000	0.993	1.000	0.846
16	Tailsitter	None	Delta Wing	0.098	0.999	0.992	0.993	0.998	0.098
17	Tailsitter	Var. RPM	Conventional	0.681	0.999	1.000	0.993	0.996	0.681
18	Tailsitter	Var. RPM	Delta Wing	0.098	0.999	0.992	0.993	0.998	0.098

Once the decision makers have an adequate understanding of the design space, they would likely use this information to select some of the families of alternatives or alternatives to explore more closely. In order to draw comparisons back to the benchmark activity, four complete alternatives were selected. Alternatives were selected across the range of identified families, to keep from overly similar alternatives. Selections were also made with the benchmark alternatives in mind, to avoid repetition of alternatives that might more traditionally be included in a preliminary analysis. It is also important to apply common sense to the process. Because the nature of the fuzzy systems use are to capture uncertainty and inexactness, the outcome of this methodology is not an exact design, but rather promising concepts to carry forward into exact analysis. As several of the models are meant to approximate expert evaluation, it is important to apply expert knowledge on the backend of this framework as well, reviewing results, and eliminating those alternatives that seem unlikely or do not make sense.

With these considerations in mind, four alternatives were selected from across the optimized population, and are outlined in Table 25. Two slowed-rotor concepts, a single main and a tandem were selected to represent their respective families. These concepts might seem to lack potential at first pass, but a closer look might be wise. They indicate poorer performance than other Pareto alternatives with respect to empty weight ratio, but were deemed to be interesting enough for further analysis. Having a hybrid electric system, and thus likely having a main rotor driven by an electric motor, would result in the ability to slow the rotor over a much greater range than a modern turboshaft is typically capable of (60-70% of optimum RPM). This could result in the necessary improvements over traditional compounds in maximum forward flight speeds. A fan-in-wing flying wing concept was selected to represent its family. The tilting forward propulsory might inadvertently provide an option for longitudinal control in a hover that other fan-in-wing concepts might not have. Lastly, a compounded tailgater was selected to represent a seemingly promising family in the identified alternatives. Compounding might provide a boost in both hover and forward flight efficiency over a shared propulsor. The fan-in-wing flying wing and compound tail-sitter are interesting variations on some of the commonly considered alternatives for high speed VTOL aircraft, and seemed worth further exploration. Each of these selected alternatives shows promise, and could be explored further.

The methods outlined here are just a few that could be used to understand the design space and make decisions. It is thought that the flexibility of the proposed framework might allow designers and decision makers to gather the data they want and structure their decision making process to fit their needs.

Table 25: New Alternatives 11-15

		Alternative 11	Alternative 12	Alternative 13	Alternative 14
	System Aspects	Compound Slowed SMR	Compound Slowed Tandem	Fan-In-Flying Wing	Tailsitter Compound
Vertical Lift	VL System	Single Main	Tandem Open Sys.	Propulsor(s) In-Wing	Tailsitter
	VL Propulsor	Rotor(s)	Rotor(s)	Ducted Fan(s)	Propeller(s)
	VL Drive	Shaft Driven	Shaft Driven	Shaft Driven	Shaft Driven
	VL Technology	Var. RPM	Variable RPM	None	Var. RPM
Fwd Propulsion	Fwd. Propulsor	Propellor(s)	Propellor(s)	Propellor(s)	Propellor(s)
	Fwd. Drive System	Shaft Driven	Shaft Driven	Shaft Driven	Shaft Driven
	Fwd. System Type	Fixed	Clutched/ Disabled	Tilting Non-VL Sys.	Clutched/ Disabled
Wing	Primary Wing Type	Conventional	Conventional	Flying Wing	Conventional
Engine	Engine Type	Hybrid Electric	Hybrid Electric	Hybrid Electric	Hybrid Electric
					

To compare the new alternatives identified back to the original benchmarked alternatives, each of the benchmarked alternatives combinations of morphological options were run through the framework individually. It would be ideal for traditional concept selection methods if these baseline concepts all performed very well, as they were identified to be ideal for the requirements outlined. The resulting FPoS values for all of the benchmarked alternatives, as well as the newly identified four alternatives are shown in Table 26. Many of the original alternatives perform very well with respect to the objectives, while some perform very poorly. Three alternatives, the Fan-in-Wing

Turbojet, the Stopped Reaction Rotor, and the Heliplane (all reaction drive or tip blown vertical lift drive types) have System Figure of Merit FPoSs of 0.0. Other alternatives, including the Auto-Gyro, Twin Rotor Tail-Sitter, and Fan-in-Wing Fixed Pusher, perform very well overall. Based on a decision maker's preferences, as well as their confidence in being able to improve on any of the objectives in the design process, any of the different alternatives could provide the best choice for moving forward.

Table 26: Comparing Benchmarked Alternatives to Newly Identified Alternatives

Alternative	Φ (Qual)	System FoM	Lift/Drag	Prop. Efficiency	Inst. Power (hp)	System
	Benchmark Alternatives					
Var. RPM Tilt-Rotor	0.038	1.0	0.998	0.980	0.971	0.380
Var. RPM. Tilt-Wing	0.377	1.0	0.998	0.980	0.135	0.135
Fan-in-Wing Turbojet	0.728	0.0	0.992	0.080	0.005	0.0
Tilting Fan-in-Wing	0.728	1.0	0.994	0.993	0.352	0.352
Stopped Reaction Rotor	0.667	0.0	0.200	0.800	0.0	0.0
Auto-Gyro	0.694	0.999	0.998	0.993	0.960	0.694
Twin Rotor Tail-Sitter	0.694	0.999	1.0	0.980	0.982	0.694
Fan-in-Wing Fixed Pusher	0.728	1.0	1.0	0.993	0.923	0.728
Heliplane	0.694	0.0	0.998	0.080	0.0	0.0
Fan-in-Body Tilt-Duct	0.377	1.0	0.992	0.993	0.999	0.377
	Identified Alternatives					
Compound Slowed SMR	0.119	0.999	0.990	0.993	0.980	0.119
Compound Slowed Tandem	0.320	1.0	0.999	0.993	0.998	0.320
Fan-In-Flying Wing	0.728	1.0	1.0	0.980	0.708	0.708
Tailsitter Compound	0.681	1.0	1.0	0.993	0.996	0.681

For a more in-depth comparison of the best alternatives, the fuzzy outputs were plotted for the results of several of the highest ranked benchmark alternatives alongside the newly identified alternatives. Each of the objectives output membership functions for these alternatives are shown in Figure 90. The results here indicate the specific uncertainty found in each alternative. These membership functions might be specifically compared to select the best alternative. Or in future phases of conceptual design, higher fidelity results could be compared back to these to better understand how another alternative might perform. Each application is likely to require slightly nuanced forms of analyzing the alternatives, and unique means of making a decision, but the framework outlined here is intended to provide enough flexibility to meet a wide range of applications.

3.4.4 Discussion

In this section, the culmination of this research effort was intended to combine all of the work done thus far and address Research question 4. Research question 4 asked "What is a feasible means to utilize the developed fuzzy system framework to explore very large design spaces and identify the best potential architectures?". As hypothesized a genetic algorithm, specifically a Nondominated Sorting Genetic Algorithm, the NSGA-II, was used to explore the design space in a number of ways. As expected, the algorithm performed fairly well in finding the "best" combinatorial alternatives for architectures. The optimizations were able to settle of populations that were close to or on the actual Pareto frontier, with the final optimization using hamming distance in the search for the FPoS objectives being almost entirely converged on the best alternatives in the design space. Though the problem was scaled to provide the ability to analyze the results of the optimization, there is little reason to expect that larger problems could not be addressed in the same manner. The results here indicate no reason to reject the hypothesis.

This was only one means to explore the design space, and it is more than likely that other approaches for optimization could provide similar results if they were able to handle the discrete nature of the problem. The *a posteriori* method of multi-criteria optimization used also did not account for designer preferences. If objective weightings (similar to those used in the benchmark) were included, they would potentially open up the problem for other means of exploration, including single objective algorithms based on an Overall Evaluation Criterion (OEC). Looking at other means of exploring the design space would be prudent in future efforts as better ways might be found.

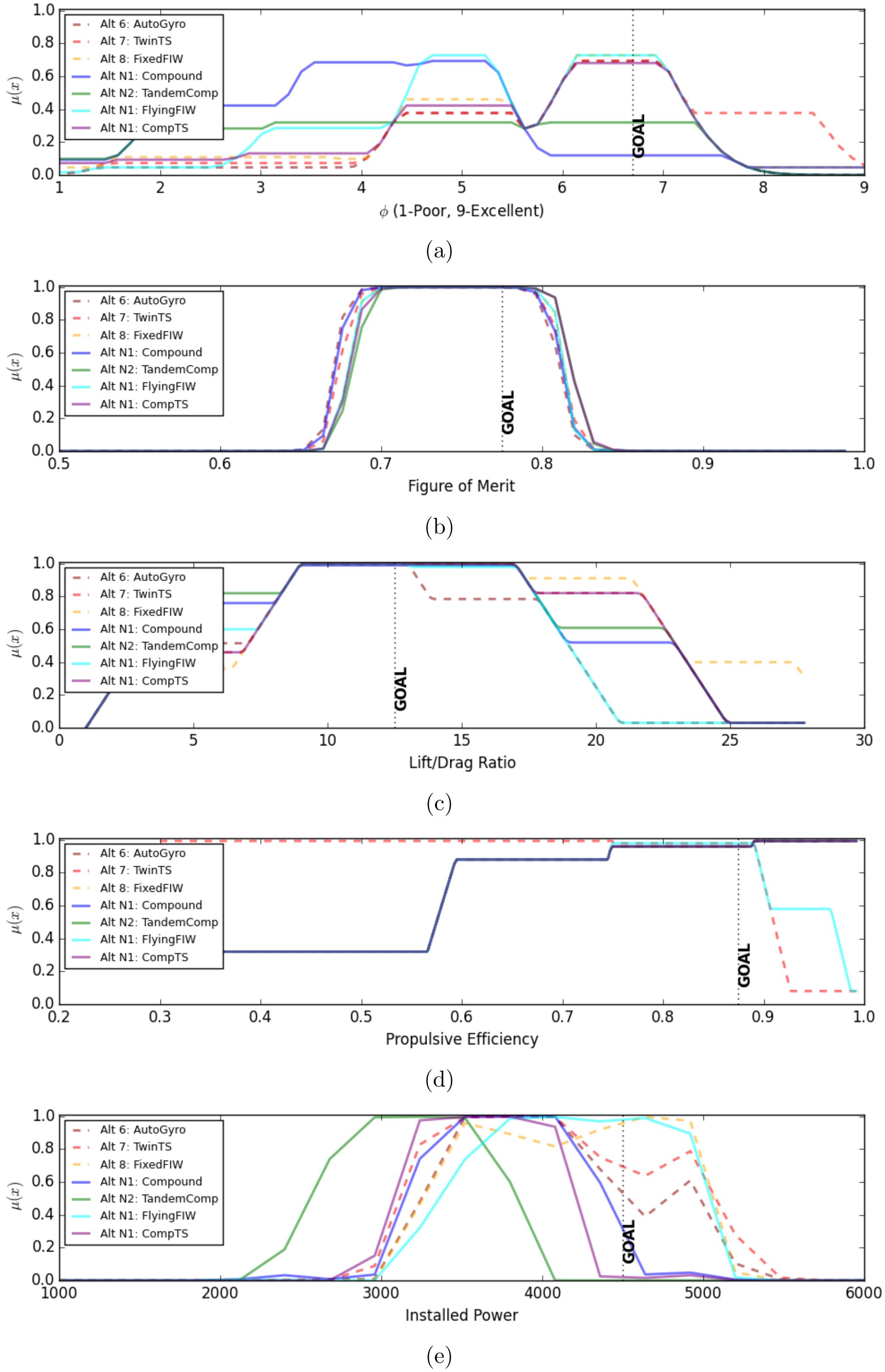


Figure 90: Comparison of Framework Evaluation of New Alternatives against Best From Benchmark for Each Attribute: a) Empty Weight Ratio; b) Figure of Merit; c) Lift/Drag; d) Propulsive Efficiency; e) Installed Power

3.5 Conclusions

It was found in this research that current methods and processes used to identify conceptual architectures for a new generation of systems, with new capabilities beyond what has been historically accomplished, lack the ability to adequately assess the full design space. Expansion of existing methods to larger number of architectural alternatives is limited by the cost and complexity of computational tools, while simpler tools require experts to identify and assess each alternative individually. The objectives of this research were to provide a means to quickly explore the full design space of architectures while capturing and understanding the uncertainty so inherent at this early stage of design. This was accomplished partially through the use of fuzzy set theory and the development of several types of fuzzy systems. Models based on fuzzy systems provided a means to quickly and logically assess alternatives with respect to a given criteria based on approaches to both model human experts and physics-based tools. These systems were linked to to a morphological matrix through analysis of the effect of architecture choices on primary design parameters. Once complete, the fuzzy models were integrated together and with the morphological matrix into a comprehensive framework. The framework, controlled by an optimizer could be used to search the design space within the morphological matrix and identify families of alternatives to carry forward into higher fidelity conceptual design.

Several research questions were identified for this research to answer in Section 1.3. Research Question 1 asked if fuzzy sets would be a feasible means of representing uncertainty in the course of this research. A comparison of probabilistic and fuzzy multi-attribute decision making (MADM) methods served as both a benchmark of current concept identification methods and answered this research question. While the results were perhaps less instinctive to understand, fuzzy set theory provided a means to address the uncertainty in the decision making problem that was comparable to probability theory. Advantages to using fuzzy sets included a more intuitive means to generate scores, slightly more information in the results, and a quicker computation time. The results of these experiments and the benchmark are discussed in Section 3.1.5.

Once fuzzy sets had been identified as a viable means to represent uncertainty in concept selection, the primary portion of the research was undertaken by exploring the development of fuzzy systems as a means for modeling system attributes of interest using input parameters gleamed from potential system architectural choices. This work sought to answer two closely linked research questions: Research Question 2 asked how fuzzy systems could be used for evaluating the architecture alternatives, while Research Question 3 asked how data should be elicited from experts in order to build and run these fuzzy system models. Several means of using fuzzy systems to evaluate different system attributes were hypothesized, and as discussed in Section 3.3.10, the results were mixed. The desire to use fuzzy inputs and produce fuzzy outputs meant using less standard means of fuzzy system construction and training. Fuzzy rule-based systems (FRBS) seemed to work best when creating simple intuitive systems where experts could define their own rules. They also provide a logical, structured fit to the elicited data for more system attributes with more complicated relationships. Neuro-fuzzy systems (NFS) were also demonstrated as an easy, accurate means to create a fuzzy model for physics-based data.

Inextricable from the process of developing these fuzzy systems, was the task of eliciting the expert data that would be used to drive the systems based on choices in the morphological matrix, as well as data used to train some of the systems. A survey was created as a framework for elicitation in conjunction with historical data to determine the effects that architectural choices have on major input parameters. Data was also collected on a set of synthesized alternatives to explore training fuzzy systems with expert data. While participation in the survey and online tool for evaluation was poor, some lessons were gleamed on how to improve the survey process and prompt experts to quickly gather consistent data. This was one area where concentrated future research could result in a much improved overall process to elicit consistent, logical expert data.

Once the fuzzy system models were built, they were linked in a single framework, driven by selections in the morphological matrix and wrapped in an optimizer. In order to answer Research Question 4, a multi-criteria genetic algorithm was utilized to try and determine as much of the set of Pareto optimal architectures as possible. Several means to identify the best alternatives in the design space were explored, including different means to maintain diversity among candidates, and a Fuzzy Possibility of Success (FPoS) metric to measure the possibility of satisfying goals in multiple criteria and all criteria. Once a Pareto population of alternatives was identified, several examples of how to visualize and explore the population with respect to the alternatives were shown in an attempt to drive decision making. Four potentially promising alternatives were selected across the major families and compared to the original benchmarked alternatives, showing the framework had identified promising new concepts.

In summary, a methodology and framework has been proposed to allow for the exploration of huge design spaces early in conceptual design, before higher fidelity modeling and simulation can be brought to bear on the problem. This approach is a meaningful tool to explore the feasibility of a new generation of systems, with new capabilities, capable of accomplishing missions that conventional vehicles cannot be empirically redesigned to perform. Rather than fall victim to a industrial momentum or designer bias, the presented methodology can provide a means to identify promising and robust concepts in the face of uncertainty. For the type of problems considered, this process represents a clear improvement over traditional concept selection methods.

Moving Forward

In reflecting upon the research completed in this effort there remain several areas of interest that could provide meaningful contributions to the concept selection process in general, and the framework presented here in particular. In completing the research here with respect to the specific goal of architecture identification and the example problem posed, it became clear that the general process could be used for other uses. One area of particular interest the general process could be used for is requirements identification. Generating specific requirements (mission, performance, capability, cost, etc.) for a system that has no historical basis can be a difficult process, and while methodologies exist for addressing the effects of requirements on individual concepts [8], the effects of requirements on architecture selection and the feasibility of requirements across varying architectures for revolutionary systems are often difficult to address. The methods and framework presented here could easily be reconfigured to include the effects of changing requirements in the models, as well as help system users and requirement developers communicate with material developers. Another area that is believed this research could be applied to is System of Systems (SoS) problems. Fuzzy models of individual system behaviors could be combined to understand how system selection for each function effects the overall SoS, using systems in the SoS in the morphological matrix in place of functional aspects.

The area most ripe for improvement with potential research for the method presented is clearly in the elicitation process. Can better means be developed to gather system evaluations to train FRBSs, and can a means to elicit rules directly be created? Answering these questions would greatly improve the process. Given the large number of potential NFS types, the question of if better systems exist for this problem is also one that could be answered specifically. If the process is to be applied to a larger real-world problem, many more system attributes would likely be modeled, while most of them would hopefully only require simple FRBS, some would inevitably require trained systems. As this effort was the most time-consuming part of the process, some means to expedite the training and optimization process would be prudent research as well. Each of these enhancements to the proposed process would likely provide significant improvement.

APPENDIX A

INPUT DATA ELICITATION SURVEY

The following survey was developed to elicit data for both the benchmark evaluation of pre-selected alternatives and the inputs to the fuzzy system portion of the framework as described in Section 3.2.

Data Elicitation
Development of a Fuzzy Expert System
(Thesis Project: Frank Patterson)

*The opinions provided in this survey are being used as example data in the research of fuzzy expert systems for the use of conceptual architecture development and selection, and any collected information will not be identifiable or attributable to individual participants in any way.

Your expert opinions are being solicited in order to develop and identify the best possible baseline conceptual configuration for the development of an aircraft design to submit for DARPA's VTOL X-Plane competition. The goal of the DARPA VTOL X-Plane is to address the challenges of developing a next-generation high-speed rotorcraft. In DARPA's words, "DARPA's VTOL experimental plane, or VTOL X-Plane, program seeks to overcome these challenges through innovative cross-pollination between the fixed-wing and rotary-wing worlds, with the goal of fostering radical improvements in VTOL flight. Rather than tweaking past designs and technologies, VTOL X-Plane challenges industry and innovative engineers to create a single hybrid aircraft that would concurrently push the envelope…". The program seeks to develop a 10,000 – 12,000 lbs demonstrator aircraft for test in 2017-2018.

Consider the following primary criteria for overall system evaluation:

1.	Empty Weight Fraction	(% Gross Weight)	Goal: $\leq$ 60%
2.	Max Cruise Speed	(KTAS)	Goal: 300-400 KTAS
3.	Vertical Lift Efficiency	(Figure of Merit)	Goal: 0.75
4.	Aerodynamic Efficiency	(Cruise L/D)	Goal: 10.0
5.	Propulsive Efficiency (Cruise)	(Cruise Max %)	Goal: n/a

Additionally, consider that the conceptual architecture for the system is defined by four general functional groups:

- Vertical Lift System: provides vertical lift for the aircraft in a hover
- Forward Drive System: provides forward thrust to propel the aircraft forward and counter drag forces in all forward flight regimes.
- Wing System: provides some or all lift during forward flight
- Propulsion System: provides the power to drive the vertical lift and forward drive systems and transmits the power to the propulsor(s)

These functional groups are further decomposed into 9 functional aspects of the system as shown below. These aspects will be used to form a morphological matrix (see the next page) that can be used to construct system configuration alternatives as shown in below.

Group:	**Vertical Lift System**				**Forward Drive System**			**Wing**	**Propulsion**
Functional Aspect:	Vertical Lift System Type	Vertical Lift Propulsor(s)	Vertical Lift Drive System	Vertical Lift Technology	Fwd Propulsor(s)	Fwd Drive System	Fwd System Type	Wing Type	Engine Type

Part 1: Example Synthesized Architecture Evaluation:

Directions:

1. Consider the following concept configurations holistically, and please estimate the range of possible system criterion values that would result from developing each system architecture. The system functional aspect options associated. You may evaluate empty weight on the qualitative scale (1-9) indicated below, where 9 is the best possible score, and 1 the worst.

You may assume for each evaluation that you are evaluating a mature system that has been designed and optimized to meet the VTOL X-Plane requirements to the extent the architecture allows. Each concept configuration is listed below, with a rough visual approximation, and the implementation for each of the identified functional aspects of the system. Part 3 may be referenced, as these configurations are assembled from the constituent options identified there.

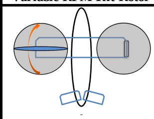
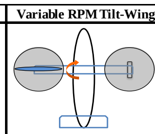
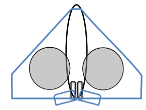
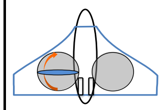
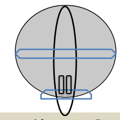
Qualitative Scale
For Empty Weight Ratio:

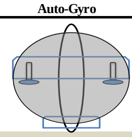
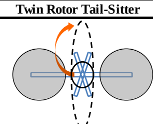
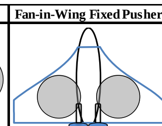
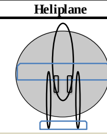
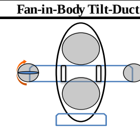
9	8	7	6	5	4	3	2	1
Best	Very Good	Good	Slightly Good	Moderate	Slightly Poor	Poor	Very Poor	Worst

Morphological Matrix

ASPECT		**OPTIONS 1**	**2**	**3**	**4**	**5**	**6**
Vertical Lift	**Vertical Lift System**	Single Main	Transverse Open System	Tandem Open System	Propulsor(s) In-Wing	Propulsor(s) In-Body	Tailsitter
	Vertical Lift Propulsor	Propellor(s)	Rotor(s)	Ducted Fan(s)	Direct Thrust		
	Vertical Lift Drive System	Shaft Driven	Reaction Drive	Tip Blown	Direct Thrust		
	Vertical Lift Technology	None	Variable RPM	Stopped Rotor	Variable Diameter	Gyrodyne	
Forward Propulsion	**Fwd Propulsor**	Propellor(s)	Rotor(s)	Ducted Fan(s)	Direct Thrust		
	Fwd Drive System	Shaft Driven	Reaction Drive	Direct Thrust			
	Fwd Type	Fixed	Tilting VL System	Clutched/Disabled	Tilting Non-VL Sys		
Wing	**Wing Type**	Conventional	Delta Wing	Flying Wing	Blended Body	Tilting Wing	Stopped Rotor
Engine(s)	**Engine Type**	TurboProp/Shaft	Turbojet	Turbofan	Hybrid Electric		

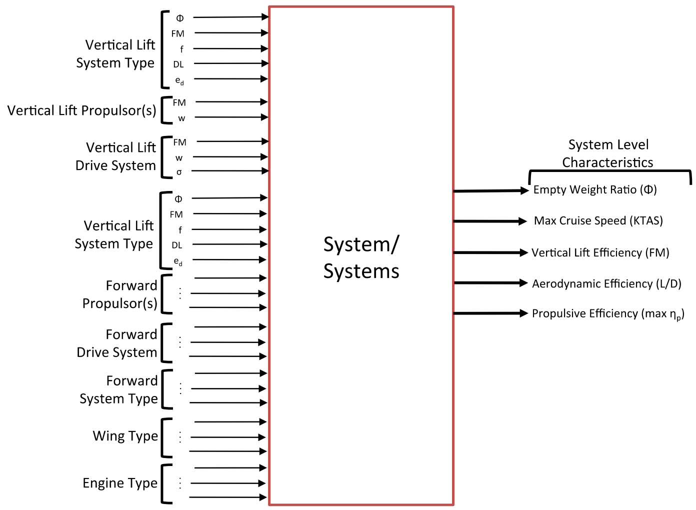
Concept Configurations:

Name		Variable RPM Tilt-Rotor	Variable RPM Tilt-Wing	Fan-In-Wing Turbojet	Tilting Fan-in-Wing	Stopped Rotor
						
ASPECT		Alternative 1	Alternative 2	Alternative 3	Alternative 4	Alternative 5
Vertical Lift	Vertical Lift System	Transverse Open System	Transverse Open System	Propulsor(s) In-Wing	Propulsor(s) In-Wing	Single Main
	Vertical Lift Propulsor	Rotor(s)	Propellor(s)	Ducted Fan(s)	Ducted Fan(s)	Rotor(s)
	Vertical Lift Drive System	Shaft Driven	Shaft Driven	Tip Blown	Shaft Driven	Reaction Drive
	Vertical Lift Technology	Variable RPM	Variable RPM	None	Variable RPM	Stopped Rotor
Forward Propulsion	Fwd Propulsor	Rotor(s)	Propellor(s)	Direct Thrust	Ducted Fan(s)	Direct Thrust
	Fwd Drive System	Shaft Driven	Shaft Driven	Direct Thrust	Shaft Driven	Direct Thrust
	Fwd Type	Tilting VL System	Fixed	Fixed	Tilting VL System	Fixed
Wing	Wing Type	Conventional	Tilting Wing	Delta Wing	Blended Body	Stopped Rotor
Engine(s)	Engine Type	TurboProp/Shaft	TurboProp/Shaft	Turbojet	TurboProp/Shaft	Turbofan
SCORES:		Min - Max	Min - Max	Min - Max	Min - Max	Min - Max
Empty Weight Fraction (% Gross Weight):		/	/	/	/	/
Max Cruise Speed (KTAS):		/	/	/	/	/
Vertical Lift Efficiency (Figure of Merit):		/	/	/	/	/
Aerodynamic Efficiency (Cruise L/D):		/	/	/	/	/
Propulsive Efficiency (Cruise Max %):		/	/	/	/	/

Name		Auto-Gyro	Twin Rotor Tail-Sitter	Fan-in-Wing Fixed Pusher	Heliplane	Fan-in-Body Tilt-Duct
						
ASPECT		Alternative 6	Alternative 7	Alternative 8	Alternative 9	Alternative 10
Vertical Lift	Vertical Lift System	Single Main	Tailsitter	Propulsor(s) In-Wing	Single Main	Propulsor(s) In-Body
	Vertical Lift Propulsor	Rotor(s)	Rotor(s)	Ducted Fan(s)	Rotor(s)	Ducted Fan(s)
	Vertical Lift Drive System	Shaft Driven	Shaft Driven	Shaft Driven	Reaction Drive	Shaft Driven
	Vertical Lift Technology	Gyrodyne	Variable RPM	None	Gyrodyne	None
Forward Propulsion	Fwd Propulsor	Propellor(s)	Rotor(s)	Propellor(s)	Direct Thrust	Ducted Fan(s)
	Fwd Drive System	Shaft Driven	Shaft Driven	Shaft Driven	Direct Thrust	Shaft Driven
	Fwd Type	Fixed	Fixed	Fixed	Fixed	Tilting Non-VL Sys
Wing	Wing Type	Conventional	Conventional	Blended Body	Conventional	Conventional
Engine(s)	Engine Type	TurboProp/Shaft	TurboProp/Shaft	TurboProp/Shaft	Turbofan	Hybrid Electric
SCORES:		Min - Max	Min - Max	Min - Max	Min - Max	Min - Max
Empty Weight Fraction (% Gross Weight):		/	/	/	/	/
Max Cruise Speed (KTAS):		/	/	/	/	/
Vertical Lift Efficiency (Figure of Merit):		/	/	/	/	/
Aerodynamic Efficiency (Cruise L/D):		/	/	/	/	/
Propulsive Efficiency (Cruise Max %):		/	/	/	/	/

Part 2:

Consider each functional aspect of the system and the listed characteristics or attributes of each aspect that affect any of the 5 system evaluation criteria. For each identified functional aspect, list any additional **primary** design variables or functional attributes that drive any of the system level criteria and are directly affected by system architecture. Indicate if there is enough information for these characteristics to be estimated quantitatively (see Part 3) during concept selection, or if they need to be estimated qualitatively. If quantitative assessment is possible, please indicate relevant units (and the full possible range of values that could be considered if possible).



Vertical Lift System

Vertical Lift System Type		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Subsystem Empty Weight (-)	1 - 9	Qualitative
Flat Plate Drag Area (-)	1 - 9	Qualitative
Disk Loading (w)	3 – 150	lb/ft²
Download (e_d)	0.0 – 0.3	%GWT
Thrust to Power Ratio (T/P)	0.1 – 20	lb/hp

Vertical Lift Drive System		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Subsystem Empty Weight (-)	1 - 9	Qualitative
Drive System Efficiency (η_d)	0.3 – 1.0	-

Vertical Lift System Propulsor(s)		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Subsystem Empty Weight (-)	1 - 9	Qualitative
Disk Loading (w)	3 – 150	lb/ ft²
Solidity (σ)	0.05 - 0.4	-

Vertical Lift Technology		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Subsystem Empty Weight (-)	1 - 9	Qualitative
Flat Plate Drag Area (-)	1 - 9	Qualitative
Disk Loading (w)	3 – 150	lb/ft²
Drive System Efficiency (η_d)	0.3 – 1.0	-

Forward Propulsion System

Forward Propulsor(s)		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Subsystem Empty Weight (-)	1 - 9	Qualitative
Prop. Efficiency (η_p)	0.6 - 1.0	-
Max Cruise Speed (kts)	150 - 550	kts

Forward Propulsion System Type		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Subsystem Empty Weight (-)	1 - 9	Qualitative
Max Cruise Speed (kts)	150 - 550	kts
Prop. Efficiency (η_p)	0.6 - 1.0	-
Flat Plate Drag Area (-)	1 - 9	Qualitative
Thrust to Power Ratio (T/P)	0.1 – 20	lb/hp
Hover Figure of Merit	0.4 - 1.0	-

Forward Propulsion Drive System		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Subsystem Empty Weight (-)	1 - 9	Qualitative
Drive System Efficiency (η_d)	0.3 – 1.0	-

Wing System

Wing Type		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Empty Weight (-)	1 - 9	Qualitative
Max Cruise Speed (kts)	150 - 550	kts
Cruise Aero. Efficiency (L/D)	5 - 25	-
Wing Loading (W/S)	5 - 150	lb/ft^2
Flat Plate Drag Area (-)	1 - 9	Qualitative

Propulsion System

Engine Type		
Major Functional Characteristics (Inputs)	**Qualitative/ Quantitative & Range For Consideration**	**Units**
Empty Weight (-)	1 - 9	Qualitative
Specific Fuel Consumption	1 – 9	Qualitative

Part 3: Evaluation of System Functional Aspects

Directions:
For Part 3, consider the conceptual architecture of the system is undefined, with the exception of the functional aspect of the system listed below. For each indicated option for that functional aspect:

Please estimate the possible range of each input characteristic (and any additional inputs you might have indicated in Part 2) when each option is selected to perform its respective system functional aspect. The qualitative scale below can be used for characteristics with qualitative scoring.

(The min/max may be left blank if no effect is anticipated or a full range possible). Some functional options may have no effect on some characteristics (e.g. a fixed forward propulsion system type may have no effect on hover figure of merit, where a tilting system may limit hover efficiency; or solidity has no application to a forward propulsor of direct thrust) and could be left blank.

9	8	7	6	5	4	3	2	1
Very High Benefit	High Benefit	Benefit	Slight Benefit	No Effect	Slight Detriment	Detriment	High Detriment	Very High Detriment

Example:

Forward Propulsor(s)

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Propeller(s)	Rotor(s)	Ducted Fan(s)	Direct Thrust		
			Min / Max	Min / Max	Min / Max	Min / Max		
Subsystem Empty Weight (-)	1 - 9	Qual	6 / 9	4 / 7	2 / 5	/		
Max Cruise Speed (kts)	150 - 550	kts	200/400	/350	/370	300/550		
Prop. Efficiency (η_p)	0.6 - 1.0	-	0.7/0.95	0.6/0.8	0.65/0.9	/0.7		

FUNCTIONAL GROUP: Vertical Lift System

Vertical Lift System Type

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Single Main	Transverse System	Tandem System	Propulsor(s) In-Wing	Propulsor(s) In-Body	Tail Sitter
			Min / Max	Min / Max	Min / Max	Min / Max	Min / Max	Min / Max
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/	/	/
Flat Plate Drag Area (-)	1 - 9	Qual	/	/	/	/	/	/
Disk Loading (w)	3 – 150	lb/ft²	/	/	/	/	/	/
Download (e_d)	0.0 – 0.3	%GWT	/	/	/	/	/	/
Thrust to Power Ratio (T/P)	0.1 – 20	lb/hp	/	/	/	/	/	/
			/	/	/	/	/	/
			/	/	/	/	/	/
			/	/	/	/	/	/

Vertical Lift Propulsor(s)

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Propeller(s)	Rotor(s)	Ducted Fan(s)	Direct Thrust		
			Min / Max	Min / Max	Min / Max	Min / Max		
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/		
Disk Loading (w)	3 – 150	lb/ ft²	/	/	/	/		
Solidity (σ)	.05 - 0.4	-	/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		

FUNCTIONAL GROUP: Vertical Lift System

Vertical Lift Drive System

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Shaft Driven	Reaction Drive	Tip Blown	Directed Thrust		
			Min / Max	Min / Max	Min / Max	Min / Max		
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/		
Drive System Efficiency (η_d)	0.3 – 1.0	-	/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		

Vertical Lift Technology

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			None	Variable RPM	Stopped Rotor	Variable Diameter Rotor	Gyrodyne	
			Min / Max	Min / Max	Min / Max	Min / Max	Min / Max	
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/	/	
Flat Plate Drag Area (-)	1 - 9	Qual	/	/	/	/	/	
Disk Loading (w)	3 – 150	lb/ft^2	/	/	/	/	/	
Drive System Efficiency (η_d)	0.3 – 1.0	-	/	/	/	/	/	
			/	/	/	/	/	
			/	/	/	/	/	
			/	/	/	/	/	
			/	/	/	/	/	

FUNCTIONAL GROUP: Forward Propulsion System

Forward Propulsor(s)

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Propeller(s)	Rotor(s)	Ducted Fan(s)	Direct Thrust		
			Min / Max	Min / Max	Min / Max	Min / Max		
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/		
Max Cruise Speed (kts)	150 - 550	kts	/	/	/	/		
Prop. Efficiency (η_p)	0.6 - 1.0	-	/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		

Forward Drive System

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Shaft Driven	Reaction Drive	Directed Thrust			
			Min / Max	Min / Max	Min / Max			
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/			
Prop. Efficiency (η_p)	0.6 - 1.0	-	/	/	/			
			/	/	/			
			/	/	/			
			/	/	/			
			/	/	/			
			/	/	/			
			/	/	/			

FUNCTIONAL GROUP: Forward Propulsion System

Forward System Type

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Fixed	**Tilting VL System**	**Clutched/Disabled**	**Tilting Non-VL System**		
			Min / Max	**Min / Max**	**Min / Max**	**Min / Max**		
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/		
Max Cruise Speed (kts)	150 - 550	kts	/	/	/	/		
Prop. Efficiency (η_p)	0.6 - 1.0	-	/	/	/	/		
Flat Plate Drag Area (-)	1 - 9	Qual	/	/	/	/		
Thrust to Power Ratio (T/P)	0.1 – 20	lb/hp	/	/	/	/		
Hover Figure of Merit	0.4 - 1.0	-	/	/	/	/		
			/	/	/	/		
			/	/	/	/		

Wing Type

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Conventional	**Delta Wing**	**Flying Wing**	**Blended Body**	**Tilting Wing**	**Stopped Rotor**
			Min / Max	**Min / Max**	**Min / Max**	**Min / Max**	**Min / Max**	**Min / Max**
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/	/	/
Max Cruise Speed (kts)	150 - 550	kts	/	/	/	/	/	/
Cruise Aero. Efficiency (L/D)	5 - 25	-	/	/	/	/	/	/
Wing Loading (W/S)	15 - 300	lb/ft^2	/	/	/	/	/	/
Flat Plate Drag Area (-)	1 - 9	Qual	/	/	/	/	/	/
			/	/	/	/	/	/
			/	/	/	/	/	/
			/	/	/	/	/	/

Engine Type

Major Functional Characteristics (Inputs)	Qualitative/ Quantitative & Range For Consideration	Units	Functional Options:					
			Turboshaft/ Turboprop	**Turbofan**	**Turbojet**	**Hybrid Electric**		
			Min / Max	**Min / Max**	**Min / Max**	**Min / Max**		
Subsystem Empty Weight (-)	1 - 9	Qual	/	/	/	/		
Specific Fuel Consump. (SFC)	1 - 9	Qual	/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		
			/	/	/	/		

APPENDIX B

SELECTED HISTORICAL DATA

The following table of approximately 150 historical aircraft and their characteristics was selected for their varying architectures and relevance to the example VTOL X-Plane example problem. The following data was combined with expert evaluation of functional system architecture options to influence the input data for many of the fuzzy system models generated in this research as described in Section 3.2.

Aircraft	VL System	VL Prop	VL Drive	VL Tech	FWD Prop	FWD Drive	FWD Sys	Wing	Engine Sys	Num FWD	GWT (lbs)	Pinst (hp)	FWD Thrust (lbs)	VL System Characteristics			Wing Characteristics				Power/Thrust Characteristics			
														Rotor Area (sqft)	Solidity	DiskLoading (lb/sqft)	Wingspan (ft)	Wing Area (sqft)	Wing Loading (lb/sqft)	Aspect Ratio	T/P (lb/hp)	T/W Vertical	T/W FWD	T/W Max
A380	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	1300000	-	304000	-	-	-	261.6	9100.0	142.9	7.5	-	0.234	0.2338462	0.234
Aerospatiale 332 UT	SMR	Rotor	Shaft	None	-	-	-	-	-	-	18960	-	-	2057	0.098	9.2	-	-	-	-	-	-	-	-
Aerospatiale 532 GE	SMR	Rotor	Shaft	None	-	-	-	-	-	-	19842	-	-	2057	0.098	9.6	-	-	-	-	-	-	-	-
Aerospatiale 550 GE	SMR	Rotor	Shaft	None	-	-	-	-	-	-	4960	-	-	966	0.063	5.1	-	-	-	-	-	-	-	-
Aerospatiale 565N GE	SMR	Rotor	Shaft	None	-	-	-	-	-	-	9370	-	-	1205	0.085	7.8	-	-	-	-	-	-	-	-
AeroVironment SkyTote	SMR	Rotor	Shaft	Tailsitter	Rotor	Shaft	Tilt	Conv	-	-	250	-	-	79	-	3.2	8	-	-	-	-	-	-	-
Alenia G-222/C-27	-	-	-	-	Thrust	Thrust	Fixed	Conv	-	-	61676	-	-	-	-	-	94.2	883.1	69.8	10.0	-	-	-	-
AMX	-	-	-	-	Thrust	Thrust	Fixed	Conv	-	-	28631	-	11023	-	-	-	29.1	225.8	126.8	3.8	-	0.385	0.385	0.385
Antonov An-124 Ruslan	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	893585	-	-	-	-	-	240.5	6764.1	132.1	8.6	-	-	-	-
Antonov An-22 Antheus	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	552412	-	-	-	-	-	211.3	3720.2	148.5	12.0	-	-	-	-
Antonov An-225, Mryia	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	6	1328171	-	-	-	-	-	290.0	9780.8	135.8	8.6	-	-	-	-
Antonov An-70	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	286271	-	-	-	-	-	144.6	2321.8	123.3	9.0	-	-	-	-
ASK 21	-	-	-	-	-	-	Glider	Conv	-	-	1323	-	-	-	-	-	55.8	193.2	6.8	16.1	-	-	-	-
Avro Vulcan	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	4	140000	-	44000	-	-	-	99.5	3554.0	39.4	2.8	-	0.314	0.314	0.314
B-17	-	-	-	-	Prop	Shaft	Fixed	Conv	Piston	4	54,000	4800	-	-	-	-	103.8	1420.0	38.0	7.6	11.3	-	-	-
B-36	-	-	-	-	Prop	Shaft	Fixed	Conv	Piston	6	262,500	22800	-	-	-	-	230.0	4772.0	55.0	11.1	11.5	-	-	-
B747	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	735000	-	204000	-	-	-	195.6	5500.0	133.6	7.0	-	0.278	0.278	0.278
BAe SeaHarrier Mk2	-	-	-	-	Thrust	Thrust	Fixed	Conv	-	-	23418	-	19249	-	-	-	25.3	179.8	130.3	3.6	-	0.822	0.822	0.822
Bell 209 SeaCobra	SMR	Rotor	Shaft	None	-	-	-	-	-	-	9998	-	-	1520	0.080	6.6	-	-	-	-	-	-	-	-
Bell 406/OH-58D AT	SMR	Rotor	Shaft	None	-	-	-	-	-	-	5512	-	-	962	0.057	5.7	-	-	-	-	-	-	-	-
Bell 407 GE	SMR	Rotor	Shaft	None	-	-	-	-	-	-	5004	-	-	962	0.064	5.2	-	-	-	-	-	-	-	-
Bell 412 UT	SMR	Rotor	Shaft	None	-	-	-	-	-	-	11596	-	-	1662	0.073	7.0	-	-	-	-	-	-	-	-
Bell 427 GE	SMR	Rotor	Shaft	None	-	-	-	-	-	-	6250	-	-	1076	0.061	5.8	-	-	-	-	-	-	-	-
Bell 533	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bell AH-1W SuperCobra	SMR	Rotor	Shaft	None	-	-	-	-	-	-	14771	-	-	1809	0.073	8.2	-	-	-	-	-	-	-	-
Bell Eagle Eye	Transverse	Rotor	Shaft	None	Rotor	Shaft	Tilt	Conv	-	-	1020	641	-	157	-	6.5	-	-	-	-	1.6	-	-	-
Bell HSL (1953)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	26500	2400	-	4170	-	6.4	-	-	-	-	11.0	-	-	-
Bell X-22	Quad	Ducted Fan	Shaft	None	Ducted Fan	Shaft	Tilt	Duel	-	-	17644	5068	-	154	-	114.6	-	39	452.4	-	3.5	-	-	-
Bell XV-15	Transverse	Rotor	Shaft	None	Rotor	Shaft	Tilt	Conv	-	-	13000	3100	-	982	0.089	13.2	57	-	-	-	4.2	-	-	-
Bell XV-3	Transverse	Rotor	Shaft	None	Rotor	Shaft	Tilt	Conv	-	-	4890	450	-	982	-	5.0	31	116	42.2	-	10.9	-	-	-
Bell/Agusta BA609	Transverse	Rotor	Shaft	None	Rotor	Shaft	Tilt	Conv	-	-	16,800	3880	-	1048	-	16.0	38.5	-	-	-	4.3	-	-	-
Bf-109	-	-	-	-	Prop	Shaft	Fixed	Conv	Turboprop	1	7495	1455	-	-	-	-	32.5	173.3	43.2	6.1	5.2	-	-	-
Boeing A160 Hummingbird	SMR	Rotor	Shaft	Variable RPM	-	-	-	-	-	-	6500	550	-	1018	-	6.4	-	-	-	-	11.8	-	-	-
Boeing B-747-400F	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	871310	-	-	-	-	-	211.4	5827.5	149.5	7.7	-	-	-	-
Boeing F/A 18E	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	-	65946	-	43920	-	-	-	44.7	499.2	132.1	4.0	-	0.666	0.666	0.666
Boeing KC-135A Stratolifter	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	315763	-	-	-	-	-	130.8	2431.7	129.9	7.0	-	-	-	-
Boeing Model 360 (1987)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	30500	8400	-	-	-	-	-	-	-	-	3.6	-	-	-
Boeing Vertol 107-II (1958)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Boeing Vertol XCH-62 (1970s)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	118000	24240	-	13260	-	8.9	-	-	-	-	4.9	-	-	-
Boeing X-48	-	-	-	-	Thrust	Thrust	Fixed	BWB	Turbojet	3	500	-	156	-	-	-	20.5	100.5	5.0	4.2	-	0.312	0.312	0.312
Boeing X-50 Dragonfly	SMR	Thrust	Tip Jet	Stopped	Thrust	Thrust	Fixed	Stopped	-	-	1422	-	-	113	-	12.6	6	9	158.0	-	-	-	-	-
BoeingVertol 114/CH-47D TW	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	54013	-	-	2828	0.085	19.1	-	-	-	-	-	-	-	-
Bristol Belvedere (1952)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	19000	2930	-	3270	-	5.8	-	-	-	-	6.5	-	-	-
Burnelli CBY-3	-	-	-	-	Prop	Shaft	Fixed	BWB	Piston	2	27000	2400	-	-	-	-	85.5	1107.0	24.4	6.6	11.3	-	-	-
Burnelli UB-14	-	-	-	-	Prop	Shaft	Fixed	BWB	Piston	2	17500	1500	-	-	-	-	71.0	686.0	25.5	7.3	11.7	-	-	-
Canadair CL-84	Transverse	Propeller	Shaft	None	Propeller	Shaft	Tilt	Tilt	-	-	12600	3000	-	308	-	40.9	34.3	233.3	54.0	-	4.2	-	-	-
Carter PAV	SMR	Rotor	-	Gyrodyne	-	-	-	Conv	-	-	3800	300	-	1590	-	2.4	45	-	-	-	12.7	-	-	-
CarterCopter	SMR	Rotor	-	Gyrodyne	-	-	-	Conv	-	-	4200	350	-	804	-	5.2	-	-	-	-	12.0	-	-	-
Cessna 152	-	-	-	-	Prop	Shaft	Fixed	Conv	Piston	1	1,670	110	-	-	-	-	33.3	160.0	10.4	6.9	15.2	-	-	-
CH-46 Sea Knight (1960) (CH46E)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	24300	3740	-	3927	-	6.2	-	-	-	-	6.5	-	-	-
CH-47 Chinook (1961) (CH47F)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	50000	9466	-	5600	-	8.9	-	-	-	-	5.3	-	-	-
CH-53E	SMR	Rotor	Shaft	None	-	-	-	-	-	-	73500	13140	-	4900	-	15.0	-	-	-	-	5.6	-	-	-
Ching-Kuo (Taiwan)	-	-	-	-	Thrust	Thrust	Fixed	Conv	-	-	27002	-	19226	-	-	-	28.0	261.1	103.4	3.0	-	0.712	0.712	0.712
Convair B-58 Hustler	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	4	67871	-	62400	-	-	-	56.8	1542.0	44.0	2.1	-	0.919	0.9193912	0.919
Convair F-102 Delta Dagger	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	1	24500	-	17200	-	-	-	38.1	661.5	37.0	2.2	-	0.702	0.7020408	0.702
Convair XFY-1 Pogo	Tailsitter	Propeller	Shaft	-	Propeller	Shaft	Tilt	Delta	-	-	14250	5100	-	201	-	70.9	27.5	355	40.1	-	2.8	-	-	-
Dassault Mirage 2K	-	-	-	-	Thrust	Thrust	Fixed	Delta	-	-	37569	-	21865	-	-	-	30.0	442.0	85.0	2.0	-	0.582	0.582	0.582
DC-3	-	-	-	-	Prop	Shaft	Fixed	Conv	Turboprop	2	25,199	2200	-	-	-	-	95.2	987.0	25.5	9.2	11.5	-	-	-
Doak VZ -4	Transverse	Ducted Fan	Shaft	None	Ducted Fan	Shaft	Tilt	Conv	-	-	3200	1000	-	39	-	81.5	25.5	96	33.3	-	3.2	-	-	-
Douglas C-133B Cargomaster	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	218076	-	-	-	-	-	179.7	2668.5	81.7	12.1	-	-	-	-
Enstrom 480 LC	SMR	Rotor	Shaft	None	-	-	-	-	-	-	2866	-	-	804	0.047	3.6	-	-	-	-	-	-	-	-
Eurocopter BO 105	SMR	Rotor	Shaft	None	-	-	-	-	-	-	5512	-	-	819	0.070	6.7	-	-	-	-	-	-	-	-
Eurocopter EC 120B	SMR	Rotor	Shaft	None	-	-	-	-	-	-	3748	-	-	845	0.050	4.4	-	-	-	-	-	-	-	-
Eurocopter EC 365N	SMR	Rotor	Shaft	None	-	-	-	-	-	-	9370	-	-	1205	0.085	7.8	-	-	-	-	-	-	-	-
Eurocopter X3[89]	SMR	Rotor	Shaft	Variable RPM	-	-	-	Conv	-	-	11464	4540	-	1342	-	8.5	?	-	-	-	2.5	-	-	-
Eurofighter 2000	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	-	50665	-	41140	-	-	-	35.9	537.8	94.2	2.4	-	0.812	0.812	0.812
F-104	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	1	20649	-	15600	-	-	-	21.8	196.1	105.3	2.4	-	0.755	0.755	0.755
Fairchild A-10	-	-	-	-	Thrust	Thrust	Fixed	Conv	-	-	49931	-	18075	-	-	-	57.5	505.8	98.7	6.5	-	0.362	0.362	0.362
Fairey Rotodyne	SMR	Rotor	Tip Jet	Gyrodyne	-	-	-	Conv	-	-	33000	2800	-	6362	-	5.2	46.6	475	69.5	-	11.8	-	-	-
Fairey Ultra-light Helicopter	SMR	Rotor	Tip Jet	None	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Fiat 7002	SMR	Rotor	Tip Jet	None	-	-	-	-	-	-	3086	542	-	1217	-	2.5	-	-	-	-	5.7	-	-	-
Focke-Achgelis Fa 223 (1941)	Transverse	Rotor	Shaft	None	-	-	-	-	-	-	8510	1000	-	2433	-	3.5	-	-	-	-	8.5	-	-	-
Focke-Wulf Fw 61 (1936)	Transverse	Rotor	Shaft	None	-	-	-	-	-	-	2094	160	-	831	-	2.5	-	-	-	-	13.1	-	-	-
General Dyn F111/F	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	-	-	-	-	-	-	-	32.0	-	-	-	-	-	0.502	-
General Dynamics YF-16	-	-	-	-	Thrust	Thrust	Fixed	Delta	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-

														VL System Characteristics			Wing Characteristics				Power/Thrust Characteristics			
Aircraft	VL System	VL Prop	VL Drive	VL Tech	FWD Prop	FWD Drive	FWD Sys	Wing	Engine Sys	Num FWD	GWT (lbs)	Pinst (hp)	FWD Thrust (lbs)	Rotor Area (sqft)	Solidity	DiskLoading (lb/sqft)	Wingspan (ft)	Wing Area (sqft)	Wing Loading (lb/sqft)	Aspect Ratio	T/P (lb/hp)	T/W Vertical	T/W FWD	T/W Max
Grumman F-14A	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	2	-	-	-	-	-	-	38.2	-	-	-	-	-	0.57	-
Gulfstream II	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	2	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Gulfstream III	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	2	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Handley Page HP.115	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	1	5050	-	1900	-	-	-	20.0	432.0	11.7	0.9	-	0.376	0.376	0.376
Heinkel Lerche	Tailsitter	Propeller	Shaft	-	Propeller	Shaft	Tilt	Annular	-	-	12346	2414	-	133	-	93.0	13	130	95.0	-	5.1	-	-	-
Hiller x-18	Transverse	Propeller	Shaft	None	Propeller	Shaft	Tilt	Tilt	-	-	33000	11000	-	616	-	53.6	48	336	98.2	-	3.0	-	-	-
Horten Ho 229	-	-	-	-	Thrust	Thrust	Fixed	BWB	Turbojet	2	6912	-	3912	-	-	-	55.0	540.4	12.8	5.6	-	0.566	0.566	0.566
Hughes XH-17	SMR	Rotor	Tip Jet	None	-	-	-	-	-	-	43500	-	-	53093	-	0.8	-	-	-	-	-	-	-	-
Ikarus C42	-	-	-	-	Prop	Shaft	Fixed	Conv	Piston	1	1041	100	-	-	-	-	31.0	134.5	7.7	7.1	10.4	-	-	-
Ilyushin Il-76MD	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	375043	-	-	-	-	-	165.7	3229.5	116.1	8.5	-	-	-	-
Ilyushin Il-96T	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	596399	-	-	-	-	-	189.2	4220.1	141.3	8.5	-	-	-	-
JD-2 Delta	-	-	-	-	Prop	Shaft	Fixed	Delta	Piston	1	1980	180	-	-	-	-	22.3	174.0	11.4	2.8	11.0	-	-	-
Jovair Sedan 4A (1963)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	2000	200	-	3041	-	0.7	-	-	-	-	10.0	-	-	-
Junkers G38	-	-	-	-	Prop	Shaft	Fixed	BWB	Piston	4	52911	2136	-	-	-	-	144.3	3100.0	17.1	6.7	24.8	-	-	-
Kaman Seasprite UT	SMR	Rotor	Shaft	None	-	-	-	-	-	-	13492	-	-	1612	0.218	8.4	-	-	-	-	-	-	-	-
Kamov Ka-22 (1959)	Transverse	Rotor	Shaft	None	Propeller	Shaft	Fixed	Conv	-	-	78,264	10848	-	8559	-	9.1	74	1130	69.3	-	7.2	-	-	-
Lockheed AH-56 Cheyenne	SMR	Rotor	Shaft	Variable RPM	-	-	-	-	-	-	25880	3925	-	2063	-	12.5	-	-	-	-	6.6	-	-	-
Lockheed C-130J Hercules	-	-	-	-	Prop	Prop	Fixed	Conv	Turboshaft	4	154543	-	-	-	-	-	132.6	1738.6	88.9	10.1	-	-	-	-
Lockheed C-141A	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Lockheed C-141B	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Lockheed C-141B StarLifter	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	342539	-	-	-	-	-	159.9	3228.6	106.1	7.9	-	-	-	-
Lockheed C-5B Galaxy	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	836784	-	-	-	-	-	222.7	6199.6	135.0	8.0	-	-	-	-
Lockheed F-117A	-	-	-	-	Thrust	Thrust	Fixed	BWB	Turbojet	2	31748	-	13080	-	-	-	43.3	551.6	57.6	3.4	-	0.412	0.412	0.412
Lockheed F-16C	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	1	42329	-	29588	-	-	-	31.0	300.4	140.9	3.2	-	0.699	0.699	0.699
Lockheed F-22A	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	2	59775	-	66769	-	-	-	44.5	838.6	71.3	2.4	-	1.117	1.117	1.117
Lockheed XFV-1	Tailsitter	Prop	Shaft	-	Prop	Shaft	Tilt	Delta	-	-	16221	5332	-	201	-	80.7	30	246	65.9	-	3.0	-	-	-
LTV XC-142	Transverse	Propeller	Shaft	None	Propeller	Shaft	Tilt	Tilt	-	-	42000	11400	-	804	-	52.2	67	534.5	78.6	-	3.7	-	-	-
MAPO MiG-29	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	-	49957	-	42913	-	-	-	37.3	470.9	106.1	3.0	-	0.859	0.859	0.859
MAPO MiG-31	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	-	89554	-	61166	-	-	-	44.2	583.0	153.6	3.4	-	0.683	0.683	0.683
McDonnell XV-1	SMR	Rotor	Tip Jet	Gyrodyne	-	-	-	Conv	-	-	5505	525	-	755	-	7.3	26	-	-	-	10.5	-	-	-
MD C-17A, Globemaster	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	585602	-	-	-	-	-	165.0	3802.1	154.0	7.2	-	-	-	-
MD KC-10A, Extender	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	4	584191	-	-	-	-	-	155.3	3859.6	151.4	6.3	-	-	-	-
MD-11F	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	3	610000	-	186000	-	-	-	169.5	3648.0	167.2	7.9	-	0.305	0.305	0.305
MD-500E LC	SMR	Rotor	Shaft	None	-	-	-	-	-	-	2998	-	-	548	0.067	5.5	-	-	-	-	-	-	-	-
Mil Mi-12 (1967)	Transverse	Rotor	Shaft	None	-	-	-	Conv	-	-	213848	26000	-	20774	-	10.3	220	-	-	-	8.2	-	-	-
Mil Mi-26 C	SMR	Rotor	Shaft	None	-	-	-	-	-	-	123459	22798	-	8657	0.146	14.3	-	-	-	-	5.4	-	-	-
Mil Mi-28 AT	SMR	Rotor	Shaft	None	-	-	-	-	-	-	25133	-	-	2501	0.124	10.0	-	-	-	-	-	-	-	-
Mitzubishi BK-117 GE	SMR	Rotor	Shaft	None	-	-	-	-	-	-	7385	-	-	1023	0.074	7.2	-	-	-	-	-	-	-	-
NAMC Q-5 (China)	-	-	-	-	Thrust	Thrust	Fixed	Conv	-	-	36409	-	19952	-	-	-	31.8	420.3	86.6	2.4	-	0.548	0.548	0.548
North American XB-70A	-	-	-	-	Thrust	Thrust	Fixed	Delta	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Northrop B-2	-	-	-	-	Thrust	Thrust	Fixed	Flying Wing	Turbofan	4	336000	-	69200	-	-	-	172.0	5140.0	65.4	5.8	-	0.206	0.206	0.206
Northrop N-1M	-	-	-	-	Prop	Shaft	Fixed	Flying Wing	Piston	2	3900	234	-	-	-	-	38.6	350.0	11.1	4.3	16.7	-	-	-
Northrop N-9M	-	-	-	-	Prop	Shaft	Fixed	Flying Wing	Piston	2	13946	600	-	-	-	-	60.0	490.0	28.5	7.3	23.2	-	-	-
Northrop YB-35	-	-	-	-	Prop	Shaft	Fixed	Flying Wing	Piston	4	180000	12000	-	-	-	-	172.0	4000.0	45.0	7.4	15.0	-	-	-
Northrop YB-49	-	-	-	-	Thrust	Thrust	Fixed	Flying Wing	Turbojet	8	133559	-	32000	-	-	-	172.0	4000.0	33.4	7.4	-	0.240	0.240	0.240
Paul P P-111	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	1	10127	-	5100	-	-	-	33.5	290.0	34.9	3.9	-	0.504	0.504	0.504
Piasecki H-21 (1953) (CH-21C)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	15200	1425	-	3041	-	5.0	-	-	-	-	10.7	-	-	-
Piasecki H-25/HUP Retriever (1952)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	6100	1100	-	1924	-	3.2	-	-	-	-	5.5	-	-	-
Pitcairn PCA-2	SMR	Rotor	-	Gyrodyne	Propeller	FALSE	Fixed	Conv	-	-	3000	330	-	1590	-	1.9	30	-	-	-	9.1	-	-	-
Platt-LePage XR-1 (1941)	Transverse	Rotor	Shaft	None	-	-	-	Conv	-	-	4730	450	-	1559	-	3.0	65	-	-	-	10.5	-	-	-
Robinson R-22	SMR	Rotor	Shaft	None	-	-	-	-	-	-	1370	124	-	497	-	2.8	-	-	-	-	11.0	-	-	-
Ryan X-13 Vertijet	Tailsitter	Thrust	Thrust	-	Thrust	Thrust	Tilt	Delta	-	-	6730	-	-	-	-	-	21	191	35.2	-	-	1.48	-	-
SAAB Gripen JS39	-	-	-	-	Thrust	Thrust	Fixed	Delta	Turbojet	-	-	-	-	-	-	-	27.6	217.6	-	3.5	-	-	0.644	-
SAAB Viggen JA37	-	-	-	-	Thrust	Thrust	Fixed	Delta	-	-	-	-	-	-	-	-	34.8	-	75.8	-	-	-	0.764	-
SATIC A300-600, Beluga	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbofan	-	244660	-	-	-	-	-	147.1	2810.7	87.0	7.7	-	-	-	-
Sikorsky S-72	SMR	Rotor	Shaft	Stopped	FALSE	FALSE	Fixed	Conv	-	-	24000	2800	-	2043	-	11.7	62	2042.8	11.7	-	8.6	-	-	-
Sikorsky X2	Coaxial	Rotor	Shaft	Variable RPM	-	-	-	-	-	-	6000	1800	-	548	-	10.9	-	-	-	-	3.3	-	-	-
SNECMA Coléoptère	Tailsitter	Propeller	Thrust	-	Propeller	Thrust	Tilt	Annualr	-	-	6614	-	-	-	-	-	15	-	-	-	-	1.24	-	-
Stout Batwing	-	-	-	-	Prop	Shaft	Fixed	BWB	Piston	1	1800	150	-	-	-	-	20.0	480.0	3.8	0.8	12.0	-	-	-
Sud-Ouest Djinn	SMR	Rotor	Tip Jet	None	-	-	-	-	-	-	1764	240	-	1023	-	1.7	-	-	-	-	7.4	-	-	-
Sukhoi Su-27	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	-	72620	-	54828	-	-	-	48.2	666.5	109.0	3.5	-	0.755	0.755	0.755
Sukhoi Su-34	-	-	-	-	Thrust	Thrust	Fixed	Conv	Turbojet	-	#VALUE!	-	-	-	-	-	48.2	-	146.4	-	-	-	0.631	-
Tornado ADV	-	-	-	-	Thrust	Thrust	Fixed	Conv	-	-	#VALUE!	-	-	-	-	-	28.2	-	-	-	-	-	0.698	-
V-22 Osprey	Transverse	Rotor	Shaft	Variable RPM	Rotor	Shaft	Tilt	Conv	-	-	47500	12300	-	2268	0.1201	20.9	45.83	301.4	157.6	-	3.9	-	-	-
Vans RV-4	-	-	-	-	Prop	Shaft	Fixed	Conv	Piston	1	1500	180	-	-	-	-	23.0	110.0	13.6	4.8	8.3	-	-	-
Vertol VZ-2	Transverse	Propeller	Shaft	None	Propeller	Shaft	Tilt	Tilt	-	-	4000	700	-	142	-	28.2	24.9	-	-	-	5.7	-	-	-
VFW-Fokker H3	SMR	Rotor	Tip Jet	None	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Westland Lynx	SMR	Rotor	Shaft	Variable RPM	-	-	-	-	-	-	11750	2480	-	1385	-	8.5	-	-	-	-	4.7	-	-	-
XV-5	FIW	Ducted Fan	Tip Blown	None	Thrust	Thrust	Fixed	Delta	-	-	12300	7750	-	170	-	72.2	30	360.3	34.1	-	1.6	0.391	-	-
Yakovlev Yak-24 (1952)	Tandem	Rotor	Shaft	None	-	-	-	-	-	-	34898	2536	-	7454	-	4.7	-	-	-	-	13.8	-	-	-
Eurocopter X3	SMR	Rotor	Shaft	Variable RPM	Propeller	Shaft	Fixed	Conv	Turboshaft	2	11464	4540	-	1342	-	8.5	-	-	-	-	2.5	-	-	-

APPENDIX C

FUZZY SYSTEM DATA

The included Fuzzy Control Language files content below was used to save and regenerate trained fuzzy rule-based systems. Input and output membership functions are listed based on the input and output ranges as piecewise linear functions with $[x, y]$ pairs. The rule block is then listed below the variable definitions.

C.0.1 FRBS Generation File for L/D System

```
FUNCTION_BLOCK

    VAR_INPUT
        SYSTEM_f:      REAL; (* RANGE(1 .. 9) *)
        WING_LoD:     REAL; (* RANGE(5 .. 25) *)
    END_VAR

    VAR_OUTPUT
        sys_LoD: REAL; (* RANGE(5 .. 25) *)
   END_VAR

    FUZZIFY WING_LoD
        TERM VeryLow := (0, 0) (5, 1) (10, 0) ;
        TERM Low :=     (5, 0) (10, 1) (15, 0) ;
        TERM Med := (10, 0) (15, 1) (20, 0) ;
        TERM High := (15, 0) (20, 1) (25, 0) ;
        TERM VeryHigh := (20,0) (25,1) (30,0) ;
    END_FUZZIFY

    FUZZIFY SYSTEM_f
        TERM Poor := (-2.2, 0) (1, 1) (4.2, 0) ;
        TERM Med := (1.8, 0) (5, 1) (8.2, 0) ;
        TERM Good := (5.8, 0) (9, 1) (12.2, 0) ;
    END_FUZZIFY

    DEFUZZIFY sys_LoD
        TERM VeryLow := (1, 0) (5, 1) (9, 1) (13, 0) ;
        TERM Low :=     (5, 0) (9, 1) (13, 1) (17, 0) ;
        TERM Med := (9, 0) (13, 1) (17, 1) (21, 0) ;
        TERM High := (13, 0) (17, 1) (21, 1) (25, 0) ;
        TERM VeryHigh := (17, 0) (21, 1) (25, 1) (29, 0) ;
        ACCU:MAX;
        METHOD: COGS;(*MoM;*)
        DEFAULT := 0;
    END_DEFUZZIFY

    RULEBLOCK
        AND:MIN;
        OR:MAX;
        ACT:MIN;
        (*ACCU:MAX;*)
```

```
        RULE 1:   IF (WING_LoD IS VeryLow) AND (SYSTEM_f IS Poor) OR (SYSTEM_f IS Med
            ) OR (SYSTEM_f IS Good) THEN (sys_LoD IS VeryLow)
        RULE 2:   IF (WING_LoD IS Low) AND (SYSTEM_f IS Poor) THEN (sys_LoD IS
            VeryLow)
        RULE 3:   IF (WING_LoD IS Low) AND (SYSTEM_f IS Med) THEN (sys_LoD IS Low)
        RULE 4:   IF (WING_LoD IS Low) AND (SYSTEM_f IS Good) THEN (sys_LoD IS Low)
        RULE 5:   IF (WING_LoD IS Med) AND (SYSTEM_f IS Poor) THEN (sys_LoD IS Low)
        RULE 6:   IF (WING_LoD IS Med) AND (SYSTEM_f IS Med) THEN (sys_LoD IS Med)
        RULE 7:   IF (WING_LoD IS Med) AND (SYSTEM_f IS Good) THEN (sys_LoD IS Med)
        RULE 8:   IF (WING_LoD IS High) AND (SYSTEM_f IS Poor) THEN (sys_LoD IS Med)
        RULE 9:   IF (WING_LoD IS High) AND (SYSTEM_f IS Med) THEN (sys_LoD IS Med)
        RULE 10:  IF (WING_LoD IS High) AND (SYSTEM_f IS Good) THEN (sys_LoD IS High)
        RULE 11:  IF (WING_LoD IS VeryHigh) AND (SYSTEM_f IS Poor) THEN (sys_LoD IS
            High)
        RULE 12:  IF (WING_LoD IS VeryHigh) AND (SYSTEM_f IS Med) THEN (sys_LoD IS
            High)
        RULE 13:  IF (WING_LoD IS VeryHigh) AND (SYSTEM_f IS Good) THEN (sys_LoD IS
            VeryHigh)

    END_RULEBLOCK

END_FUNCTION_BLOCK
```

C.0.2 FRBS Generation File for Empty Weight System

```
FUNCTION_BLOCK

        VAR_INPUT
            VL_SYS_TECH_phi:      REAL; (* RANGE(-1.80390625 .. 13.796875)
            WING_SYS_TYPE_LD:      REAL; (* RANGE(-2.962890625 .. 34.26171875)
            VL_SYS_TECH_LD:     REAL; (* RANGE(-5.544921875 .. 36.865234375)
            FWD_SYS_DRV_eta_d:      REAL; (* RANGE(0.328759765625 .. 1.15986328125)
            FWD_SYS_TYPE_phi:      REAL; (* RANGE(-2.0234375 .. 10.70078125)
            VL_SYS_TECH_f:     REAL; (* RANGE(-2.23125 .. 13.49609375)
            FWD_SYS_PROP_eta_p:      REAL; (* RANGE(0.488671875 .. 1.207109375)
            VL_SYS_TYPE_phi:      REAL; (* RANGE(-1.6609375 .. 11.66328125)
            VL_SYS_TECH_w:     REAL; (* RANGE(-47.28515625 .. 170.288085938)
            VL_SYS_TYPE_w:     REAL; (* RANGE(-67.6318359375 .. 185.390625)
            WING_SYS_TYPE_f:      REAL; (* RANGE(-3.21171875 .. 11.3734375)
            VL_SYS_PROP_w:     REAL; (* RANGE(-73.5205078125 .. 252.3046875)
            VL_SYS_TYPE_f:     REAL; (* RANGE(-2.8515625 .. 11.8203125)
            FWD_SYS_TYPE_TP:      REAL; (* RANGE(-10.048828125 .. 26.619140625)
            VL_SYS_PROP_phi:      REAL; (* RANGE(-1.1078125 .. 12.22890625)
        END_VAR

        VAR_OUTPUT
            sys_phi:     REAL; (* RANGE(-1.41375 .. 11.61890625)
        END_VAR

        FUZZIFY VL_SYS_TECH_phi
            TERM A1 := (0.81171875,mean) (0.871875,std) ;
            TERM A3 := (5.08046875,mean) (1.3859375,std) ;
            TERM A2 := (2.33359375,mean) (0.67109375,std) ;
            TERM A5 := (9.69296875,mean) (1.36796875,std) ;
            TERM A4 := (7.01640625,mean) (0.78828125,std) ;
        END_FUZZIFY

        FUZZIFY WING_SYS_TYPE_LD
            TERM A1 := (5.65625,mean) (2.873046875,std) ;
            TERM A3 := (14.47265625,mean) (2.572265625,std) ;
            TERM A2 := (10.9765625,mean) (1.982421875,std) ;
            TERM A5 := (26.2578125,mean) (2.66796875,std) ;
            TERM A4 := (20.87890625,mean) (2.849609375,std) ;
        END_FUZZIFY
```

```
FUZZIFY VL_SYS_TECH_LD
    TERM A1 := (3.30859375,mean) (2.951171875,std) ;
    TERM A3 := (12.9921875,mean) (2.330078125,std) ;
    TERM A2 := (8.43359375,mean) (1.908203125,std) ;
    TERM A5 := (26.65234375,mean) (3.404296875,std) ;
    TERM A4 := (19.69921875,mean) (1.5234375,std) ;
END_FUZZIFY

FUZZIFY FWD_SYS_DRV_eta_d
    TERM A1 := (0.482421875,mean) (0.051220703125,std) ;
    TERM A3 := (0.73876953125,mean) (0.045068359375,std) ;
    TERM A2 := (0.600390625,mean) (0.069384765625,std) ;
    TERM A5 := (1.01748046875,mean) (0.0474609375,std) ;
    TERM A4 := (0.8974609375,mean) (0.061376953125,std) ;
END_FUZZIFY

FUZZIFY FWD_SYS_TYPE_phi
    TERM A1 := (0.8921875,mean) (0.971875,std) ;
    TERM A3 := (4.159375,mean) (1.0203125,std) ;
    TERM A2 := (2.5625,mean) (0.78671875,std) ;
    TERM A5 := (7.9,mean) (0.93359375,std) ;
    TERM A4 := (5.903125,mean) (0.871875,std) ;
END_FUZZIFY

FUZZIFY VL_SYS_TECH_f
    TERM A1 := (1.27265625,mean) (1.16796875,std) ;
    TERM A3 := (5.11484375,mean) (0.775,std) ;
    TERM A2 := (3.52109375,mean) (1.1546875,std) ;
    TERM A5 := (9.32890625,mean) (1.3890625,std) ;
    TERM A4 := (7.06640625,mean) (0.9359375,std) ;
END_FUZZIFY

FUZZIFY FWD_SYS_PROP_eta_p
    TERM A1 := (0.62859375,mean) (0.046640625,std) ;
    TERM A3 := (0.8565625,mean) (0.0380859375,std) ;
    TERM A2 := (0.728515625,mean) (0.0479296875,std) ;
    TERM A5 := (1.03109375,mean) (0.058671875,std) ;
    TERM A4 := (0.931875,mean) (0.0627734375,std) ;
END_FUZZIFY

FUZZIFY VL_SYS_TYPE_phi
    TERM A1 := (0.72734375,mean) (0.79609375,std) ;
    TERM A3 := (4.21640625,mean) (1.32265625,std) ;
    TERM A2 := (2.50546875,mean) (1.18515625,std) ;
    TERM A5 := (8.03046875,mean) (1.2109375,std) ;
    TERM A4 := (5.86328125,mean) (1.35625,std) ;
END_FUZZIFY

FUZZIFY VL_SYS_TECH_w
    TERM A1 := (0.0,mean) (15.76171875,std) ;
    TERM A3 := (59.150390625,mean) (14.150390625,std) ;
    TERM A2 := (29.4140625,mean) (16.2744140625,std) ;
    TERM A5 := (129.287109375,mean) (13.6669921875,std) ;
    TERM A4 := (103.53515625,mean) (18.9697265625,std) ;
END_FUZZIFY

FUZZIFY VL_SYS_TYPE_w
    TERM A1 := (0.0,mean) (22.5439453125,std) ;
    TERM A3 := (65.390625,mean) (12.4072265625,std) ;
    TERM A2 := (35.44921875,mean) (15.1171875,std) ;
    TERM A5 := (123.8671875,mean) (20.5078125,std) ;
    TERM A4 := (93.28125,mean) (14.794921875,std) ;
END_FUZZIFY

FUZZIFY WING_SYS_TYPE_f
    TERM A1 := (0.86171875,mean) (1.3578125,std) ;
    TERM A3 := (4.37578125,mean) (0.78359375,std) ;
    TERM A2 := (2.25234375,mean) (0.7359375,std) ;
```

```
    TERM A5 := (7.29765625,mean) (1.35859375,std) ;
    TERM A4 := (5.82734375,mean) (0.63125,std) ;
END_FUZZIFY

FUZZIFY VL_SYS_PROP_w
    TERM A1 := (0.0,mean) (24.5068359375,std) ;
    TERM A3 := (76.69921875,mean) (20.1708984375,std) ;
    TERM A2 := (36.533203125,mean) (25.25390625,std) ;
    TERM A5 := (178.740234375,mean) (24.521484375,std) ;
    TERM A4 := (126.73828125,mean) (20.80078125,std) ;
END_FUZZIFY

FUZZIFY VL_SYS_TYPE_f
    TERM A1 := (1.08359375,mean) (1.31171875,std) ;
    TERM A3 := (5.11015625,mean) (1.37109375,std) ;
    TERM A2 := (2.34140625,mean) (1.271875,std) ;
    TERM A5 := (9.10859375,mean) (0.90390625,std) ;
    TERM A4 := (7.59921875,mean) (1.3,std) ;
END_FUZZIFY

FUZZIFY FWD_SYS_TYPE_TP
    TERM A1 := (0.0,mean) (3.349609375,std) ;
    TERM A3 := (8.89453125,mean) (2.28125,std) ;
    TERM A2 := (5.1328125,mean) (2.94921875,std) ;
    TERM A5 := (19.0078125,mean) (2.537109375,std) ;
    TERM A4 := (12.90234375,mean) (2.4609375,std) ;
END_FUZZIFY

FUZZIFY VL_SYS_PROP_phi
    TERM A1 := (1.2171875,mean) (0.775,std) ;
    TERM A3 := (4.55,mean) (1.14453125,std) ;
    TERM A2 := (2.6140625,mean) (1.2109375,std) ;
    TERM A5 := (8.3921875,mean) (1.27890625,std) ;
    TERM A4 := (6.8484375,mean) (1.09453125,std) ;
END_FUZZIFY

DEFUZZIFY sys_phi
    TERM A1 := (1.0303125,mean) (0.8146875,std) ;
    TERM A0 := (0.2784375,mean) (0.38203125,std) ;
    TERM A3 := (3.946875,mean) (0.79171875,std) ;
    TERM A2 := (2.619375,mean) (0.7115625,std) ;
    TERM A5 := (6.555,mean) (0.7378125,std) ;
    TERM A4 := (4.9546875,mean) (0.5953125,std) ;
    TERM A7 := (9.3721875,mean) (0.4321875,std) ;
    TERM A6 := (7.8440625,mean) (0.688125,std) ;
    TERM A8 := (10.12125,mean) (0.49921875,std) ;
END_FUZZIFY

RULEBLOCK
     AND:MIN;
    OR:MAX;
    ACT:MIN;
    (*ACCU:MAX;*)

    RULE 0:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
        AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
        VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
        IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
        VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
        IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
        WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
    RULE 1:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
        AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
        VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
        IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
        VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
        IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
        WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
```

```
RULE 2:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 3:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 4:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 5:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 6:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 7:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 8:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 9:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 10:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
```

```
RULE 11:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 12:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 13:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 14:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 15:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 16:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
RULE 17:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 18:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 19:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
```

```
RULE 20:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 21:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 22:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 23:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 24:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 25:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 26:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 27:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 28:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
```

```
RULE 29:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 30:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 31:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 32:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 33:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 34:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 35:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 36:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 37:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
```

```
RULE 38:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 39:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 40:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
RULE 41:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 42:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 43:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A2) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 44:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 45:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 46:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
```

```
RULE 47:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 48:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A1);
RULE 49:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 50:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
RULE 51:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 52:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 53:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 54:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 55:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
```

```
RULE 56:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 57:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 58:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 59:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 60:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 61:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 62:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 63:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 64:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
```

```
RULE 65:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 66:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 67:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A6);
RULE 68:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 69:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A1);
RULE 70:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 71:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 72:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A2) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 73:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
```

```
RULE 74:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 75:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 76:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 77:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 78:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 79:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 80:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 81:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A2) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
RULE 82:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
```

```
RULE 83:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 84:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 85:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 86:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 87:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 88:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 89:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 90:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 91:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
```

```
RULE 92:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A6);
RULE 93:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 94:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 95:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 96:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 97:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 98:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 99:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 100:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
```

```
RULE 101:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A2) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 102:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 103:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 104:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 105:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 106:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 107:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 108:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 109:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
```

```
RULE 110:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 111:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
RULE 112:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
RULE 113:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 114:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 115:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 116:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 117:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 118:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
```

```
RULE 119:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 120:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 121:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
RULE 122:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 123:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 124:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 125:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 126:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 127:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
```

```
RULE 128:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 129:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 130:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 131:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 132:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 133:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 134:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 135:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 136:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
```

```
RULE 137:     IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 138:     IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 139:     IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 140:     IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 141:     IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A2) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A1);
RULE 142:     IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 143:     IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A6);
RULE 144:     IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 145:     IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
```

```
RULE 146:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 147:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 148:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 149:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 150:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 151:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 152:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 153:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A6);
RULE 154:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A6);
```

```
RULE 155:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 156:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 157:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 158:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A1) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 159:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 160:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 161:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 162:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 163:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
```

```
RULE 164:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 165:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 166:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 167:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
RULE 168:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 169:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 170:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 171:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 172:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A2);
```

```
RULE 173:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 174:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 175:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 176:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
RULE 177:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
RULE 178:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
RULE 179:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 180:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 181:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A2) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A4);
```

```
RULE 182:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 183:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A2);
RULE 184:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 185:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 186:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A4) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A1) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
RULE 187:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 188:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A3) AND (VL_SYS_TYPE_w IS A4) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A5);
RULE 189:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
    AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A4) AND (VL_SYS_PROP_w IS A1) AND (VL_SYS_TECH_LD
    IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
RULE 190:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
    AND (FWD_SYS_TYPE_TP IS A2) AND (FWD_SYS_TYPE_phi IS A3) AND (
    VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
    IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A3) AND (
    VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A4) AND (VL_SYS_TYPE_phi
    IS A5) AND (VL_SYS_TYPE_w IS A3) AND (WING_SYS_TYPE_LD IS A3) AND (
    WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A5);
```

```
        RULE 191:    IF (FWD_SYS_DRV_eta_d IS A2) AND (FWD_SYS_PROP_eta_p IS A1)
            AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
            VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
            IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
            VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
            IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
            WING_SYS_TYPE_f IS A3) THEN (sys_phi IS A3);
        RULE 192:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A2)
            AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
            VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
            IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
            VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
            IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A1) AND (
            WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);
        RULE 193:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
            AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
            VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
            IS A1) AND (VL_SYS_TECH_f IS A2) AND (VL_SYS_TECH_phi IS A3) AND (
            VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A2) AND (VL_SYS_TYPE_phi
            IS A5) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
            WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A3);
        RULE 194:    IF (FWD_SYS_DRV_eta_d IS A4) AND (FWD_SYS_PROP_eta_p IS A4)
            AND (FWD_SYS_TYPE_TP IS A3) AND (FWD_SYS_TYPE_phi IS A3) AND (
            VL_SYS_PROP_phi IS A3) AND (VL_SYS_PROP_w IS A3) AND (VL_SYS_TECH_LD
            IS A3) AND (VL_SYS_TECH_f IS A3) AND (VL_SYS_TECH_phi IS A2) AND (
            VL_SYS_TECH_w IS A3) AND (VL_SYS_TYPE_f IS A3) AND (VL_SYS_TYPE_phi
            IS A4) AND (VL_SYS_TYPE_w IS A1) AND (WING_SYS_TYPE_LD IS A2) AND (
            WING_SYS_TYPE_f IS A5) THEN (sys_phi IS A4);

    END_RULEBLOCK
END_FUNCTION_BLOCK
```

C.0.3 FRBS Generation File for Propulsive Efficiency System

```
FUNCTION_BLOCK

    VAR_INPUT
        FWD_SYS_eta_p:    REAL; (* RANGE(0.6 .. 1.0) *)
        FWD_DRV_eta_d:     REAL; (* RANGE(0.5 .. 1.0) *)
    END_VAR

    VAR_OUTPUT
        sys_etaP: REAL; (* RANGE(0.6 .. 1.0) *)
   END_VAR

    FUZZIFY FWD_SYS_eta_p
        TERM VeryLow  := (0.60, 0) (0.65, 1) (0.70, 0) ;
        TERM Low      := (0.65, 0) (0.74, 1) (0.83, 0) ;
        TERM Med      := (0.70, 0) (0.80, 1) (0.90, 0) ;
        TERM High     := (0.85, 0) (0.90, 1) (0.95, 0) ;
        TERM VeryHigh := (0.92, 0) (0.97, 1) (1.1, 0) ;
    END_FUZZIFY

    FUZZIFY FWD_DRV_eta_d
        TERM Low      := (0.50, 0) (0.70, 1) (0.84, 0) ;
        TERM Med      := (0.72, 0) (0.84, 1) (0.94, 0) ;
        TERM High     := (0.86, 0) (0.95, 1) (1.00, 0) ;
    END_FUZZIFY


    DEFUZZIFY sys_etaP
        TERM VeryLow  := (0.30, 0) (0.30, 1) (0.65, 1) (0.70, 0) ;
        TERM Low      := (0.55, 0) (0.60, 1) (0.75, 1) (0.80, 0) ;
        TERM Med      := (0.71, 0) (0.75, 1) (0.89, 1) (0.93, 0) ;
        TERM High     := (0.85, 0) (0.89, 1) (0.95, 1) (0.99, 0) ;
        TERM VeryHigh := (0.91, 0) (0.95, 1) (1.00, 1) (1.00, 0) ;
```

```
        ACCU:MAX;
        METHOD: COGS;(*MoM;*)
        DEFAULT := 0;
    END_DEFUZZIFY

    RULEBLOCK
        AND:MIN;
        OR:MAX;
        ACT:MIN;
        (*ACCU:MAX;*)

        RULE 1:   IF (FWD_DRV_eta_d IS Low) THEN (sys_etaP IS VeryLow)
        RULE 2:   IF (FWD_DRV_eta_d IS Med) AND (FWD_SYS_eta_p IS VeryLow) THEN (
            sys_etaP IS VeryLow)
        RULE 3:   IF (FWD_DRV_eta_d IS Med) AND (FWD_SYS_eta_p IS Low) THEN (sys_etaP
             IS VeryLow)
        RULE 4:   IF (FWD_DRV_eta_d IS Med) AND (FWD_SYS_eta_p IS Med) THEN (sys_etaP
             IS Low)
        RULE 5:   IF (FWD_DRV_eta_d IS Med) AND (FWD_SYS_eta_p IS High) THEN (
            sys_etaP IS Med)
        RULE 6:   IF (FWD_DRV_eta_d IS Med) AND (FWD_SYS_eta_p IS VeryHigh) THEN (
            sys_etaP IS High)
        RULE 7:   IF (FWD_DRV_eta_d IS High) AND (FWD_SYS_eta_p IS VeryLow) THEN (
            sys_etaP IS VeryLow)
        RULE 8:   IF (FWD_DRV_eta_d IS High) AND (FWD_SYS_eta_p IS Low) THEN (
            sys_etaP IS Low)
        RULE 9:   IF (FWD_DRV_eta_d IS High) AND (FWD_SYS_eta_p IS Med) THEN (
            sys_etaP IS Med)
        RULE 10:  IF (FWD_DRV_eta_d IS High) AND (FWD_SYS_eta_p IS High) THEN (
            sys_etaP IS High)
        RULE 10:  IF (FWD_DRV_eta_d IS High) AND (FWD_SYS_eta_p IS VeryHigh) THEN (
            sys_etaP IS VeryHigh)

    END_RULEBLOCK

END_FUNCTION_BLOCK
```

C.0.4 FRBS Generation File for Figure of Merit System *(Not Used in Framework)*

```
FUNCTION_BLOCK

        VAR_INPUT
            DATA_e_d:      REAL; (* RANGE(-0.129357421875 .. 0.43869140625)
            DATA_sigma:      REAL; (* RANGE(-0.0823984375 .. 0.566400390625)
            DATA_w:     REAL; (* RANGE(-56.2060546875 .. 220.456054688)
            DATA_eta:      REAL; (* RANGE(0.3059375 .. 1.23388671875)
        END_VAR

        VAR_OUTPUT
            sys_FoM:     REAL; (* RANGE(0.288481445312 .. 1.18575683594)
        END_VAR

        FUZZIFY DATA_e_d
            TERM A1 := (0.0,mean) (0.043119140625,std) ;
            TERM A3 := (0.176220703125,mean) (0.037541015625,std) ;
            TERM A2 := (0.101235351562,mean) (0.0375,std) ;
            TERM A5 := (0.32619140625,mean) (0.0375,std) ;
            TERM A4 := (0.251206054688,mean) (0.0375,std) ;
        END_FUZZIFY

        FUZZIFY DATA_sigma
            TERM A1 := (0.05,mean) (0.0441328125,std) ;
            TERM A3 := (0.256069335938,mean) (0.044515625,std) ;
            TERM A2 := (0.168586425781,mean) (0.0471953125,std) ;
            TERM A5 := (0.43486328125,mean) (0.043845703125,std) ;
```

```
    TERM A4 := (0.343552246094,mean) (0.0441328125,std) ;
END_FUZZIFY

FUZZIFY DATA_w
    TERM A1 := (0.0439453125,mean) (18.75,std) ;
    TERM A3 := (75.849609375,mean) (19.119140625,std) ;
    TERM A2 := (37.5366210938,mean) (18.75,std) ;
    TERM A5 := (163.959960938,mean) (18.83203125,std) ;
    TERM A4 := (126.467285156,mean) (19.078125,std) ;
END_FUZZIFY

FUZZIFY DATA_eta
    TERM A1 := (0.5,mean) (0.0646875,std) ;
    TERM A3 := (0.793701171875,mean) (0.0625,std) ;
    TERM A2 := (0.668725585937,mean) (0.0625,std) ;
    TERM A5 := (1.04638671875,mean) (0.0625,std) ;
    TERM A4 := (0.921411132812,mean) (0.06263671875,std) ;
END_FUZZIFY

DEFUZZIFY sys_FoM
    TERM A1 := (0.47623046875,mean) (0.0447978515625,std) ;
    TERM A0 := (0.4009375,mean) (0.0374853515625,std) ;
    TERM A3 := (0.63150390625,mean) (0.0374853515625,std) ;
    TERM A2 := (0.55181640625,mean) (0.0374853515625,std) ;
    TERM A5 := (0.78150390625,mean) (0.0399228515625,std) ;
    TERM A4 := (0.70650390625,mean) (0.0374853515625,std) ;
    TERM A7 := (0.95142578125,mean) (0.0374853515625,std) ;
    TERM A6 := (0.87642578125,mean) (0.0399228515625,std) ;
    TERM A8 := (1.07330078125,mean) (0.0374853515625,std) ;
END_FUZZIFY

RULEBLOCK
     AND:MIN;
    OR:MAX;
    ACT:MIN;
    (*ACCU:MAX;*)

    RULE 0:     IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS A1
        ) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
    RULE 1:     IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS A5
        ) AND (DATA_w IS A2) THEN (sys_FoM IS A4);
    RULE 2:     IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS A1
        ) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
    RULE 3:     IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS A2
        ) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
    RULE 4:     IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS A4
        ) AND (DATA_w IS A1) THEN (sys_FoM IS A3);
    RULE 5:     IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS A3
        ) AND (DATA_w IS A1) THEN (sys_FoM IS A4);
    RULE 6:     IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS A1
        ) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
    RULE 7:     IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS A4
        ) AND (DATA_w IS A1) THEN (sys_FoM IS A4);
    RULE 8:     IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS A2
        ) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
    RULE 9:     IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS A3
        ) AND (DATA_w IS A1) THEN (sys_FoM IS A4);
    RULE 10:     IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
        A2) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
    RULE 11:     IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
        A4) AND (DATA_w IS A2) THEN (sys_FoM IS A0);
    RULE 12:     IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
        A2) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
    RULE 13:     IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
        A4) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
    RULE 14:     IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
        A2) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
```

```
RULE 15:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A5) THEN (sys_FoM IS A3);
RULE 16:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 17:    IF (DATA_e_d IS A4) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A6);
RULE 18:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 19:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 20:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A5) THEN (sys_FoM IS A5);
RULE 21:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A1) THEN (sys_FoM IS A3);
RULE 22:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 23:    IF (DATA_e_d IS A5) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 24:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 25:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 26:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A4);
RULE 27:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 28:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A4);
RULE 29:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 30:    IF (DATA_e_d IS A3) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A6);
RULE 31:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 32:    IF (DATA_e_d IS A5) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A5) THEN (sys_FoM IS A1);
RULE 33:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 34:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 35:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 36:    IF (DATA_e_d IS A2) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A6);
RULE 37:    IF (DATA_e_d IS A2) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A6);
RULE 38:    IF (DATA_e_d IS A1) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 39:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 40:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A0);
RULE 41:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 42:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
RULE 43:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 44:    IF (DATA_e_d IS A5) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 45:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 46:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 47:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A5) THEN (sys_FoM IS A3);
RULE 48:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
```

```
RULE 49:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A4);
RULE 50:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A5) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 51:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 52:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A5) THEN (sys_FoM IS A2);
RULE 53:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A5) THEN (sys_FoM IS A5);
RULE 54:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 55:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 56:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A5) THEN (sys_FoM IS A5);
RULE 57:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 58:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 59:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 60:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 61:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 62:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A4);
RULE 63:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A5);
RULE 64:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 65:    IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 66:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 67:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A1) THEN (sys_FoM IS A4);
RULE 68:    IF (DATA_e_d IS A2) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 69:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 70:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 71:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A3);
RULE 72:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A5) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 73:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 74:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 75:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 76:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 77:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 78:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 79:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 80:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 81:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
RULE 82:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
```

```
RULE 83:    IF (DATA_e_d IS A2) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A6);
RULE 84:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A5) AND (DATA_w IS A4) THEN (sys_FoM IS A0);
RULE 85:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 86:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A2);
RULE 87:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 88:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A5) THEN (sys_FoM IS A2);
RULE 89:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 90:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A5) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 91:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A0);
RULE 92:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 93:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 94:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 95:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A1) THEN (sys_FoM IS A2);
RULE 96:    IF (DATA_e_d IS A1) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
RULE 97:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 98:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A5) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 99:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 100:    IF (DATA_e_d IS A5) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 101:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 102:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 103:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A3);
RULE 104:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 105:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 106:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 107:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 108:    IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 109:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 110:    IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
RULE 111:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A4);
RULE 112:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
RULE 113:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
RULE 114:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 115:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 116:    IF (DATA_e_d IS A1) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
```

```
RULE 117:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 118:    IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 119:    IF (DATA_e_d IS A1) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A5);
RULE 120:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 121:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 122:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A5) AND (DATA_w IS A1) THEN (sys_FoM IS A4);
RULE 123:    IF (DATA_e_d IS A5) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 124:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 125:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A2);
RULE 126:    IF (DATA_e_d IS A1) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 127:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 128:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 129:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
RULE 130:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A5) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 131:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 132:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 133:    IF (DATA_e_d IS A2) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A5) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 134:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A5) THEN (sys_FoM IS A1);
RULE 135:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 136:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 137:    IF (DATA_e_d IS A1) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 138:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
RULE 139:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A5) THEN (sys_FoM IS A5);
RULE 140:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 141:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 142:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A5);
RULE 143:    IF (DATA_e_d IS A5) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 144:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 145:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 146:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 147:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 148:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 149:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 150:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
```

```
RULE 151:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A1) THEN (sys_FoM IS A2);
RULE 152:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
RULE 153:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A1) THEN (sys_FoM IS A2);
RULE 154:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 155:    IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 156:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 157:    IF (DATA_e_d IS A1) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 158:    IF (DATA_e_d IS A5) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 159:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
RULE 160:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 161:    IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
RULE 162:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A0);
RULE 163:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 164:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A5) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 165:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 166:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 167:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 168:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 169:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 170:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A3);
RULE 171:    IF (DATA_e_d IS A5) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
RULE 172:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 173:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A3);
RULE 174:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A4);
RULE 175:    IF (DATA_e_d IS A1) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 176:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 177:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A0);
RULE 178:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 179:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 180:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 181:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 182:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 183:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A0);
RULE 184:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
```

```
RULE 185:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A1) AND (DATA_w IS A1) THEN (sys_FoM IS A1);
RULE 186:    IF (DATA_e_d IS A1) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 187:    IF (DATA_e_d IS A1) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A1);
RULE 188:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 189:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 190:    IF (DATA_e_d IS A5) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 191:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 192:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 193:    IF (DATA_e_d IS A4) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A4);
RULE 194:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 195:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 196:    IF (DATA_e_d IS A1) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 197:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A4) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
RULE 198:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A4) AND (DATA_w IS A1) THEN (sys_FoM IS A2);
RULE 199:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 200:    IF (DATA_e_d IS A3) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A3);
RULE 201:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A5) THEN (sys_FoM IS A1);
RULE 202:    IF (DATA_e_d IS A2) AND (DATA_eta IS A5) AND (DATA_sigma IS
    A3) AND (DATA_w IS A1) THEN (sys_FoM IS A5);
RULE 203:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A1);
RULE 204:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A0);
RULE 205:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A5) THEN (sys_FoM IS A4);
RULE 206:    IF (DATA_e_d IS A4) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
RULE 207:    IF (DATA_e_d IS A3) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A2);
RULE 208:    IF (DATA_e_d IS A2) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 209:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A5) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 210:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
RULE 211:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A4) THEN (sys_FoM IS A5);
RULE 212:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A1) AND (DATA_w IS A4) THEN (sys_FoM IS A4);
RULE 213:    IF (DATA_e_d IS A2) AND (DATA_eta IS A2) AND (DATA_sigma IS
    A2) AND (DATA_w IS A1) THEN (sys_FoM IS A2);
RULE 214:    IF (DATA_e_d IS A4) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A0);
RULE 215:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
    A2) AND (DATA_w IS A2) THEN (sys_FoM IS A1);
RULE 216:    IF (DATA_e_d IS A4) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A2) AND (DATA_w IS A3) THEN (sys_FoM IS A5);
RULE 217:    IF (DATA_e_d IS A2) AND (DATA_eta IS A3) AND (DATA_sigma IS
    A4) AND (DATA_w IS A3) THEN (sys_FoM IS A3);
RULE 218:    IF (DATA_e_d IS A3) AND (DATA_eta IS A4) AND (DATA_sigma IS
    A3) AND (DATA_w IS A3) THEN (sys_FoM IS A4);
```

```
        RULE 219:    IF (DATA_e_d IS A1) AND (DATA_eta IS A2) AND (DATA_sigma IS
            A3) AND (DATA_w IS A3) THEN (sys_FoM IS A2);
        RULE 220:    IF (DATA_e_d IS A3) AND (DATA_eta IS A1) AND (DATA_sigma IS
            A4) AND (DATA_w IS A4) THEN (sys_FoM IS A0);
        RULE 221:    IF (DATA_e_d IS A2) AND (DATA_eta IS A4) AND (DATA_sigma IS
            A1) AND (DATA_w IS A2) THEN (sys_FoM IS A5);
        RULE 222:    IF (DATA_e_d IS A5) AND (DATA_eta IS A2) AND (DATA_sigma IS
            A3) AND (DATA_w IS A2) THEN (sys_FoM IS A2);
        RULE 223:    IF (DATA_e_d IS A5) AND (DATA_eta IS A5) AND (DATA_sigma IS
            A4) AND (DATA_w IS A2) THEN (sys_FoM IS A5);

    END_RULEBLOCK

END_FUNCTION_BLOCK
```

REFERENCES

[1] Abraham, a., "Adaptation of fuzzy inference system using neural learning," *Fuzzy Systems Engineering*, vol. 83, pp. 53–83, 2005.

[2] Alcala, R., Casillas, J., Herrera, F., and Zwir, S., "Techniques for Learning and Tuning Fuzzy Rule-Based Systems for Linguistic Modeling and their Applications," pp. 1–47, 2000.

[3] AugustaWestland, "ERICA : THE EUROPEAN TILTROTOR DESIGN AND CRITICAL TECHNOLOGY PROJECTS," in *AIAA/ICAS International Air and Space Symposium and Exposition*, no. July, 2003.

[4] Augustine, M., Yadav, O. P., Jain, R., and Rathore, A. P. S., "Concept convergence process: A framework for improving product concepts," *Computers & Industrial Engineering*, vol. 59, pp. 367–377, Oct. 2010.

[5] Aya, Z., "A fuzzy AHP-based simulation approach to concept evaluation in a NPD environment," *IIE Transactions*, vol. 37, pp. 827–842, Sept. 2005.

[6] Baas, S. and Kwakernaak, H., "Rating and ranking of multiple-aspect alternatives using fuzzy sets," *Automatica*, vol. 13, pp. 47–58, 1977.

[7] Babuska, R., "Fuzzy Systems, Modeling and Identification," 2001.

[8] Baker, A. P. and Mavris, D. N., "Assessing the simultaneous impact of requirements, vehicle characteristics, and technologies during aircraft design," in *AIAA Aerospace Sciences Meeting and Exhibit 39th Reno NV UNITED STATES 811 Jan 2001*, vol. AIAA 01-05, p. 10, 2001.

[9] Bandte, O., *A Probabilistic Multi-Criteria Decision Making Technique for Conceptual and Preliminary Aerospace Systems Design.* PhD thesis, Georgia Institute of Technology, 2000.

[10] Bandte, O., Mavris, D. N., and DeLaurentis, D. a., "Viable Designs Through a Joint Probabilistic Estimation Technique," *SAE Technical Papers*, 1999.

[11] Banuelas, R. and Antony, J., "Modified analytic hierarchy process to incorporate uncertainty and managerial aspects," *International Journal of Production Research*, vol. 42, no. 18, pp. 3851–3872, 2004.

[12] Bartie, K. B. H., Schneider, J., and Wilkerson, J., "Technology Needs for High Speed Rotorcraft (1)," tech. rep., Ames Research Center, Moffett Field, CA, 1991.

[13] Bevilacqua, M., Ciarapica, F. E., and Marchetti, B., “Development and test of a new fuzzy-QFD approach for characterizing customers rating of extra virgin olive oil,” *Food Quality and Preference*, vol. 24, pp. 75–84, Apr. 2012.

[14] Bilgiç, T. and Türken, I., “Measurement of membership functions: theoretical and empirical work,” *Fundamentals of fuzzy sets, Springer US*, pp. 195–227, 2000.

[15] Biltgen, P. T. and Mavris, D. N., “Technique for Concept Selection Using Interactive Probabilistic Multiple Attribute Decision Making,” *Journal of Aerospace Computing, Information, and Communication*, vol. 6, pp. 51–67, Jan. 2009.

[16] Bortolan, G. and Degani, R., “A Review of Some Methods for Ranking Fuzzy Subsets,” *Fuzzy sets and Systems*, vol. 15, pp. 1–19, 1985.

[17] Buckley, J., “Ranking Alternatives Using Fuzzy Numbers,” *Fuzzy sets and systems*, vol. 15, pp. 21–31, 1985.

[18] Buonanno, M. and Mavris, D., “A New Method for Aircraft Concept Selection Using Multicriteria Interactive Genetic Algorithms,” in *43rd AIAA Aerospace Sciences Meeting and Exhibit*, no. January, (Reston, Virigina), pp. 1–12, American Institute of Aeronautics and Astronautics, Jan. 2005.

[19] Buonanno, M. A. and Mavris, D. N., “Aerospace Vehicle Concept Selection Using Parallel, Variable Fidelity Genetic Algorithms,” in *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, no. September, (Albany, New York), 2004.

[20] Carter Aviation, “Carter Aviation Technologies,” 2014.

[21] Chai, Y. C. Y., Jia, L. J. L., and Zhang, Z. Z. Z., “Mamdani Model Based Adaptive Neural Fuzzy Inference System and its Application in Traffic Level of Service Evaluation,” *2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 4, no. 3, pp. 739–746, 2009.

[22] Chameau, J.-L. and Santamarina, J. C., “Membership functions I: Comparing methods of measurement,” *International Journal of Approximate Reasoning*, vol. 1, no. 3, pp. 287–301, 1987.

[23] Chan, L.-K. and Wu, M.-L., *Quality function deployment: A literature review*, vol. 143. Dec. 2002.

[24] Chen, S., Wang, S., and Chang, S., “Some Properties of Graded Mean Integration Representation of LR Type Fuzzy Numbers,” *Tamsui Oxford Journal of Mathematical . . .*, vol. 22, no. 2, pp. 185–208, 2006.

[25] Chen, S. and Nikolaidis, E., "Comparison of probabilistic and fuzzy set methods for designing under uncertainty," *American Institute of Aeronautics and Astronautics*, no. AIAA-99-1579, 1999.

[26] Churchill, A. W., Husbands, P., and Philippides, A., "Multi-objectivization of the Tool Selection Problem on a Budget of Evaluations," pp. 600–614, 2013.

[27] Cingolani, P. and Alcala-Fdez, J., "jFuzzyLogic: a Java Library to Design Fuzzy Logic Controllers According to the Standard for Fuzzy Control Programming," *International Journal of Computational Intelligence Systems*, vol. 6, pp. 61–75, 2013.

[28] Cordón, O., "A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems," *International Journal of Approximate Reasoning*, vol. 52, no. 6, pp. 894–913, 2011.

[29] Dalkey, N. and Helmer, O., "An Experimental Application of the Delphi Method to the Use of Experts," *Management Science*, vol. 9, no. 3, pp. 458–467, 1963.

[30] DARPA, "Broad Agency Announcement - Vertical Take-Off and Landing Experimental Aircraft (VTOL X-Plane)," *Tactical Technology Office (TTO)*, no. DARPA-BAA-13-19.

[31] DARPA, "Briefing Prepared for the VTOL X-Plane Proposers Day Meeting," 2013.

[32] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[33] DeTore, J. B. T. and Conway, S., "Technology needs for high-speed rotorcraft (3)," tech. rep., Ames Research Center, Moffett Field, CA, 1991.

[34] Dieter, G. E., *Engineering Design.* McGraw-Hill Higher Education, 5th ed., 2012.

[35] Dong, Y., Griendling, K., and Mavris, D., "An Uncertainty-Based QFD Framework for Aircraft Conceptual Design," in *AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, no. January, (Grapevine, Texas), pp. 1–16, American Institute of Aeronautics and Astronautics, 2013.

[36] Drezewski, R. and Siwi, "Techniques for Maintaining Population Diversity in Classical and Agent-Based Multi-objective Evolutionary Algorithms," *Computational Science - ICCS 2007*, pp. 904–911, 2007.

[37] Dubois, D. and Prade, H., *Fuzzy Sets and Systems: Theory and Applications*, vol. 27. New York: Academic Press, June 1988.

[38] Dubois, D. and Prade, H., "On the use of aggregation operations in information fusion processes," *Fuzzy Sets and Systems*, vol. 142, no. 1, pp. 143–161, 2004.

[39] Dutta, P., Boruah, H., and Ali, T., "Fuzzy Arithmetic with and without using α -cut method : A Comparative Study," *International Journal of Latest Trends in Computing*, vol. 2, no. 1, pp. 99–107, 2011.

[40] Engler, W. O., Biltgen, P. T., and Mavris, D. N., "Concept Selection Using an Interactive Reconfigurable Matrix of Alternatives (IRMA)," in *45th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, Nevada), 2007.

[41] Fernandez, M. G., "Decision Support in Concurrent Engineering - The Utility-Based Selection Decision Support Problem," *Concurrent Engineering*, vol. 13, pp. 13–27, Mar. 2005.

[42] Floros, M. W. and Johnson, W., "Stability and Control Analysis of the Slowed-Rotor Compound Helicopter Configuration," tech. rep., US Army Research Laboratory,Langley Research Center, Hampton, VA, 2005.

[43] Fullér, R., *Introduction to Neuro-Fuzzy Systems*. Springer, 2000.

[44] Fullér, R. and Majlender, P., "On interactive fuzzy numbers," *Fuzzy Sets and Systems*, vol. 143, no. 2004, pp. 355–369, 2004.

[45] Gerdes, R., "Lift-fan aircraft - Lessons learned from XV-5 flight experience," in *International Powered Lift Conference*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, Dec. 1993.

[46] Haddox, C., "Phantom Swift: Putting the rapid into rapid prototyping," 2013.

[47] Hajjari, T., Ozel, C., and Kilicman, A., "Measuring Distance of Fuzzy Numbers by Trapezoidal Fuzzy Numbers," vol. 346, pp. 346–356, 2010.

[48] Hayashi, Y. and Buckley, J. J., "Approximations between fuzzy expert systems and neural networks," *International Journal of Approximate Reasoning*, vol. 10, no. 1, pp. 63–73, 1994.

[49] Herrera, F. and Martinez, L., "An approach for combining linguistic and numerical information based on the 2-tuple fuzzy linguistic representation model in decision-making," *Int Journal of Uncertainty Fuzziness and KnowledgeBased Systems*, vol. 8, pp. 539–562, 2000.

[50] Hsu, H.-M. and Chen, C.-T., "Aggregation of fuzzy opinions under group decision making," *Fuzzy Sets and Systems*, vol. 79, no. 3, pp. 279–285, 1996.

[51] Jang, J., Sun, C., and Mizutani, E., "Fuzzy Inference Systems," *Neuro-Fuzzy and Soft Computing: A . . .*, 1997.

[52] Johnson, W., "NDARC - NASA Design and Analysis of RotorCraft Validation and Demonstration," *AHS Aeromechanics Specialists Conference 2010*, no. February, pp. 804–837, 2010.

[53] Johnson, W., Moodie, A. M., and Yeo, H., "Design and Performance of Lift-Offset Rotorcraft for Short-Haul Missions," 2012.

[54] Kahraman, C., *Fuzzy Multi-Criteria Decision Making*, vol. 16 of *Springer Optimization and Its Applications*. Boston, MA: Springer US, 2008.

[55] Khoo, L. and Ho, N., "Framework of a fuzzy quality function deployment system," *International Journal of Production Research*, vol. 34, no. 2, p. 299, 1996.

[56] Kirby, M. R. and Mavris, D. N., "A Method for Technology Selection Based on Benefit, Available Schedule and Budget Resources," Oct. 2000.

[57] Klir, G. J. and Yklsit, B., *Fuzzy Sets and Fuzzy Logic Theory and Applications*. Upper Saddle River, NJ: Prentice Hall PTR, 1995.

[58] Laarhoven, P. V. and Pedrycz, W., "A Fuzzy Extension of Saaty's Priority Theory," *Fuzzy sets and Systems*, vol. 11, pp. 229–241, 1983.

[59] Lafleur, J. M., "Probabilistic AHP and TOPSIS for multi-attribute decision-making under uncertainty," *IEEE Aerospace Conference Proceedings*, 2011.

[60] Lafleur, J. M., Sharma, J. L., and Apa, J., "From Mission Objectives to Design : An Efficient Framework for Downselection in Robotic Space Exploration," *AIAA SPACE 2007 Conference & Exposition*, no. September, 2007.

[61] Lee, C., "DARPA awards VTOL X-Plane contracts to Sikorsky, Aurora Flight Sciences," *IHS Jane's Defence Weekly*, Dec. 2013.

[62] Lee, W., Lau, H., Liu, Z.-z., and Tam, S., "A fuzzy analytic hierarchy process approach in modular product design," *Expert Systems*, pp. 32–42, 2001.

[63] Leishman, J. G., *Principles of Helicopter Aerodynamics*. New York: Cambridge University Press, second edi ed., 2006.

[64] Li, Y., DeLaurentis, D., and Mavris, D., "Advanced Rotorcraft Concept Development and Selection Using a Probabilistic Methodology," in *AIAA's 3rd Annual Aviation Technology, Integration, & Operations (ATIO) Forum*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, Nov. 2003.

[65] Li, Y., Mavris, D., and DeLaurentis, D., "The Investigation of a Decision-Making Technique Using the Loss Function," in *AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum*, no. September, (Reston, Virigina), pp. 1–12, American Institute of Aeronautics and Astronautics, Sept. 2004.

[66] Lichtblau, D., "Differential Evolution in Discrete Optimization," *International Journal of Swarm Intelligence and Evolutionary Computation*, vol. 1, pp. 1–10, 2012.

[67] Lin, C.-T. and Lee, C. S. G., *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems.* Prentice Hall, 1996.

[68] Mamdani, E. and Assilian, S., "An experiment in linguistic synthesis with a fuzzy logic controller," 1975.

[69] Marichal, J.-l., *Aggregation Operators for Multicriteria Decision Aid.* PhD thesis, University of Liege, 1999.

[70] Marsh, E., "Hierarchical decision making in machine design," 1993.

[71] Mattson, C. and Messac, A., "Pareto Frontier Based Concept Selection Under Uncertainty , with Visualization," *Optimization and Engineering*, vol. 6, pp. 85–115, 2005.

[72] Mattson, C. and Messac, A., "Development of a Pareto-based Concept Selection Method," in *43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, no. April, (Reston, Virigina), American Institute of Aeronautics and Astronautics, Apr. 2002.

[73] Mattson, C. a. and Messac, A., "Concept Selection Using s-Pareto Frontiers," *AIAA Journal*, vol. 41, pp. 1190–1198, June 2003.

[74] Mavris, D. and DeLaurentis, D., "A stochastic approach to multi-disciplinary aircraft analysis and design," *36th Aerospace Sciences ...*, 1998.

[75] McCulloch, J., Wagner, C., and Aickelin, U., "Measuring the Directional Distance Between Fuzzy Sets," *UKCI 2013, the 13th Annual Workshop on Computational Intelligence, Surrey University*, no. 2, pp. 38–45, 2013.

[76] Mosher, T., "Conceptual spacecraft design using a genetic algorithm trade selection process," *Journal of Aircraft*, vol. 36, no. 1, pp. 200–208, 1999.

[77] Mullur, A., Mattson, C., and Messac, A., "New Decision Matrix Based Approach to Concept Selection using Linear Physical Programming," in *AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, (Norfolk, Virginia), Paper No. AIAA-2003-1446, 2003.

[78] Mullur, A. A., Mattson, C. A., and Messac, A., "Pitfalls of the Typical Construction of Decision Matrices for Concept Selection," *41st Aerospace Sciences Meeting and Exhibit*, no. AIAA 2003-0466, 2003.

[79] Nauck, D. and Kruse, R., "A Neuro-Fuzzy Approach to Obtain Interpretable Fuzzy Systems for Function Approximation," in *IEEE International Conference on Fuzzy Systems (FUZZ_IEEE'98)*, (Anchorage, AK), pp. 1–5, 1998.

[80] (NCAT), T. N. C. f. A. T., "Technology For Affordability: A Report on the Activities of the Working Group, Integrated Product/Process Development Simplified Contracting Dual-Use Manufacturing," tech. rep., 1993.

[81] Negnevitsky, M., *Artificial Intelligence: A Guide to Intelligent Systems.* Essex, England: Addison-Wesley, second edi ed., 2005.

[82] Okudan, G. E. and Tauhid, S., "Concept selection methods a literature review from 1980 to 2008," *International Journal of Design Engineering*, vol. 1, no. 3, p. 243, 2008.

[83] Patel, M., Lewis, K., Maria, A., and Messac, A., "System design through subsystem selection using physical programming," *AIAA journal*, vol. 41, no. 6, pp. 1089–1096, 2003.

[84] Pegg, R. J., "Summary of Flight Test Results of the VZ-2 Tilt-Wing Aircraft," tech. rep., Langley Research Center, Langley Air Force Base, VA, 1962.

[85] Perez, R., Chung, J., and Behdinan, K., "Aircraft Conceptual Design Using Genetic Algorithms," in *8th Symposium on Multidisciplinary Analysis and Optimization*, no. c, (Reston, Virigina), American Institute of Aeronautics and Astronautics, Sept. 2000.

[86] Pugh, S., *Creating innovative products using total design : the living legacy of Stuart Pugh.* Reading, Mass.: Addison-Wesley Pub. Co., 1996.

[87] Quackenbush, T. R., Wachspress, D. A., Mckillip, R. M., and Sibilia, M. J., "Aerodynamic Studies of High Advance Ratio Rotor Systems," in *67th Annual Forum of the American Helicopter Society*, (Virginia Beach, VA), 2011.

[88] Raymer, D. P., *Aircraft Design: A Conceptual Approach.* Reston, Virigina: American Institute of Aeronautics and Astronautics, 5th ed. ed., 2012.

[89] Ritchey, T., "General Morphological Analysis," 1998.

[90] Rutherford, J. M., O'Rourke, M., Martin, C., Lovenguth, M., and Mitchell, C., "Technology Needs for High-Speed Rotorcraft," tech. rep., Ames Research Center, Moffett Field, CA, 1991.

[91] Saaty, T. L., "The Analytic Hierarchy Process," *Education*, pp. 1–11, 1980.

[92] Sakawa, M., *Genetic Algorithms and Fuzzy Multiobjective Optimization*, vol. 14 of *Operations Research/Computer Science Interfaces Series.* Boston, MA: Springer US, 2002.

[93] SANTAMARINA, J. C. and CHAMEAU, J.-L., "Membership functions II: Trends in fuzziness and implications," *International Journal of Approximate Reasoning*, vol. 1, no. 3, pp. 303–317, 1987.

[94] SANTOS, F. J. J. and CARLOS, S. A., "Fuzzy Systems for Multicriteria Decision Making," *CLEI Electronic Journal*, vol. 13, no. 3, pp. 2–9, 2010.

[95] SCHRAGE, D. P., "Vehicle Synthesis for Advanced VTOL Aircraft," 2004.

[96] SCHRAGE, D. P. and MAVRIS, D. N., "Rotorcraft system design for affordability through Integrated Product/Process Development (IPPD)," in *European Rotorcraft Forum 22nd Brighton United Kingdom Proceedings UNITED KINGDOM 1719 Sept 19961216*, vol. 1, p. 2, 1996.

[97] SCHRAGE, D., "Technology for rotorcraft affordability through integrated product/process development (IPPD)," in *American Helicopter Society 55th Annual Forum*, (Montreal, Canada), American Helicopter Society, Inc., 1999.

[98] SCOTT, M. S. A., "Technology Needs for High Speed Rotorcraft (2)," tech. rep., Ames Research Center, Moffett Field, CA, 1991.

[99] SHEN, X., TAN, K., and XIE, M., "The implementation of quality function deployment based on linguistic data," *Journal of Intelligent Manufacturing*, vol. 12, pp. 65–75, 2001.

[100] SHTOVBA, S. D., "Fuzzy model tuning based on a training set with fuzzy model output values," *Cybernetics and Systems Analysis*, vol. 43, no. 3, pp. 334–340, 2007.

[101] SILVA, C., YEO, H., and JOHNSON, W., "Design of a Slowed-Rotor Compound Helicopter for Future Joint Service Missions," *American Helicopter Society*, 2010.

[102] SIMONDS, R., "A Generalized Graphical Method of Minimum Gross Weight Estimation," *. . . Society of Aeronautical Weight Engineers,(San Diego, . . .*, 1956.

[103] SINGH, H., GUPTA, M. M., MEITZLER, T., HOU, Z.-G., GARG, K. K., SOLO, A. M. G., and ZADEH, L. A., "Real-Life Applications of Fuzzy Logic," *Advances in Fuzzy Systems*, vol. 2013, pp. 1–3, 2013.

[104] SINGPURWALLA, N. D. and BOOKER, J. M., "Membership Functions and Probability Measures of Fuzzy Sets," *Journal of the American Statistical Association*, vol. 99, pp. 867–877, Sept. 2004.

[105] SMALING, R. and DE WECK, O., "Fuzzy pareto frontiers in multidisciplinary system architecture analysis," in *AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, no. September, (Albany, New York), 2004.

[106] Sodhi, B. and T., P., "A Simplified Description of Fuzzy TOPSIS," p. 3, May 2012.

[107] Spiegelhalter, David J.; Dawid, A. Philip; Lauritzen, Steffen L.; Cowell, R. G., "Spiegelhalter - 1993 - Bayesian Analysis in Expert Systems.pdf," *Statistical Science*, vol. 8, pp. 219–283, 1993.

[108] Srdjevic, B., "Combining different prioritization methods in the analytic hierarchy process synthesis," *Computers & Operations Research*, vol. 32, pp. 1897–1919, July 2005.

[109] Storn, R. and Price, K., "Differential evolutiona simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization*, pp. 341–359, 1997.

[110] Strawbridge, Z., Mcadams, D. A., and Stone, R. B., "A COMPUTATIONAL APPROACH TO CONCEPTUAL DESIGN," in *ASME 2002 Design Engineering Technical Conference*, (Montreal, Canada), 2002.

[111] Sun, X. and Li, Y., "An Intelligent Multi-Criteria Decision Support System for Systems Design," *10th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, pp. 1–11, Sept. 2010.

[112] Tai, J. C., Mavris, D. N., and Schrage, D. P., "A comparative assessment of high-speed rotorcraft concepts (HSRC) - Reaction driven stopped rotor/wing and variable diameter tiltrotor," *AIAA and SAE, 1997 World Aviation Congress, Anaheim, CA; UNITED STATES; 13-16 Oct. 1997*, no. 975548, 1997.

[113] Takagi, T. and Sugeno, M., "Fuzzy identification of systems and its applications to modeling and control," *Ieee Transactions On Systems Man And Cybernetics*, vol. 15, pp. 116–132, 1985.

[114] Tauhid, S. and Okudan, G., "Fuzzy information axiom approach for design concept evaluation," *... Conference on Engineering Design*, no. August, pp. 1–12, 2007.

[115] Textron, B. H., "Bell Valor V-280," 2014.

[116] Thapar, A., Pandey, D., and Gaur, S., "Satisficing solutions of multi-objective fuzzy optimization problems using genetic algorithm," *Applied Soft Computing*, vol. 12, pp. 2178–2187, Aug. 2012.

[117] Thurston, D. L., "A formal method for subjective design evaluation with multiple attributes," *Research in Engineering Design*, vol. 3, pp. 105–122, June 1991.

[118] Thurston, D. L. and Locascio, A., "Decision Theory for Design Economics," *The Engineering Economist*, vol. 40, pp. 41–71, Jan. 1994.

[119] Tong, R. and Bonissone, P., "Linguistic Approach to Decisionmaking with Fuzzy Sets," *Systems, Man and Cybernetics, ...*, vol. 75, no. 1, pp. 716–723, 1980.

[120] Tran, L. and Duckstein, L., "Comparison of fuzzy numbers using a fuzzy distance measure," *Fuzzy Sets and Systems*, vol. 130, pp. 331–341, Sept. 2002.

[121] Triantaphyllou, E., *Multi-criteria decision making methods: a comparative study*, vol. 44. 2000.

[122] Triantaphyllou, E. and Lin, C.-t., "Development and Evaluation of Five Fuzzy Multiattribute Decision-Making Methods," *International Journal of Approximate Reasoning*, vol. 14, pp. 281–310, 1996.

[123] Vieira, J., Dias, F., and Mota, A., "Neuro-Fuzzy Systems : A Survey," *... on Neural Networks and Applications, Udine ...*, 2004.

[124] Villeneuve, F. and Mavris, D., "A New Method of Architecture Selection for Launch Vehicles," *AIAA/CIRA 13th International Space Planes and Hypersonics Systems and Technologies Conference*, pp. 1–17, May 2005.

[125] Vollmer, D. T., Soule, T., and Manic, M., "A Distance Measure Comparison to Improve Crowding in Multi-Modal Optimization Problems," in *3rd International Symposium on Resiliant Control Systems* (Laboratory, I. N., ed.), 2010.

[126] Wang, J. R., "Ranking engineering design concepts using a fuzzy outranking preference model," *Fuzzy Sets and Systems*, vol. 119, pp. 161–170, Apr. 2001.

[127] Wang, L.-X. and Mendel, J. M., "Generating Fuzzy Rules by Learning from Examples," *Ieee Transactions On Systems Man And Cybernetics*, vol. 22, no. 6, pp. 1414–1427, 1992.

[128] Wang, Y.-M. and Elhag, T. M., "Fuzzy TOPSIS method based on alpha level sets with an application to bridge risk assessment," *Expert Systems with Applications*, vol. 31, pp. 309–319, Aug. 2006.

[129] Wang, Y.-M. and Luo, Y., "On rank reversal in decision analysis," *Mathematical and Computer Modelling*, vol. 49, no. 5-6, pp. 1221–1229, 2009.

[130] Warner, J., "SciKits - scikit-fuzzy," 2015.

[131] Warwick, G., "Groen Brothers starts work on high-speed VTOL Heliplane for DARPA," *Flight International, Vol. 168, no. 5015*, pp. 13–19, Dec. 2005.

[132] Weber, R. G., Condoor, S. S., and Louis, S., "Conceptual Design Using a Synergistically Compatible Morphological Matrix," 1998.

[133] Xu, Z.-S. and Chen, J., "An interactive method for fuzzy multiple attribute group decision making," *Information Sciences*, vol. 177, pp. 248–263, Jan. 2007.

[134] Yager, R., "On ordered weighted averaging aggregation operators in multi criteria decision making," *IEEE Trans. Syst. Man Cybern.*, vol. 18, no. 1, pp. 183–190, 1988.

[135] Yang, R., Chuang, C.-H., and Che, X., "Recent applications of topology optimization," in *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, Sept. 1998.

[136] Yoon, K. and Hwang, C.-L., *Multiple attribute decision making: an introduction.* 1995.

[137] Yuen, K., "On Limitations of the Prioritization Methods in Analytic Hierarchy Process: A Study of Transportation Selection Problems," *Proceedings of the International MultiConference of ...*, vol. II, 2009.

[138] Zadeh, L. A., "Fuzzy sets," *Information and Control*, vol. 8, pp. 338–353, 1965.

[139] Zadeh, L., "Outline of a new approach to the analysis of complex systems and decision processes," *Systems, Man and Cybernetics, IEEE Transactions ...*, vol. SMC-3, no. 1, pp. 28–44, 1973.

[140] Zadeh, L., "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems*, vol. 1, pp. 3–28, Jan. 1978.

[141] Zhai, L.-Y., Khoo, L.-P., and Zhong, Z.-W., "Design concept evaluation in product development using rough sets and grey relation analysis," *Expert Systems with Applications*, vol. 36, pp. 7072–7079, Apr. 2009.

[142] Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J., "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization," 1997.