ABSTRACT

TORTORICE, MARCUS, ALLAN. Investigation of Reconfigurable Systems in the Context of Robust Design and Preference Effects on Associated Trade-off Decisions. (Under the direction of Dr. Scott Ferguson).

Reconfigurable system design is one area of research that seeks to address trade-off decisions with the ability to modify the system after deployment. In addition to trade-offs designers must accommodate unexpected changes. Traditionally, Robust Design, which explores the trade-off between maximum and consistent performance, has been one solution. The potential benefit of a reconfigurable system is that it might address this trade-off and improve both system performance and minimize sensitivity to uncontrollable variations. This leads to the first research question: ***How effective is reconfigurability at mitigating the performance impact of uncontrollable system variations?*** Answering this question addresses a challenge associated with Robust design; adopting a robust solution often results in performance loss with minimal performance variation.

Also, the added complexity inherent in reconfigurable systems requires designers to make trade-offs such as performance versus complexity. Balancing performance gains and the “cost” of reconfigurability requires a designer to evaluate the system making the decision dependent on individual preferences. This leads to the second research question: ***How sensitive to designer preference are the tradeoff decisions associated with choosing a reconfigurable system?*** Answering this research question will emphasize the importance of giving designers options, and evaluating each in the context of designer preferences.

To answer the first research question, a racecar case study is used, and uncontrollable variations are introduced causing performance variations over the course of a race. Epoch-Era Analysis is used to limit reconfigurability by discretizing the operating regime, and controlling the transition criteria. Schemes defined by the number of unique end states are introduced and compared to a static robust system. A multi-objective optimization using two objectives; performance - mean lap time and robustness - standard deviation, results in Pareto frontiers containing optimal designs.

Two of the systems, the two and ten state, dominate the benchmark with comparable standard deviations and improved lap times. These reconfigurable systems maintain mean lap times 8% to 12% faster than the static system. This investigation shows that the reconfigurable systems can mitigate performance variations while eliminating the sacrifice associated with the trade-off between performance and robustness.

For the second research question, the effects of designer preferences on two trade-off decisions are examined using these frontiers. Two virtual designers are created with differing perspectives, and their preferences are applied to system attributes. The aggregate utilities are compared, and the results are displayed as a heat map to illustrate each designer's focus.

The investigation demonstrates some effects associated with the designer preferences such as a divergent focus. The designers often focus on completely opposite ends of the frontiers, implying an effect on the performance versus robustness trade-off. Additionally, the designers target different reconfiguration schemes, implying an effect on the performance versus complexity trade-off. The designers do not always diverge illustrated by some non-obvious inconsistencies introduced in the focus of the designers. The results illustrate that trade-off decisions are greatly impacted by the designer preferences and even subtle differences can change the outcome of the decision in question.

These investigations demonstrate the importance of choice in the design process. The addition of reconfigurability gives designers a means of mitigating performance variations outside of traditional Robust design, while improving overall system performance. Additionally, the different reconfiguration schemes inhabit a variety of locations in the design space, and introduce varying levels of complexity allowing designers greater flexibility when attempting to balance factors in a complex system. For the second research question, it is shown that different designer preferences can greatly affect the outcome of trade-off decisions. The inclusion of an expanded array of choices would ensure that designer preferences are more likely to be realized in the final system decision.

Investigation of Reconfigurable Systems in the Context of Robust Design and Preference Effects on Associated Trade-off Decisions

by
Marcus Allan Tortorice

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Master of Science

Aerospace Engineering

Raleigh, North Carolina

2013

APPROVED BY:

Dr. Scott Ferguson
Committee Chair

Dr. Gregory Buckner

Dr. Andre Mazzoleni

BIOGRAPHY

Marcus Tortorice graduated from the North Carolina School of Science and Mathematics in 2006. He then studied Aerospace Engineering at North Carolina State University, and completed his undergraduate career by participating in a space-themed senior design project, based on bio-inspired planetary rovers. In 2010, Marcus graduated Magna Cum Laude with a Bachelor's of Science in Aerospace Engineering. He then started graduate work at NC State in the System Design and Optimization Lab, advised by Dr. Ferguson.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction and Motivation

1.1 Challenges of complex system design

Technological advancements and the desire for greater performance have led to engineered systems with increased complexity. Complex systems are those that are "made up of a large number of parts that interact in a nonsimple way..." [1]. Additionally, Simon states that "given the properties of the parts and the laws of their interaction, it is not a trivial matter to infer the properties of the whole." In support of this statement, a defining characteristic of complex systems has been identified as the challenge of predicting the global behaviors that result from the interactions among a large number of simple parts [2].

From an engineering perspective, an additional challenge of designing complex systems is the need for multiple design teams that span multiple engineering disciplines. Often, these teams are in conflict as they have their own goals and constraints. However, these teams must also work together to satisfy system-level performance targets, requirements, and constraints [3]. This interaction is made more difficult by the fact that the subsystems / multiple design teams are often coupled.

Overall, system complexity affects a designer's ability to find the best design. To even create an acceptable design, designers must balance the many factors that influence the value of the final system. These factors can include criteria such as performance, cost, complexity, and reliability. The balancing act associated with these factors is inherent in the design of all systems, but is especially challenging for complex systems. In complex systems, subsystem interaction may not be fully understood, and changes based on one factor can affect the others in non-obvious ways. This idea is explored in research pertaining to change propagation [4]. Change propagation analysis is developed around the notion that any time there is a design change, it may set off a series of other changes that transform into a propagating flow that affects large sections of the design [5]. Publications in this area of research highlight the difficulty associated with predicting the spread of changes because some aspects that change may not be directly connected to the initially changed component [6]. Furthermore, the degree to which changes can propagate is dependent on the complexity of the system in question [5].

While it may be easy to suggest that a designer should aim for the best of all system properties, a majority of the time these factors are in competition. Gains in one measure may adversely affect another, and the interaction of any two parts can be positive, negative, neither, or change unpredictably [7]. This give and take between goals is a trade-off, and the occurrence of such trade-offs increases as the complexity of systems increases [8]. As designers attempt to balance the various factors that contribute to a good design, they are forced to make trade-offs in order to meet the objective of the system in question.

1.2 Exploring trade-offs in complex system design

To arrive at the best design, a number of trade-off decisions are made over the course of the design process. These occur when design decisions conflict with other decisions, and a designer's understanding of the trade-offs present in a system influence the entire design process [9]. The consequence of a trade-off from a design perspective is that a designer is often forced to sacrifice in one area to gain in others [8]. In simplified terms, a trade-off can be thought of as a compromise.

Handling trade-offs in engineering design is often done in a multi-objective environment. Each factor that influences the value of the final system can be thought of as an objective function for a multiobjective problem formulation. The values of the objectives are determined using models that link system performance measures to design variables that represent the physical characteristics of the system. While single objective optimizations are defined by the identification of a single optimal solution, a multiobjective optimization approach searches for a set of non-dominated solutions – commonly referred to as the Pareto frontier [10]. This frontier of non-dominated solutions forms a curve or surface depending on the number of objectives, and the endpoints of the curve represent extreme designs where one objective has been sacrificed as much as possible for gains in the other. Figure 1.1 illustrates an example of a Pareto frontier for a two-objective minimization problem formulation.

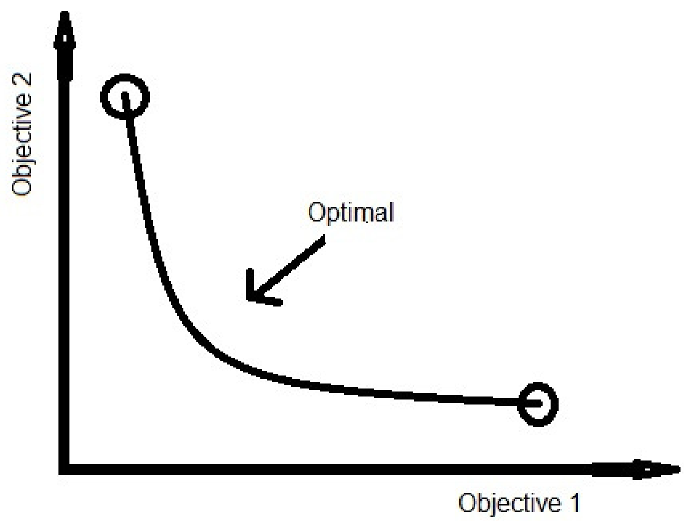

Figure 1.1 Two Objective Pareto Frontier

In the example above, the circled endpoints represent designs with the best performance in each objective. The left-most point is a design that optimizes Objective 1 while sacrificing performance in Objective 2. The opposite is true for the right-most point, where performance in Objective 2 is gained at the expense of performance in Objective 1. The curve between these points represents varying levels of trade-off or compromise between the competing objectives. By tracing along the Pareto Frontier, designers can visualize the effects that trade-offs will make on the system. The challenge is selecting a point from the Pareto frontier and accepting some degree of performance tradeoff.

1.3 Using reconfigurability to mitigate the need for trade-offs

The traditional design mindset is that a designer must accept the trade-offs present in a multiobjective space. Recent research has begun to challenge this notion, as designers look for systems with increased performance capabilities and longer life-cycles. Reconfigurable system design is one area of research that explores the necessary trade-offs in a multiobjective space by exploring systems that can change their configuration in reversible and repeatable ways after being put into service [11]. From a multiobjective context, a reconfigurable system is able to occupy different locations in the performance space throughout its lifecycle. For a designer, this ability to move through the performance space provides a tool for managing the trade-offs between competing performance objectives throughout a product's lifecycle.

Previous work in this area has shown that reconfigurable systems are capable of mitigating sacrifices between different performance objectives, and it has been suggested that certain performance trade-offs could be eliminated altogether [12], [13]. Example applications that have been studied include a Formula 1 style race car that encounters various track segments or aircraft required to have good maneuverability and high endurance [8], [14], [15], [16].

Without an ability to reconfigure, designers are forced to balance trade-off decisions between multiple, conflicting objectives. For the race car example, the optimal configuration for turning is different than the configuration for traveling a straight. An effective turning configuration must be highly stable when subjected to lateral forces while maintain sufficient grip. This calls for stiff side-to-side suspension and high downforce. Conversely, the optimal straight configuration would need little in the way of lateral stability but would try to maintain traction to maximize power transfer to the road. This requires a stiff rear suspension and a different downforce profile.

However, the presence of multiple, conflicting objectives is not the only challenge faced in system design. Designers must also accommodate unexpected and unplanned changes – to both the system and the environment – while maintaining an effective level of performance. This issue is addressed in the next section.

1.4 Managing performance variability using robust design

In addition to the challenge of competing performance objectives, designers must also understand how uncontrollable factors can influence system performance. It has been stated that:

> *"In deterministic design we assume there is no uncertainty in the design variables and/or modeling parameters. Therefore there is no variability in simulation outputs. However, there exists inherent input and parameter variation that results in output variation." [17]*

In other words, a system can run into a number of non-ideal conditions or experience unexpected variations both internally and externally. A common example of an internal variation is the variability associated with a manufacturing process. In this context, a designer often establishes a tolerance that is deemed acceptable to minimize the impact of such variabilities. An example of an external variation would be weather or other environmental factors. Overall, the presence of uncontrollable variations forces designers to address performance consistency when designing a system.

The goal of robust design, as proposed by Taguchi, is to diminish or mitigate the effects of uncontrollable variations without eliminating the variations themselves [18]. Robust Design explores the trade-off between maximum and consistent performance. This trade-off is made so that variations in the design variables, or operating parameters, will have a minimal effect on system performance. This method is illustrated in the Figure 1.2.

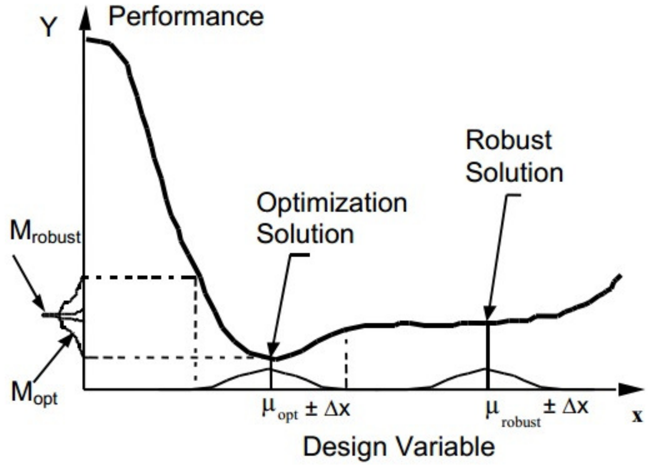

Figure 1.2: Example of the Principles of Robust Design [18]

The performance curve in this figure illustrates the trade-off that a designer must navigate. The performance objective on the y-axis is to be minimized, while the x-axis represents the value of a system design variable. The optimum value for this variable is given by μ_{opt}, as it correlates with minimum system performance. In this case, the uncontrollable noise factors are represented by a distribution from the mean design variable value (plus or minus Δx). In this example, a deviation from the optimal design variable value causes a significant increase in system performance. The range of performance values for this scenario is represented by M_{opt}. A robust solution is represented by μ_{robust}. While the performance of the system for this configuration is not as effective as the optimal design, the range of system performance due to the uncontrollable factor is much smaller. This range is shown by M_{robust}.

From a multiobjective perspective, the principle of Robust design can be considered as a two objective problem. The first objective can be formulated to represent the system performance at the mean expected value of the design variables. Then, the second objective can be formulated to represent performance consistency due to uncontrollable variations. This approach has been demonstrated in previous work, such as in the design of compressor fan blades robust to their own deterioration [19]. Additionally, reliability and robustness have been used to formulate a multiobjective problem where the mean and variation of multiple performance measures are simultaneously minimized [17].

While Robust design can help manage the impact of uncontrollable variations, resulting solutions often occur at significant performance loss. The objective of this thesis, as explained by the research questions presented in the next section, is to explore how reconfigurability may be used to manage the tradeoffs associated with uncontrollable factors while mitigating the required overall performance loss.

1.5 Formulating the research questions

Previous research has demonstrated that reconfigurable systems can be effective at improving mitigation of the effects of trade-offs when competing performance objectives are considered [8], [14], [20]. However, an additional motivation for reconfigurability is the potential to minimize the impact of various uncertainties (or various uncontrollable noise factors) that arises once a system has been deployed. This motivation leads to the first research question:

How effective is reconfigurability at mitigating the performance impact of uncontrollable system variations?

Answering this question will explicitly address a challenge associated with Robust design. That is, adopting a robust solution often results in a generalized performance loss with minimal performance variation. A designer must then navigate the trade-off between mean system performance and performance consistency. Reconfigurability has been proposed as a means of addressing design challenges associated with survivability, reliability, efficiency and consistency [20], [21]. Previous work in the area of reconfigurability has confirmed the performance advantages over a static system in the context of a multi-objective problem, and the increased benefit of such systems as the degree of uncertainty increases [14], [20], [22].

The potential benefit of a reconfigurable system is that it might improve both system performance and minimize sensitivity to uncontrollable variations. Answering this question will require identifying possible sources of performance variability and characterizing how the system might change. This will be done by comparing the capabilities of a reconfigurable system against a system designed using Robust design principles.

As this process will likely involve a multiobjective frontier of non-dominated solution, the designer is faced with challenge of selecting the configuration that they most prefer. The final outcome of this decision is unique for each designer, and will vary depending on both the system in question and the designer's preferences. Additionally, while the multiobjective space may represent possible performance trade-offs, reconfigurable systems are likely to be more complex due to the addition of sensors, actuators and controllers needed to enact the desired configuration changes [11]. This may make the reconfigurable system prohibitively impractical as a design option, or cause one reconfigurable system to be preferred to another solution. Therefore, it is also necessary to explore the ramifications of reconfigurability in addition to the possible performance benefits.

In this case, there is a trade-off between performance and complexity where increasing performance is accompanied by the negative effect of increasing complexity. This trade-off alone can necessitate challenging decisions on the part of the designer. The challenge is compounded when the interconnections between factors are taken into account. As an example, the previous section discussed reconfigurability as one way to mitigate the effects of the performance to consistency trade-off. Unfortunately, this not only increases performance, but it also increases complexity. Therefore, because of the interconnected nature of the factors affecting system worthiness, addressing one trade-off between factors can cause unintended changes to other factors of the system.

Balancing performance gains and the “cost” of reconfigurability requires a designer to evaluate a system concept across many factors. Because of this, the decision outcome will be highly dependent on the individual preferences of each designer. Therefore, the second research question explored in this work is:

How sensitive to designer preference are the tradeoff decisions associated with choosing a reconfigurable system?

Answering this question will require the identification of the aspects that influence the decision and the application of a preference structure to compare the candidate designs. This will require defining strength of preference curves and exploring how the form of these curves influence the chosen system.

1.6 Outline of the remaining thesis chapters

The outline for the rest of the thesis is as follows. Chapter 2 will discuss the background of reconfigurable system design and introduce the technical foundation behind the technologies used to answer the two research questions. The first research question is addressed in Chapter 3, while Chapter 4 explores the second research question. Finally, Chapter 5 highlights the conclusions from this work and offers avenues for future work.

CHAPTER 2

Thesis Background Research

2.1 Reconfigurability

Although the idea of reconfigurability, as it relates to the engineering design thought process, is still a relatively new idea, there have been multiple groups of scientists who have worked to develop both a cohesive definition of ‘reconfigurability’ as well as the possible methods for how to apply this theory to engineering design. As one of the main focuses of this thesis paper is to study reconfigurability with regard to complex system design, it is important to review several of the previous definitions and methods of using reconfigurability that have been developed.

2.1.1 Uses of reconfigurability

Ferguson et. al. describe reconfigurable systems as “a subset of changeable systems,” where the term ‘changeable systems’ refers to systems whose configurations can be changed after production and deployment [11]. Per Ferguson et. al.’s definition, reconfigurable systems should be able to change both repeatedly as well as reversibly, which gives designers a valuable tool for solving complex design objectives. Additionally, Siddiqi et. al. describe reconfigurable systems as “systems able to achieve distinct physical configurations or states through alteration of form or function within an acceptable reconfiguration time and cost [20].” Siddiqi et. al.’s definition is more specific in that it includes restraints such as time and cost. To help further explain the definition of and uses for reconfigurability, Siddiqi et. al include a figure in their work, detailing each category of need that can be addressed by reconfigurable systems. Refer to Figure 2.1.

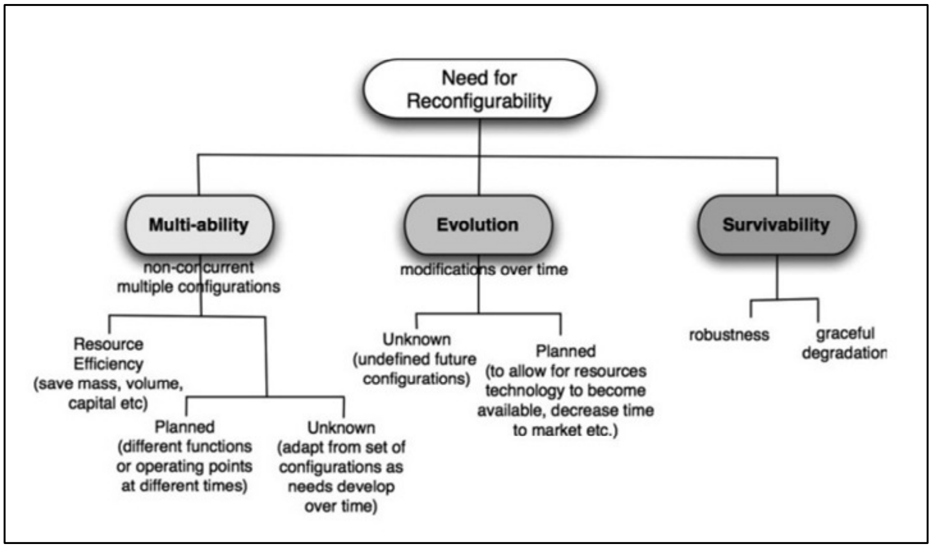

Figure 2.1: Needs for Implementing Reconfigurability [20]

As Figure 1 indicates, there are three main categories for including reconfigurability within a system design: multi-ability needs, evolution needs and survivability needs. Reconfigurability due to 'multi-ability needs' most commonly refers to design situations in which several different configurations are desirable, and all need to be included due to specific limits, such as volume, capital, operation points, etc. Reconfigurability due to 'evolution needs' refers to situations in which the design is anticipated to change over time, whether to provide flexibility to unknown variables or in response to planned changes. Lastly, reconfigurability due to 'survivability' refers to design situations that require either an elevated robustness or slow, graceful degradation of a unit.

There are several researchers who have already investigated the use of reconfigurability in the context of these three categories. For multi-ability systems, "the primary motivation for reconfigurability … comes from the inherent trade-offs incorporated when resolving the issue of conflicting objectives. Physical reconfigurations are used after deployment to maintain a high level of performance [12]." A prime example of this can be seen in reconfigurable aircraft, whether it's to determine which individual components can be reconfigured, or by considering reconfigurability itself as a variable [15], [16]. Both of these applications use reconfigurability in a multi-ability setting.

As stated above, reconfigurability with regard to an 'evolutionary system' refers to situations in which the design is anticipated to change over time. One example of the potential of this category can be seen in the work of Lewis et. al., where a modular, manual irrigation pump was designed with the purpose of poverty alleviation [23]. Reconfigurable components were included in the product to allow the pump to change over time as the customer needs and economic status changed. By using a multi-objective approach, they were able to take into account the relationship between affordability and income potential.

Lastly, within the 'system survivability' category, the intent of the system is to provide robustness or graceful degradation. An example of this can be seen in the work of Sullivan et. al., where a race car model is used to demonstrate how reconfigurability within a design can add consistency benefits [24]. In the experiment, static systems, robust systems and reconfigurable systems were simulated to monitor how each type of system mitigated the effects of uncontrollable variations, such as tire wear and fuel use. The reconfigurable system improved both speed and robustness when compared to the static systems.

Additional examples for reconfigurability can be found for a variety of products that are already in use. Siddiqi et. al. incorporated reconfigurability into the design of the wheels of planetary surface vehicles to address concerns over mobility issues in variable terrain [20]. By considering reconfigurability in their design, they were able to demonstrate a considerable improvement in tractive performance. Ferguson et. al. combined reconfigurability with the idea of product platforming to create a family of reconfigurable race cars, using lap time as the system objective [14]. By considering reconfigurability in the race car design, the authors were able to show improvements in lap time performance compared to static baseline vehicles.

2.1.2 Utilizing reconfigurability in the design process

In addition to categorizing reconfigurable design based on a need, reconfigurable systems themselves can be classified based on how the system transforms from one state to another. Although it is not necessary to classify each reconfigurable system in this way, it can help a designer understand how to include reconfigurable principles when designing a system. Within the work of Singh et al., the authors describe how they examined a variety of reconfigurable systems, and determined three categories of transformation principles for reconfigurable systems: expand/collapse, expose/cover, and fuse/divide [25]. The three categories are defined as follows:

> "**Expand/collapse**: changing the physical dimension of an object to bring about an increase/decrease in occupied volume, primarily along an axis, in a plane, or in three dimensions.
>
> **Expose/cover**: revealing or concealing a new surface to alter functionality.
>
> **Fuse/divide**: making a single functional device become two or more devices, at least one of which has its own distinct functionality defined by the state of the transformer."

Singh et. al. also describe how a variety of everyday products currently use or could use these three principles to reconfigure [26]**.** The application of transformation principles to current products was further elaborated on by Parkinson and Haldaman, with the addition of a fourth transformation principle, 'reorientation [27].' This principle refers to "products that create a new configuration by reorienting some aspect of the product in a new way." Parkinston and Haldaman state that these four transformation principles can be used by a designer to analyze how a physical change occurs within a system, especially to facilitate brainstorming of new reconfigurable products. Lastly, Kuhr et. al. used the same three transformation principles described by Singh et. al. to create a "visual concept-generating process" [28]. The aim of their process is to help designers generate new concepts for reconfigurable systems, revealing portions of the design space that would be previously unexplored.

Multiple design approaches have been developed through the years that focus on the decision-making involved in designing reconfigurable systems. Introductory research in reconfigurable system design used the Decision-Based Design (DBD) framework, which was introduced by Hazelrigg to determine the increased costs of reconfiguration [29]. Optimization techniques were used to select the ranges that produced the best reconfigurable system performance. Olewnik and Lewis then built upon this model by using conjoint analysis to assess the component ‘part-worth’ for each attribute comprising a product. This made it possible to calculate the product’s total utility [13]. Building upon the Decision-Based Design, Olewnik and Lewis also outlined a decision support framework for making critical decisions in the design for reconfigurable systems. Their decision support framework takes into consideration multiple variables and measures, such as corporate preferences, budget, adaptability, and robustness.

A different approach for managing reconfigurability in the design process is to focus on how to choose the appropriate reconfigurable variables. Khire and Messac introduced the Variable-Segregating Mapping-Function (VSMF), which can be used to integrate design variable selection, both adaptive and fixed variables, with the optimization of system performance [30]. By integrating these two processes, a possible source of sub-optimality, i.e. the necessity of the designer to choose which variables are adaptive, can be eliminated. The VSMF method optimizes the number of adaptable variables and the system at the same time, creating a bi-objective optimization that maximizes adaptability and minimizes the penalty. Along the same lines, a constraint-based approach was developed by Ferguson and Lewis, to aid in design variable selection by focusing on changes in system mass [31]. The authors reasoned that, if the mass of the extra components needed to achieve reconfigurability is too large, the performance advantage of reconfigurability is offset. This reasoning led to establishing mass as the effective system constraint when determining which variables are allowed to reconfigure. Other researchers have used design variable variations or treated morphing as an “independent variable” to determine which components should be changed, and by what magnitude [15], [16]. For example, Martin and Crossley designed a family of non-morphing systems to help determine the most beneficial reconfigurable variables, and Roth and Crossley treated morphing as a design variable in their system optimization process, to determine the level of reconfigurability that was appropriate for the system.

2.1.3 Methods of modeling reconfigurable systems

The last aspect of reconfigurability that will be discussed focuses on different methods of modeling that have been explored with regard to reconfigurable systems. In the work of Siddiqi and deWeck, a variety of modeling techniques are demonstrated for reconfigurable systems, including Markov models and control theory methods [21]. Using these models, Siddiqi and deWeck deduce a number of overarching principles that can be applied to the design of reconfigurable systems, such as the Principle of Reconfigurability, the Principle of Self-Similarity, and the Principle of Information Reconfiguration. In another paper written by Siddiqi, the implementation of a controls-based approach is described which allows designers to identify ‘good’ configurations that the system is able to adopt over the course of its operations [20]. Ferguson and Lewis have also written on the use of control theory methods to model the changes from one configuration to another [8]. They explain how a system can be optimized for a pre-determined number of functional objectives, and the pathway of each reconfiguration is also optimized with the use of a linear tracking controller. Finally, Chmarra et. al examine the behavior of reconfigurable systems as it relates to the transition between end states [32]. They apply graph search algorithms to the design of a reconfigurable printer in an attempt to gain the optimal trajectory for state transition.

2.2 Robust Design

The second focus of this thesis paper is to explore the idea of 'robust design' as it relates to complex systems. There have been many researchers who have explored the subtleties of this idea, and have developed methods of improving the design process. The first robust design techniques were introduced by Taguchi, who specifically addressed manufacturing uncertainties, such as material anomalies or manufacturing tolerances. The technique focuses on minimizing performance degradation and/or variation when faced with uncontrollable variations. These influences on system performance have been classified as *noise factors*, and are characterized by inducing some loss in a system [33], [34], [35]. Noise factors can come from an assortment of areas, including manufacturing uncertainties and environmental variations. Separately, Clausing categorizes the noise factors into three different types: variations in conditions of use, production variations, and deterioration [36]. Oftentimes in design or optimization, noise factors are associated with a statistical distribution for each design variable, and are described as being too expensive or difficult to control [37]. Noise factors often have the effect of making what is an optimal solution, for a specific set of design variables, a highly non-optimal solution when the variables are perturbed. The inability to remove or control noise factors is a main motivation behind robust design principles, which are concerned with minimizing the effects of uncertainty or design parameter variation without eliminating the source of these factors [18]. This idea is graphically illustrated in Figure 2.2.

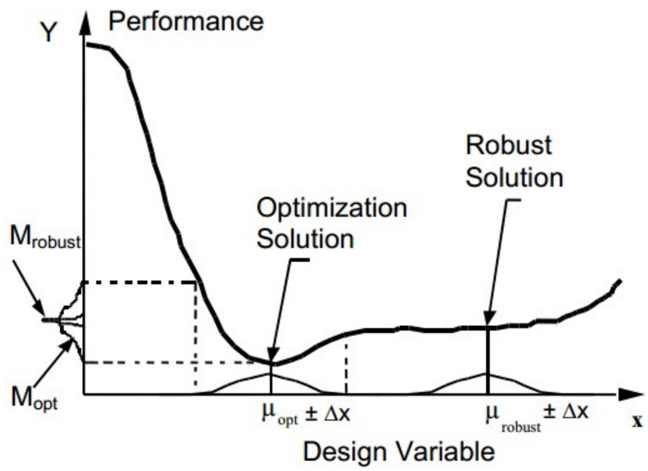

Figure 2.2: Illustration of Robust Design [18]

The figure is a representation of system performance on the y-axis, graphed against a design variable on the x-axis. In this example, the optimization goal is to minimize the performance of the system. The optimum design variable value is signified by μ_{opt}, and it correlates with the minimum performance of the system. In this case, the noise factors are represented by the distribution $\pm \Delta x$. The graph demonstrates that if the design variable deviates from its optimal value, the performance sharply increases, which is undesirable. The range of performance values that correspond to the perturbed optimal values are highlighted by M_{opt}. The goal of robust design is to create a system that is affected very little by noise factors. The robust system is represented by μ_{robust} in the figure. It should be noted that the performance value at this location is not as low as the optimal design, but at the same time the range of perturbed performance, represented by M_{robust}, is much smaller. This perfectly exemplifies the idea behind robust design – the optimum solution varies wildly if design variables deviate from optimum, while the robust solution is nearly unaffected by the same changes.

The implementation of robust design can be handled in a variety of ways. A large portion of the research in this area has been focused on the use of *design of experiments*, and *signal-to-noise ratios* to improve design and manufacturing processes [38], [39], [40]. An alternative direction has been to include robust design principles and simulations in the preliminary design process. This is done in an attempt to preemptively mitigate loss due to variations caused by noise factors, without removing the source [41].

Another important aspect of robust design is its relation to production variations inherent in the design variables, such as variation in materials or machining tolerances. This topic generally manifests itself as designing or optimizing in the presence of uncertainty. For example, Deb and Gupta combined robust design theory with multi-objective optimization to create Pareto frontiers of robust designs [42]. The goal for their approach was to have designs whose sensitivity to all relevant objectives is minimized.

Within the study of robust design, two main approaches have been identified with relation optimization. These include optimizing the expectation of the system, and optimizing the variance of performance. In his work, Das observed and explained weaknesses in both main approaches to robust design [43]. Optimizing the expectation of a function can theoretically produce designs that are not optimal at all, due to positive and negative variations in the performance canceling out, resulting in a non-robust design. In addition, minimizing the variance of the performance function can create designs that are highly robust, but are not actually optimal.

Another approach to robust design has been to treat the robustness as a second objective. Thus, the robust design problem becomes a multi-objective optimization problem: minimizing the mean of performance and minimizing the variation in performance. In general, there is a trade-off between these two objectives, and a Pareto frontier of robust designs can be formed to illustrate this trade-off between performance and robustness. For example, Jin and Sendhoff use an evolutionary algorithm to optimize the multi-objective robust design problem [44]. The authors' solution to the weaknesses observed by Das is to include a measure of variance as a second objective in the optimization of the system. The optimization problem then becomes one in which the performance is optimized and the variance is minimized. With their method, the designer is presented with a group of optimal designs with varying robustness. This allows the designer to make the decision of how much performance to trade-off for robustness.

Kumar et. al. used this multi-objective technique to design compressor fan blades that are robust with respect to their own deterioration [19]. Their approach called for the problem to be formulated as the minimization of the performance mean, and the minimization of performance variation. They used design of experiments to choose a variety of fan blades and erosion patterns which were then evaluated for their pressure characteristics. The mean and variance of these performances were used as the objectives in the optimization problem.

2.3 Epoch-Era Analysis

A third important component of this research is the idea of Epoch-Era analysis as it applies to system design. Epoch-Era Analysis is a technique that was developed by Ross and Rhodes. They state that "Epoch-Era Analysis is an approach for conceptualizing system timelines using natural value-centric timescales, wherein the context and expectations define the timescales [45]." Specifically, this refers to the break down a particular timeline, referred to as the *era*, into discrete segments, referred to as *epochs*. This technique was developed from the notion that it can be very challenging to continually make decisions over a long length of time, due to the dynamic nature of conditions and context. To improve this type of decision-making, Epoch-Era Analysis attempts to break down a timeline into separate Epochs, which are defined by constant conditions and context. The transitions between these Epochs occur when the conditions and/or context change.

This idea was applied to the design of systems by McManus et. al [46]. Although the main idea is the same, in this context Epochs discretize a system's operating regime based on three conditions: changes to context (operating environment), changes to the needs (expectations), or changes to the system. Significant changes in any of these three properties, as defined by the designer, constitute the transition to a new Epoch. This is visually depicted in Figure 2.3.

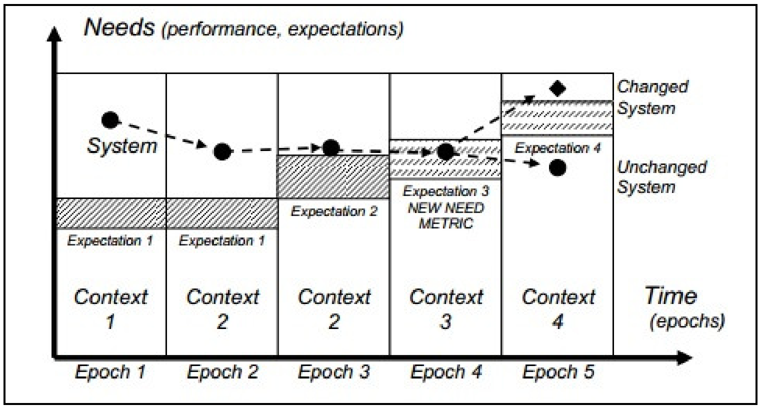

Figure 2.3: Epoch-Era Analysis [46]

Figure 2.3 illustrates a system transitioning through a number of Epochs. The shaded band represents the threshold between minimum and maximum expectations of the system. Each Epoch transition is brought about by a change in either context or needs. For example, there is a change in context between Epoch 1 and 2, causing a performance decrease because of new operating conditions. Between Epochs 2 and 3 the shift is caused by a change in the system's performance expectations. Epoch 4 illustrates a shift in both context and need, resulting in a slight decrease in performance. For Epochs 1 through 3, the system outperforms maximum expectations, but the change of expectations in Epoch 4 prevents this. Finally, in Epoch 5 both a context and need change occur again. This time, however, the system does not meet minimum performance expectation if left unchanged. This illustrates a location where reconfigurability would provide a significant benefit.

2.4 Decision Making in Design

The final component of this research addresses designer preferences in the context of making trade-off decisions when choosing a reconfigurable system. To answer this question, it is important to review the background of decision making as it pertains to design. There are a number of methods that can be employed in the decision making process. These range from simple pair-wise comparisons to hypothetical equivalents and inequivalents. Whatever method is used, the main goal is to compare the choices available and select a best choice. The process becomes harder when multi-attribute decisions, in which alternatives have many attributes to compare, are introduced. Numerous authors have investigated the decision making process and pointed out both the advantages and flaws of nearly all methods available.

The process for applying designer decisions can be split into two main stages. The first stage includes determining a representation of the designer's preferences concerning relevant attributes of the design. The second includes applying the preferences to designs in order to compare the worthiness of each design.

A common approach for the first stage in determining designer preferences is to explicitly ask them, via surveys and questionnaires [47]. These preferences can then be constructed into mathematical equations to apply to decision attributes, i.e. characteristics of the system where designer preferences affect the worthiness of a system. Another approach to capturing designer preferences is presented that aggregates the preferences of an entire design team. This work highlights the fact that many design projects are carried out not by individuals, but by a team of designers. Therefore, it is important to determine the decision preferences of the unit as a whole. Once the preferences are determined, they can be quantified in a number of ways. One such method is utility theory.

Utility theory was originally developed to aid in economic and management decision making processes [48]. However, the same theory has since been applied to decision making in the context of engineering design. One notable example is Hazelrigg's work on Decision Based Design, which breaks down the design process into a series of decisions [29]. The goal of Utility theory is to assign a utility value to the attributes of each choice. Unlike the attributes themselves, these utilities can be directly compared to one another. A variety of methods have been developed for determining utilities in the context of engineering design. All of these methods are concerned with capturing the designer's attitudes towards key decision-making components, and mathematically applying them to the decision in question.

After preferences are determined and applied, the results must be compared in order to find the most appropriate decision. One way this is accomplished is through the application of weights to the utility values, based on importance or priority. In their work, Scott and Antonsson state that these *importance weights* are intended to allow for the meaningful comparison of many options [49. They also point out that, among the multiple types of decision-making methods, *weighted sum aggregation* and *direct specification of weights* are among the most common. Both of these methods allow for the application of *importance weights* to the decision attributes, which signify their relative importance in the eyes of the decision maker.

Ultimately, there is an ever-increasing variety of methods that can be applied to the decision-making process. For example, See and Lewis examined a variety of simple decision making methods, including pair-wise comparison, ranking, normalization and arbitrary weighting schemes [50]. For each method they point out significant flaws that highly affect the outcome of the decision-making process. In the context of multi-attribute decisions, See and Lewis suggest Hypothetical Equivalents as the preferred means of establishing a more controlled decision-making process. The use of the equivalents allows for a set of weights to be constructed based on actual choices made by a decision maker, and it is more mathematically rigorous than many of the other decision making methods. In another work, the robustness associated with decisions reached using this method is examined [51]. The authors introduce a generalized process for ensuring consistent preference determination. This is then used with a Hypothetical Equivalents and Inequivalents Method case study problem to show its effectiveness.

The research which is contained in the following chapters does not examine the actual process of assigning preferences to a designer, nor is it intended to investigate variations of methods with regard to design decision-making. For these reasons, it was determined that the most common decision-making methods of utility theory and weighted sum aggregation would be used. In addition, it was decided to apply them using a set of virtual designers, because virtual designers allow for the simulation of a broader variety of designer preferences. These broader preferences can then be used to determine the effect these preferences have on the outcome of trade-off decisions.

The research highlighted in this chapter serves as a foundation for the investigation carried out in the rest of the thesis. The next two chapters will examine the two research questions presented in Chapter 1. The work in Chapter 3 examines the intersection of Reconfigurability and Robust Design with the help of Epoch-Era Analysis to implement the reconfigurable designs. In Chapter 4, the trade-off decisions associated with robust design and reconfigurable systems are examined from the perspective of decision based design.

CHAPTER 3

Reconfigurability as a Means of Mitigating Performance Variations

3.1 Introduction

The last chapter gave several examples of the performance advantages of reconfigurable systems when multiple competing objectives are considered. However, the trade-off between competing objectives is just one of the factors that must be considered when designing a multi-ability complex system. These systems will rarely operate in ideal conditions and will be subjected to a variety of uncontrollable variations. Such uncontrollable variations cause irregularities in system performance. Therefore, designers must also consider performance consistency to avoid radical, unwanted changes to system performance. One approach to mitigating performance variation in the presence of noise factors is Robust design. In this approach, a degree of system performance is sacrificed to reduce the impact of performance variations without removing their source. In this chapter, the ability of a reconfigurable system to reduce the magnitude of this performance sacrifice while maintaining consistent performance is explored.

3.2 Defining the approach

This section introduces the approach used to explore the possible benefits of using reconfigurability to mitigate performance variations in the presence of uncontrollable factors. The flow chart in Figure 3.1 highlights the major steps of this approach.

The first step in this approach is to identify the problem. This involves selecting and defining a system, and characterizing relevant uncontrollable variations that might exist. Next, the system components that can be made reconfigurability are defined. This allows the extent of reconfiguration to be determined by establishing limits and/or triggers for the act of reconfiguring. Implementation of the reconfiguration schemes allows for system performance to be analyzed, and comparisons to static designs to be made. This allows for a determination of the benefits and challenges associated with using reconfigurability in this context.

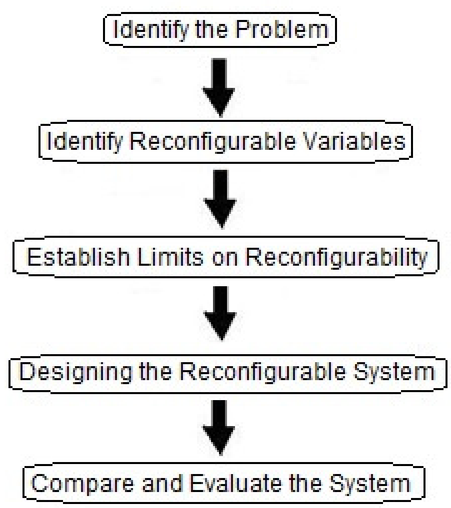

Figure 3.1: Approach used in this exploration

<u>Step 1: Identifying the problem</u>

This first step involves a thorough definition and description of the problem. This involves determining key characteristics of the overall problem:

- system design variables,
- system performance measures,
- system objective functions,
- and, relevant uncontrollable factors.

Identifying which uncontrollable factors should be included in the system model requires an understanding of the effects these uncontrollable factors have on system performance. Therefore, these terms must be defined last.

Step 2: Identify reconfigurable variables

Having identified and described the problem, the parts of the system that will be allowed to reconfigure must be specified. Since the design variables represent the physical characteristics of the system, the possible reconfigurations involve a sub-set of these variables. Therefore, a designer must decide what physical changes the system can feasibly make, and map them to location changes in the design space.

Step 3: Establish limits on reconfigurablility

To manage the complexity associated with reconfigurability, a designer may wish to limit the degree of system reconfiguration. These limitations can come in the form of constraining the:

- number of reconfigurations,
- frequency of changes,
- and the magnitude of changes.

All of these serve to reduce the complexity of a system and avoid an infinitely changing, highly complicated system. Once again, this step will differ for each system in question, and should be carried out at the discretion of the designer that is familiar with the system.

Step 4: Designing the reconfigurable system

This step involves modeling the reconfigurable system in the presence of the uncontrollable variations. Then, the reconfigurable system is optimized when uncontrollable variations are considered. The process will give the designer insight into the performance capabilities of the system and define the magnitude of physical changes that will be required.

Step 5: System comparison and evaluation

After the reconfigurable system is configured the benefits of the proposed concepts must be assessed. This is done with respect to other system concepts – both static and using different amounts of reconfigurability.

The next section of this thesis illustrates the implementation of this approach using a case study problem centered around the design of a Formula 1 style race car.

3.3 Using reconfigurability to mitigate performance loss due to uncontrollable variations

The objective of this section is to demonstrate the application of the approach presented in the previous section. The following sub-sections describe a detailed application of the approach when designing a reconfigurable Formula 1 style race car.

3.3.1 Identifying the problem

For this investigation, a simplified model of a Formula 1 style race car is used. This model has been used in prior work to demonstrate the performance advantages of a reconfigurable system when competing performance objectives are considered [8], [14], [52]. The track considered for the race car is representative of Pocono Raceway, which is made up of 3 turns and 3 straight areas. The geometry of the race track is shown in Table 3.1.

Table 3.1: Racetrack Geometry

Track	Segment	Bank (deg)	Radius (ft)	Distance (ft)
Pocono Raceway	Turn 1	14	602	1565
	Straight 1	-	-	3740
	Turn 2	8	760	1115
	Straight 2	-	-	1780
	Turn 3	6	736	1630
	Straight 3	-	-	3055

Defining the static system

In developing the model for the race car, three subsystem performance characteristics that critically affect the overall performance of the vehicle are evaluated. These subsystems are the center of gravity location, the roll stiffness distribution, and the aerodynamic downforce distribution. The aerodynamic downforce is created by the shape and angle of attack of the front and rear airfoils. Therefore, the design variables associated with this subsystem include the angle of attack, camber location, maximum camber, and airfoil thickness of both the front and rear airfoils. The roll stiffness distribution is a representation of the suspension of the vehicle, more specifically its resistance to lateral rotation, and in the model it is summed up with a single variable. The center of gravity of the race car is determined by the weight distribution of the chassis. This characteristic is represented with two variables, the chassis center of gravity and the fuel tank position. This leads to 11 design variables that are used to represent the physical configuration of the race car. A visual depiction of this and the schematic for the racecar center of gravity can be seen in Figure 3.2.

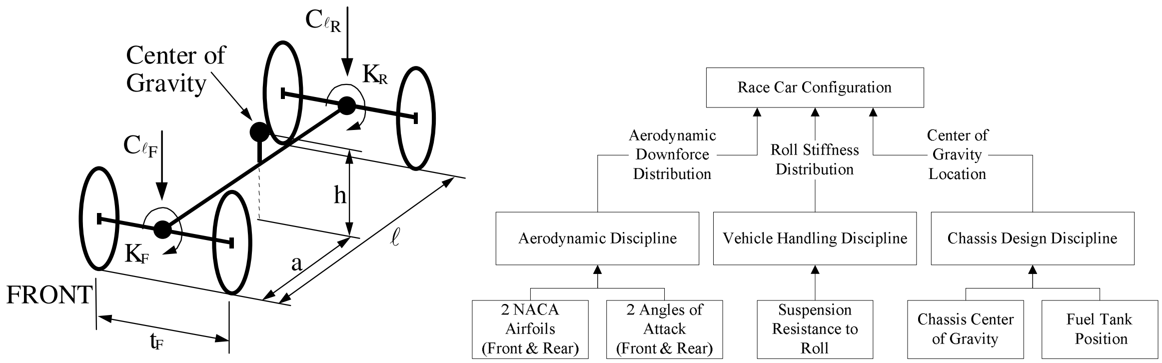

Figure 3.2: Schematic of the race car and Flow chart of design variables

Defining performance measures and objective functions

The design objective for this problem is to minimize mean lap time. A genetic algorithm is used to solve this problem as it can handle a large number of design variables and is a non-gradient based approach. Implementation of the genetic algorithm is completed using the MatLab optimization toolbox with default settings and a population size that is ten times the number of design variables.

Evaluating the performance of the vehicle requires analysis of racetrack turns and racetrack straights. This process begins by solving for the maximum steady state cornering velocity for each turn. A tire model is used to relate normal force and slip angle to the lateral force outputted by the tire [53]. This in turn influences the maximum speed of the vehicle. The tire model is based on empirical data taken from a tire-testing machine over a range of loads. It describes the nonlinear characteristics of the tire data, the variation of the tire cornering stiffness with tire normal load, and the variations in lateral force due to slip angle. Once the steady state cornering speeds have been determined for each of the turns, they can be used as endpoint velocities for the straight sections of track.

Having optimized the vehicle for steady state cornering, the next step of the analysis is to maximize performance of the vehicle as it travels down the straights. This is done by solving for the appropriate acceleration and braking profiles. The acceleration curve is created by starting at the steady state cornering speed of the previous turn, and accelerating as fast as possible down the entire length of the straight. The braking curve is then constructed by working backwards from the steady state cornering speed of the post-straight turn. The two curves are then compared and the intersection point is located. This intersection marks the optimal position on the straight where the vehicle transitions from accelerating to braking. By combining the time in each phase – accelerating and braking - a total time for each straight section can be found. Collecting the time to travel through each section of the track allows for the total lap time to be determined. The result of the optimization is the configuration of the race car that will run the fastest possible lap. An example of the acceleration and braking curves are shown in Figure 3.3.

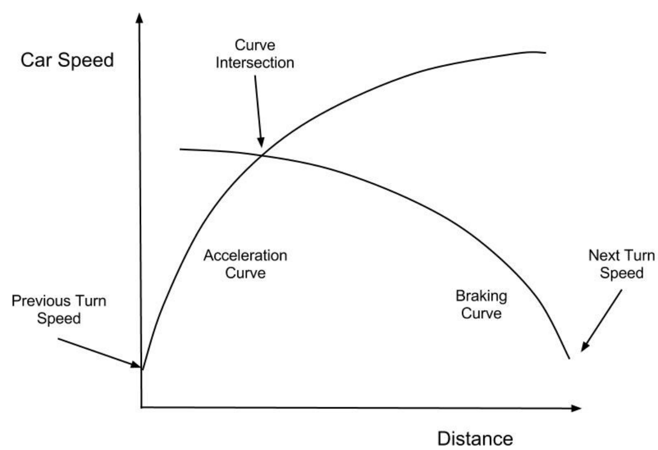

Figure 3.3: Acceleration and Braking Curve Profile

Initially, the vehicle is considered to have a static configuration. In the absence of noise factors, every lap would have exactly the same time. This static configuration serves as a benchmark for exploring the impact that uncontrollable variations and reconfigurability have on system performance. Table 3.2 displays the configuration and mean lap time of the benchmark static car.

Table 3.2: Static Car Benchmark Configuration

Design Parameter	Optimized Values
Chassis CG	0.443
Fuel Position (from rear)	0.250
K'	0.237
Front AoA (°)	15.101
Rear AoA (°)	9.351
Front NACA Airfoil	1412
Rear NACA Airfoil	4107
Lap Time (sec)	**31.749**

Identifying uncontrollable variations

The benchmark simulation of the car contained no variation in the track nor in the car itself, and every lap that the car runs is exactly the same. For this case study problem, two uncontrollable variations are considered: fuel usage and tire wear. It is also assumed that for this race, the race must be completed without a pit stop to change the tires or to refuel.

Dynamic Center of Gravity

A race car burns fuel to generate a propulsive force, and over the course of a race this will reduce the mass of the vehicle. If the fuel tank is not aligned with the chassis longitudinal center of gravity, the burning fuel will influence the overall position of the longitudinal center of gravity. In a Formula 1 style race car, fuel can account for as much as 16.3% of the total vehicle mass. Therefore, a vehicle with a fuel tank located behind the chassis center of gravity will experience a forward shift in the total center of gravity as the fuel mass is lost.

From the perspective of the designer, the amount of fuel in the tank is an uncontrollable variation, and the center of gravity of the vehicle can become an uncontrollable variable. To accommodate this uncontrollable variation into the vehicle model, the vehicle’s center of gravity is represented as two components. The first component is the longitudinal center of gravity of the chassis, and the second is the longitudinal location of the fuel tank. The combination of the chassis longitudinal center of gravity, the fuel tank location, and the current fuel mass form the dynamic longitudinal center of gravity of the vehicle. Figure 3.4 and Equations 3.1 through 3.4 illustrate how the total center of gravity can be calculated.

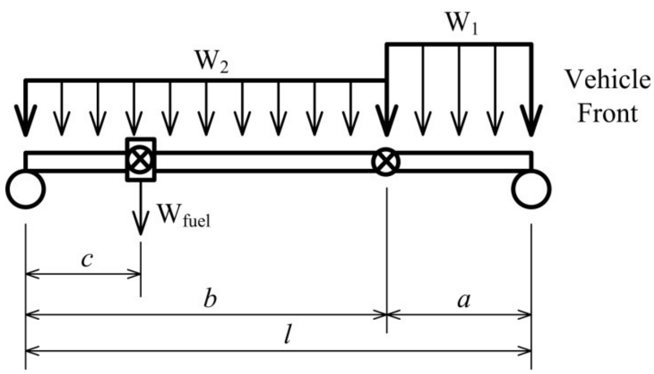

Figure 3.4: Updated Center of Gravity Diagram

$$W_2 = \frac{empty\ mass * \frac{a}{2}}{\left(\frac{a}{2}\right) + \left(\frac{b}{2}\right)} \tag{3.1}$$

$$W_1 = empty\ mass - W_2 \tag{3.2}$$

$$W_M = fuel\ position\ * W_{Fuel}(t)\ + \frac{b}{2} * W_2 + b + \frac{a}{2} * W_1 \tag{3.3}$$

$$a_{Norm} = 1 - \frac{W_M}{vehicle\ mass} \tag{3.4}$$

In Figure 3.4 and the associated Equations 3.1 through 3.4, the chassis longitudinal center of gravity is represented by the distance *a* from the front of the vehicle and the value b represents the distance from the center of gravity to the rear of the vehicle. Further, *l* corresponds to the normalized wheelbase of the vehicle. *W1* and *W2* correspond to the distributed empty weights of the vehicle fore and aft of the chassis center of gravity location. These values are calculated using the facts that the sum of moments about the center of gravity must equal zero and the *W1* and *W2* add up to the empty weight of the vehicle. When a full race investigation is conducted, the fuel tank position becomes a design variable that is bounded between 0.1 and 0.5 from the rear of the vehicle. Therefore, knowing the weight of the fuel in the tank $W_{Fuel}(t)$ at a given moment in time allows for the overall vehicle center of gravity to be calculated.

As the simulation model can become computational expensive, a simplification is made in model implementation. In a normal race, a vehicle's fuel will steadily decrease until the driver finishes the race or runs out of fuel. For this case study problem, five values of fuel percentage are used to represent the decrease in total fuel mass throughout a race. The values are 100%, 80%, 60%, 40% and 20% fuel levels. Each value will be used to simulate one lap around the track. This method gives five snapshots of vehicle performance during a race.

Tire Wear Variation

One of the most important parts of the race car are the tires because they are the only component that connects the vehicle to the ground. Therefore, all vehicle design changes must be made with the operating conditions of the tires in mind. Similar to the amount of fuel, the tires experience changes over the course of a race. Most notably, they wear out and lose effectiveness in being able to sustain loads imparted by the vehicle. As a race progresses, the vehicle will not be able to perform to the same standards as at the beginning of the race.

To implement tire wear into the model, a tire wear variable is used. This variable is used in conjunction with empirical data to determine the lateral forces that the tires are able to produce. The variable becomes a coefficient for the empirical data representing the reduced ability to provide forces from the tires. It is assumed that at the beginning of a race, the tires are 100% effective, but by the end of a race, they retain only 80% of their initial effectiveness. As is the case with the fuel usage variation, to reduce computational intensity, five snapshots of the race are used. The values for tire effectiveness are 100%, 95%, 90%, 85% and 80%.

Having identified sources of variation the next step is to determine their effects on the system in question. This involves implementing and simulating the variations in the system model. These results help to create a benchmark for use in determining the effectiveness of the reconfigurable system.

Simulating the effects of uncontrollable variations

To test the effects of the proposed variations alone, the vehicle model was simulated using only one turn of the race track. By doing so, the effects of the variation can be investigated without any confounding from the multi-objective nature of the track itself. The first turn of the track, with a radius of 602 feet and a bank angle of 14 degrees, was chosen for this investigation.

The variation due to fuel usage is tested first using the original optimized configuration. Table 3.3 shows the results of the single turn dynamic fuel mass simulation.

Table 3.3: Single Turn with Decreasing Fuel Mass

Lap	Fuel Mass (slugs)	a'	Turn 1 Time (sec)
Lap 1	8.110	0.493	4.258
Lap 2	6.488	0.484	4.348
Lap 3	4.866	0.475	4.228
Lap 4	3.244	0.465	4.169
Lap 5	1.622	0.455	4.201
Time to Complete Turn 1 Through 5 Laps			**21.204**

Because a vehicle starts the race with ideal conditions concerning fuel mass and tire effectiveness, the first turn time represents the time that would be posted for all five laps if the uncontrollable variations were not included. As illustrated in Table 3.3, burning fuel moves the center of gravity forward with each lap. This is shown by the variable a' which represents the longitudinal center of gravity of the vehicle. The results in Table 3.3 also show that the time to complete the turn changes with each lap. This is due to changes in both the center of gravity and total vehicle weight. The lighter the car is, the faster it is able to go around the turn while maintaining traction. In addition, the center of gravity affects the race car's ability to brake into the turn and maintain traction through the turn. Therefore, it can be seen that the addition of this uncontrollable variation introduces perturbations in the performance of the vehicle.

The second variation introduced is the reduced tire effectiveness due to tire wear. Once again, this variation is tested on one turn across five different laps to understand how vehicle performance changes when the tires become less effective. The single turn times are shown in Table 3.4.

Table 3.4: Single Turn with Decreasing Tire Effectiveness

Lap	Tire Effectiveness	Turn 1 Time (sec)
Lap 1	1.00	4.258
Lap 2	0.95	5.871
Lap 3	0.90	6.126
Lap 4	0.85	6.454
Lap 5	0.80	6.839
Time to Complete Turn 1 Through 5 Laps		**29.547**

The first turn time represents the time that would be posted for all five laps without variation. Therefore, these results show that the inclusion of tire wear negatively affects performance.

The next step of the process was to combine the dynamic fuel model with the tire wear model. This was done using the original optimized configuration simulated over the same five laps for a single turn of the race track. The results of this simulation can be seen in Table 3.5.

Table 3.5: Single Turn with Combined Model

Lap	Fuel Mass (slugs)	a'	Tire Effectiveness	Turn 1 Time (sec)
Lap 1	8.110	0.493	1.00	4.258
Lap 2	6.488	0.484	0.95	4.615
Lap 3	4.866	0.475	0.90	6.201
Lap 4	3.244	0.465	0.85	6.514
Lap 5	1.622	0.455	0.80	6.873
Time to Complete Turn 1 Through 5 Laps				**28.461**

Once again, the trend set by the tire wear model continues, and the performance is negatively affected by the addition of the uncontrollable variations.

Another approach to examine the effects of the uncontrollable variations on the performance of the vehicle is to use the standard deviation of the lap times. Table 3.6 shows the total time to complete the turn five times, the average time to complete the turn, and the standard deviation of the turn times.

Table 3.6: Single Turn Performance Comparison

	No Variations	Tire Wear Model	Fuel Usage Model	Combined Model
Total Turn 1 Time-5 Laps (sec)	21.29	29.547	21.204	28.461
Mean Turn 1 Time (sec)	4.258	5.909	4.241	5.692
Standard Deviation (sec)	0	0.992	0.068	1.177

The results in Table 3.6 show that while the tire model produces the most negative effect on the race car’s performance from a lap time perspective, the combined model induces the most variation in the performance. For comparison, the original model has zero standard deviation between laps. Also, the time to complete the five laps in the combined model is actually lower than the times simulated with the tire wear model alone. This is because as the fuel is used through the course of the race, the car becomes lighter and therefore faster. Alternately, in the simulation with tire wear alone, the car maintains a constant weight throughout the race and the tires become less effective.

While it is important to examine the effects of the variations in the context of a single turn, it is more useful and more realistic to examine them over the course of a race. For this study, a full race is assumed to be an aggregation of the five full laps. In this model formulation, the variations are updated each lap to simulate the passage of time during the race.

To start this analysis, the original static design was simulated without taking into account the uncontrollable variations. A genetic algorithm was used to optimize the static car over the five lap race. The end result was five identical laps of 31.749 seconds, for a total race time of 158.745 seconds. This result serves as the benchmark against which the updated model will be tested.

The next step was to include the uncontrollable variations. Once again, fuel is assumed to decrease by 20 percent each lap after starting at 100 percent in the first lap. The tire wear coefficient decreases by 5 percent each lap after starting at 100 percent.

The optimized static design is simulated in each variation model as shown in Table 3.7. From these results, it is found that the model including only dynamic fuel improves the performance of the original car. This is due to the fact that the car becomes lighter as the race progressed. On the opposite end of the spectrum are the tire wear model and the combined model simulations. When simulated in these models, the original optimal design sees a significant decrease in performance. For tire wear alone, there is a 22.4 percent increase in the time, and in the combined model there was a 12.7 percent increase. Once again, the car in the combined model did better than tire wear alone likely because it became lighter as the race went on.

Table 3.7: Full Race Comparison

Lap	Lap Times (sec) Original Model	Tire Wear Model	Fuel Usage Model	Combined Model
Lap 1	31.749	31.749	31.749	31.749
Lap 2		35.411	31.936	32.589
Lap 3		41.139	31.644	35.773
Lap 4		42.306	31.460	36.804
Lap 5		43.713	31.546	42.029
Race Time (sec)	**158.745**	**194.319**	**158.335**	**178.944**
Total Time Difference	**-**	**22.4%**	**-0.3%**	**12.7%**

Another important aspect of the updated models is the variety among the lap times which would indicate whether the uncontrollable variations cause variations in the performance of the vehicle or not. The amount of variations can be quantified by examining the standard deviation of the lap times. Table 3.8 provides an overview of the relevant information from the simulations.

Table 3.8: Summary of Variation Simulations

	Original Model	Tire Wear Model	Fuel Usage Model	Combined Model
Race Time (sec)	158.745	194.319	158.335	178.944
Mean Lap Time (sec)	31.749	38.864	31.667	35.789
Lap Standard Deviation (sec)	-	5.076	0.185	4.078

With the system in question characterized and modeled, and the uncontrollable variations identified, the next step is to analyze the system for reconfigurability. This is the step that will determine which aspects of the reconfigurable system will be allowed to change after deployment.

3.3.2 Identify reconfigurable variables

In this step, the designer would examine the system to determine the system aspects that could be allowed to feasibly reconfigure. This process involves separating the aspects of the system that would likely help mitigate performance variations and those that would have no effect. One method to address this is the Variable-Segregating Mapping Function introduced by Khire and Messac [30]. This optimizes both the system and the number of variables allowed to adapt at the same time using a bi-objective approach. While this is an option for this step in the approach, it adds an additional layer of optimization and complexity.

In this case study, the system is represented by a model that combines eleven design variables to determine the overall performance of the vehicle. Therefore, the reconfigurable aspects could be the entire set, or a subset, of the design variables. Since the model is already streamlined to include only variables that affect system performance, all of the variables are assumed to be relevant for their potential to help mitigate variations rather than applying another layer to the optimization process.

The other consideration is the physical possibility that such a change can be made. For example, in the case study model, one of the design variables is the center of gravity of the chassis. While it is true that this variable will have an effect on system performance, questions arise with how easily this can be changed during operation. Accomplishing the task of moving the center of gravity would require either large shifts of mass or small shifts of very large mass. Both of these options seem infeasible when considering the architecture of the racecar itself. Therefore, the chassis center of gravity is not allowed to change during operation because it would be prohibitively complicated to implement. The other design variables would be more easily changed, and so they will be allowed to change during reconfigurations.

Determining the changeable aspects of the system in question is one important step in developing a reconfigurable system. Another step concerns the amount of reconfigurability that is allowed to take place. This involves determining reconfiguration limits on such as the number of reconfigurations, amount of change in each reconfiguration, or frequency of reconfiguration.

3.3.3 Apply limits to reconfigurability

The next step is to provide limits for the reconfigurable system. A variety of limits may be applied to the system that can address different aspects of the reconfigurations. For example, if a designer determines that it is very expensive to implement rapid changes in the system, they could limit the frequency at which the changes occur. On the other hand, large changes in the design variables may be very challenging to implement. In this scenario, limits could be applied on the amount of change available to each variable.

To apply appropriate limits to the reconfiguration of the system, the amount of reconfigurability needs to be quantified. The quantification of reconfigurability allows for the differentiation between reconfigurable systems. Prior work by Gumasta, used Multi-attribute Utility Theory to establish an index based on modularity, convertability, diagonsability, and scalability [54]. However, this work was primarily focused on categorizing reconfigurability in manufacturing systems and may not apply to all reconfigurable systems. To this end, it is necessary to establish a more generalizable method.

If the operating regime of a system can be discretized, the number of unique end states can be counted to give a numerical value for reconfigurability. The total number of unique end states then becomes a numerical representation of the level of reconfigurability present in the system. However, this method of quantifying reconfigurability is based on the discretization of the operating regime. To do this in a controlled manner, this work builds upon Epoch-Era analysis introduced by McManus et. al [46]. Here, Epochs discretize a system's operating regime based on changes to context (operating environment), needs (expectations), or the system itself. Significant changes in any of these three properties, as defined by the designer, constitute the transition to a new Epoch. A visual depiction of this can be seen in Figure 3.5.

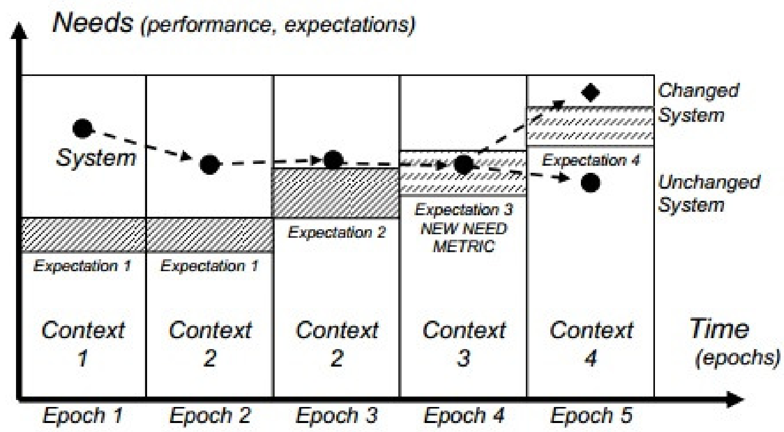

Figure 3.5: Example of Epoch Transition [46]

Figure 3.5 illustrates a system transitioning through a number of Epochs. The shaded band represents the threshold between minimum and maximum expectations of the system. Each Epoch transition is brought about by a change in either context or needs. For example, there is a change in context between Epoch 1 and 2, causing a performance decrease because of new operating conditions. Between Epochs 2 and 3 the shift is caused by a change in the system's performance expectations. Epoch 4 illustrates a shift in both context and need, resulting in a slight decrease in performance. For Epochs 1 through 3, the system outperforms maximum expectations, but the change of expectations in Epoch 4 prevents this. Finally, in Epoch 5 both a context and need change occur. This time, however, the system does not meet minimum performance expectation if left unchanged. This illustrates a location where reconfigurability would provide a significant benefit.

As reconfigurability allows a system to change its physical configuration, a given system will be made up of a variety of unique configurations. Each of these arrangements can be considered a state of the system, and the act of reconfiguring moves the system from one state to another. Ideally, each state would be the optimum design for a specific Epoch, and the transition between states occurs at the boundary between Epochs. In this context, identifying the level of reconfigurability can be accomplished by counting the number of possible unique end-states.

As an example, an umbrella can be broken down into two main Epochs. In the first Epoch, the expectation on the system is to be small and storable. When it rains, the expectation of the system shifts to becoming a rain shield. The transition between the two Epochs serves as a transition from one unique state to another. Therefore, the umbrella can be considered a two state system: having a closed state (Epoch 1) and an open state (Epoch 2).

In general, the benefits of reconfigurability occur because the system is allowed to transition to an optimal configuration for every situation. This work manages reconfigurability using reconfiguration schemes to limit the number of possible system end-states. The objective of limiting the total number of end-states is to reduce the complexity of the system. Therefore, to retain the benefits associated with reconfigurability, it is important that Epoch transition is carefully defined. Appropriate transition points can be chosen by examining system properties such as multi-ability expectations or performance effects of uncontrollable variations.

The first scheme considered in this work addresses the multi-ability requirement of a race track. It should be noted that while the test track is made of three distinct turns and three straights, this scheme does not account for the different properties between each turn or straight. Instead, the two states address the act of turning or the act of driving on a straight section. The optimal configuration for each of these objectives is likely very different, and therefore the system as a whole would benefit from reconfigurability for these transitions. By following this line of thinking, the Epoch transitions can be placed where the system context changes or where the car switches from a turn to a straight and vice versa, illustrated in Figure 3.6. The proposed reconfiguration scheme is a two state model which includes a state for turning and a state for driving on straight sections.

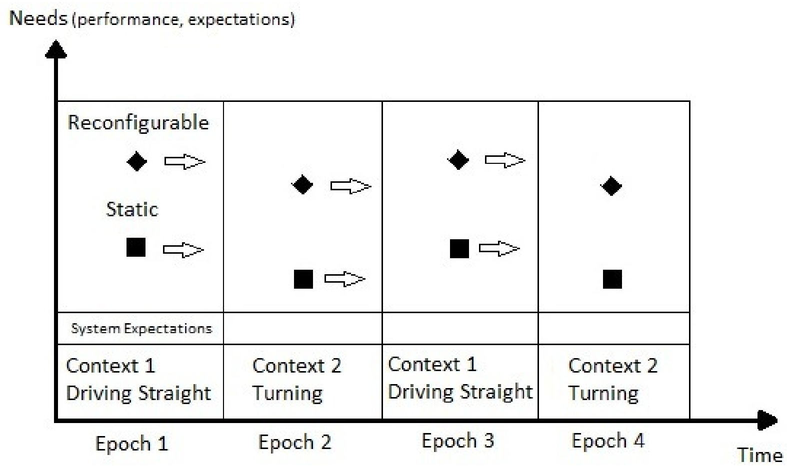

Figure 3.6: Two State Epoch Transition

The Epoch breakdown for a two state system is illustrated in Figure 3.6. The four Epochs shown are just an excerpt of the operating regime of the race car, but they serve to exemplify the multi-ability Epoch transition criteria. The context shift caused by a change from driving straight to turning is the trigger for changing Epochs. In the figure, the squares serve as a representation of a static system, and the diamonds represent a reconfigurable system. Comparatively, the reconfigurable system has a higher performance than the static system because it is allowed to maintain the a better configuration for each Epoch. Conversely, the static system is designed for its entire lifecycle.

The second scheme studied in this work aims to address robustness. Because the model has been simplified such that the uncontrollable conditions update after every lap, it follows that there is a noticeable change in performance characteristics for each lap. Epoch transitions would occur for context changes that the system encounters, and here the context changes are due to the uncontrollable variations present in the model. In reality, a race car would not experience these discrete jumps in conditions or performance. Because the simulation is carried out over the course of five laps, this choice of Epoch transition criteria produces a five state reconfiguration scheme. Figure

3.7 illustrates an excerpt from the operating regime of the five state system showing four Epoch transitions.

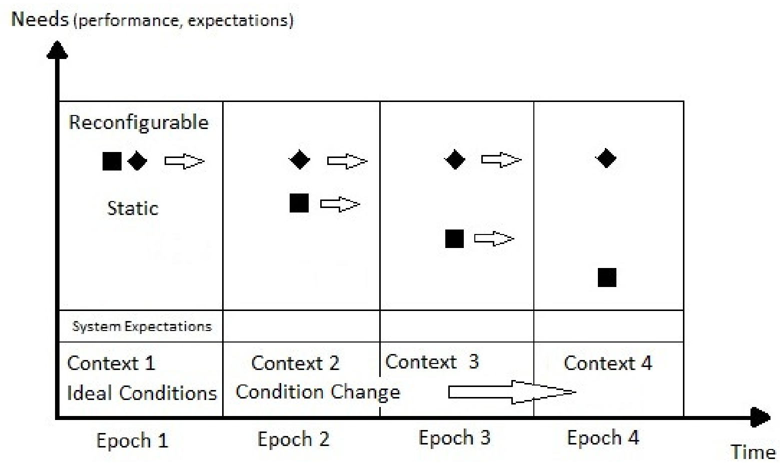

Figure 3.7: Five State Epoch Transition

The five state transition scheme is displayed in Figure 3.7. Once again, the four Epochs shown in the figure are an excerpt of the entire life of the system. The context shift is caused by a noticeable change in the operating conditions of the system. This time the squares representing the static system lose performance with each Epoch transition due to the effects of the uncontrollable variations. On the other hand, the reconfigurable system can maintain a consistent level of performance even in the face of changing conditions.

The third scheme examined is a combination of the two previous schemes, and it aims to address both multi-ability and robustness requirements It incorporates the Epoch transitions from both of the previous strategies by transitioning for changes in needs and changes in context. The Epochs would again transition for each new objective as in turns and straights, but unlike the two state scheme the Epochs for each lap would be differentiated from one another. This translates to a turning state and a straight state that are unique for each lap or ten unique states.

The different approaches to establishing Epoch transitions and limiting reconfigruability provide a number of examples of how reconfigurability can be applied to address performance variation. Once the limits on reconfigurability are established, the reconfigurable systems can be implemented and tested.

3.4 Implementing and testing reconfigurable systems

Having defined the different transition schemes, the next step is to implement them and assess system performance. Because the vehicle is now reconfigurable, each unique configuration must be optimized. This produces the best configuration possible for each stage of the race. The stages are then combined to form an entire race. Figure 3.8 provides an outline of this process.

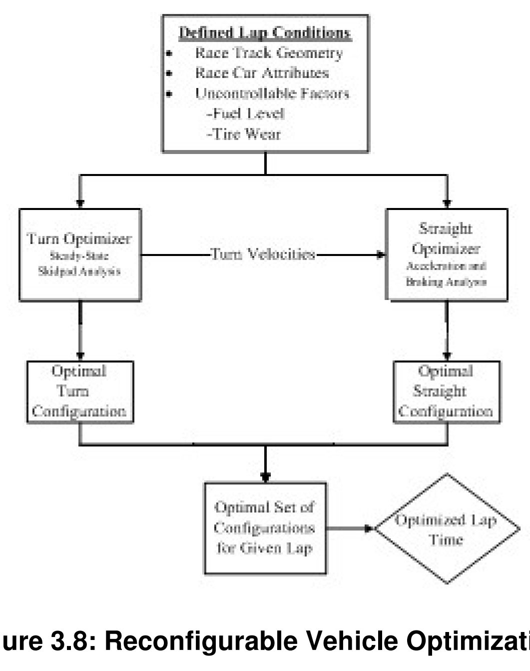

Figure 3.8: Reconfigurable Vehicle Optimization

As show in Figure 3.8, the optimization process is broken down into a unique configuration for each of the subsections of the track, turns and straights. The process in the chart shows the optimization of a single lap, and therefore in this problem it is repeated five times. This general process is used for all of the reconfiguration schemes with specific differences for each. For example, the two state scheme uses the same optimal turn and straight configurations for each of the five laps, but the ten state scheme introduces a new optimal configuration for each lap. Finally, for the five state scheme, the turn configuration and straight configuration are identical which creates one configuration for each lap. This configuration is changed for each lap.

As shown in the figure, the reconfigurable optimization is broken down first into a turn optimization and a straight optimization. Instead of one car being optimized through all sections of the track, a new set of eleven design variables is optimized for each segment. Since the design variables represent the physical aspects of the system, the different sets of variables represent the different configurations that the car takes throughout the race. Also, because the vehicle consists of all of the configurations, each new set of design variables increases the total number of variables for the vehicle. For example, the two state scheme contains twice as many design variables as the static system. This trend holds true for the other schemes as well.

In this work, two objectives functions are used to characterize the performance of a design. Typically, the main goal of winning any race is to complete it in fastest time possible. As only five laps are used in this simulation, a surrogate function for fastest time is to minimize the mean lap time. Now, consider a scenario where the racing body gives points not only for where you finish in a race, but also awards points to the vehicle with the most consistent lap time. While this is a hypothetical example, race governing bodies such as Indycar give points to the driver who leads the most laps and the driver who captures the pole position [55]. In this situation it would be beneficial to the racecar driver to have a car that will perform consistently for all conditions encountered in a race. They will then have to make minimal adjustments to racing strategy, and be able to post consistent lap times. Therefore, the second objective characterizes the robustness of the design, and is quantified using the standard deviation of the lap times. Here, a smaller standard deviation signifies a more consistent design.

Opitmization of the vehicle configuration is completed using the multi-objective genetic algorithm toolbox in Matlab. Default settings are used for this optimization, and the initial population is set to ten times the number of design variables.. A MOGA is used because of its ability to handle large sets of design variables and the ability to optimize in situations where gradient information is non-existent or too computationally expensive to determine.

The results for the two-state scheme are displayed in the Figure 3.9. In this figure performance of the configurations range from mean lap times of 30.22 to 31.46 seconds and standard deviations from 0 to 0.0895 seconds. The higher lap times are associated with the lower standard deviations which illustrates the trade-off between performance and consistency.

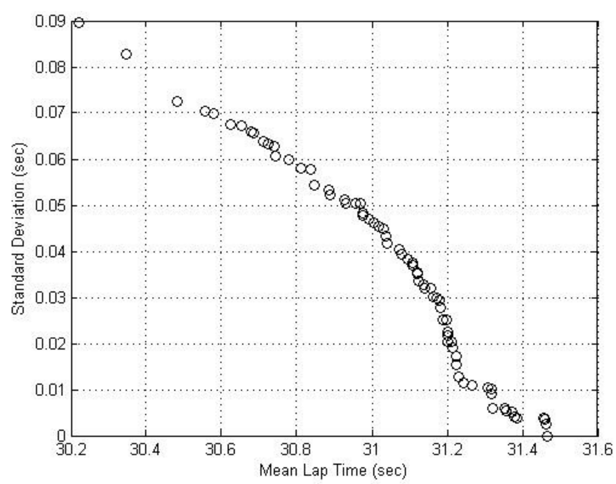

Figure 3.9: Two State Pareto Frontier

An important outcome from this simulation is that all of the resulting configuration have a lower mean lap time than the original static system. Results from the five-state scheme optimization are shown in Figure 3.10.

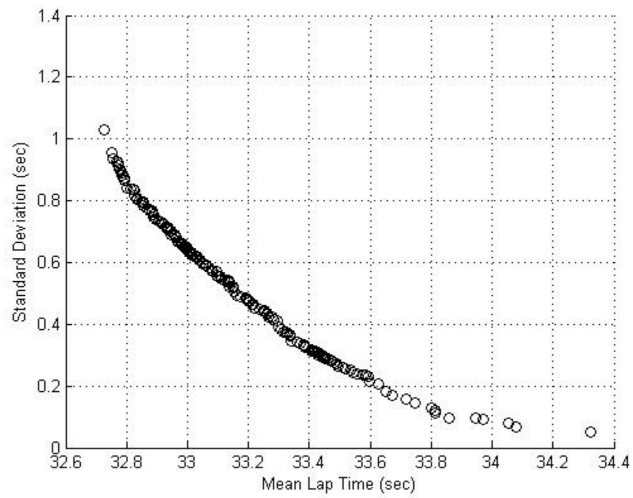

Figure 3.10: Five State Pareto Frontier

This optimization produces configurations with performances ranging from 32.72 to 34.32 seconds and standard deviations from 0.05 to 1.03 seconds. Once again, the lower standard deviations are associated with the higher lap times. Unlike in the two-state simulation however, not all of the mean lap times are better than the original static model. Finally, the results of the ten-state scheme, are displayed in Figure 3.11. This reconfiguration scheme shows the best mean lap performance with times ranging from 29.31 to 32.23 seconds. Also, the standard deviations associated with these configurations ranged from 0.0021 to 0.8076 seconds. These results place the ten state scheme in the same range, in terms of standard deviation, as the two state scheme, but there are better lap times associated with the ten state scheme. Very much like the two state results, all of the mean lap times are better than the original static vehicle.

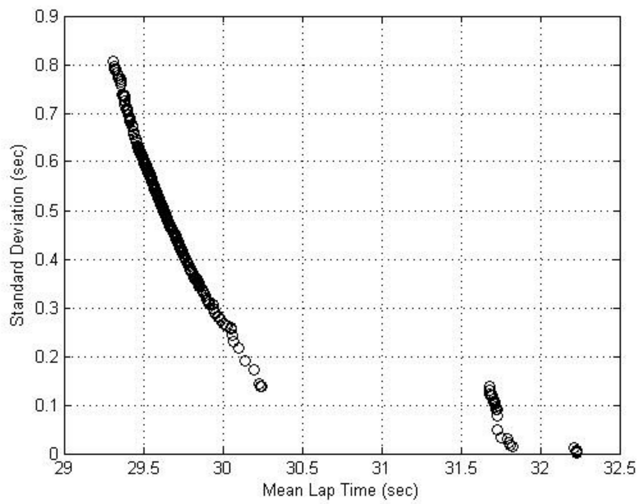

Figure 3.11: Ten State Pareto Frontier

It can be seen that there is a gap between the cluster of points on the right, and the cluster on the left side of the frontier. After examining the five designs closest to each side of the gap, a clear difference can be seen. The designs on the left side of the gap had a higher average magnitude of change between consecutive states for eight of the nine state transitions than the points on the right. Additionally, on the left side 21 of the 101 design variables underwent changes greater than 0.3 out of 1. This is compared to the right side which contained only six of these large transition variables. Therefore, it can be seen that the gap coincides with an increase in the amount of change between states.

While some insight can be gained by examining the three Pareto Frontiers individually, comparing them to each other is also beneficial. Figure 3.12 displays the three Pareto Frontiers on the same graph, with the x-axis representing mean lap time and the y-axis representing the standard deviation of the lap times. One conclusion that can be drawn from Figure 3.12 is that the two-state and ten-state schemes outpace the five state scheme along the mean lap time axis. In terms of variation in the lap times though, all three schemes produce designs that have very small standard deviations.

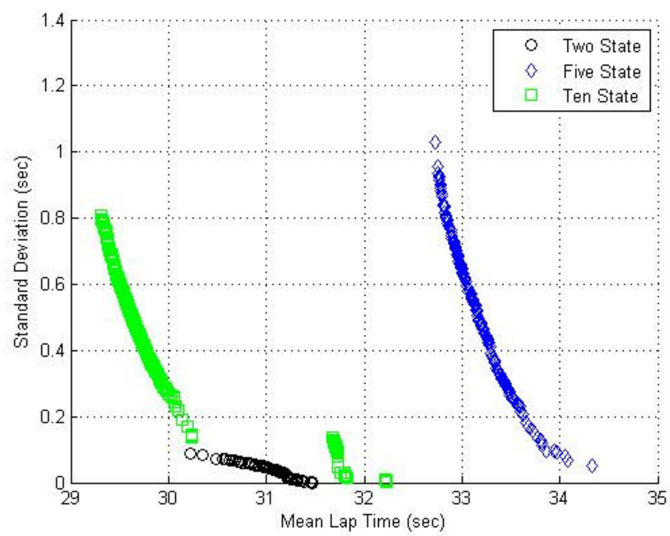

Figure 3.12: Three Pareto Frontiers Side by Side

It is also beneficial to compare the reconfigurable systems to a static one designed under the same multiobjective problem formulation. The Pareto frontier for this static vehicle can then be directly compared to the three frontiers created by the reconfigurable systems. Figure 3.13 displays the three reconfigurable Pareto Frontiers along with the results for the static vehicle.

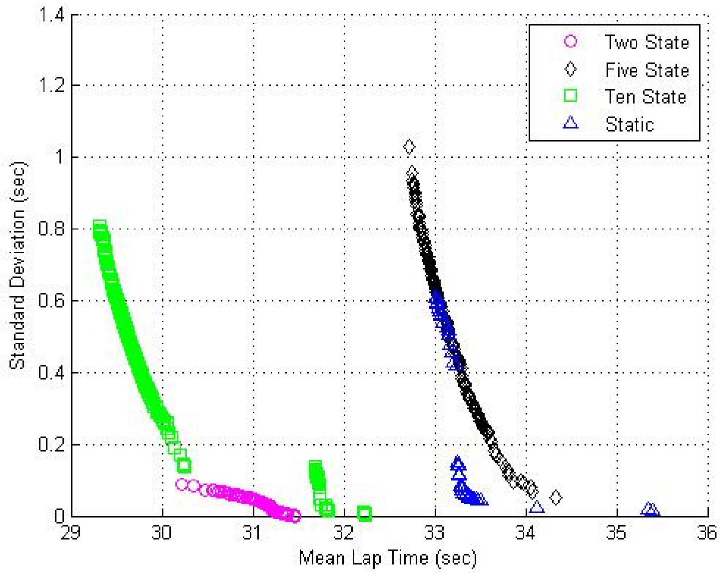

Figure 3.13: Pareto Frontier Comparison

A first conclusion drawn from these results is that the two- and ten-state schemes produce similar variation in lap times while increasing system performance over both the five-state scheme and the static vehicle. This indicates that the particular reconfigurability limitations chosen for the five-state scheme do not offer as much benefit as the other reconfiguration schemes.

3.5 Conclusions

In this chapter, three reconfiguration schemes were created and explored to examine how various amounts of reconfigurability can be used to handle the trade-off between performance and consistency. This comparison was accomplished by comparing the resulting Pareto frontiers to that of a static vehicle using a multi-objective approach for robust design. The approach considered mean lap time and the standard deviation of the lap times as problem objectives to demonstrate the trade-off between performance and consistency. The benchmark comparison showed that two of the reconfiguration schemes, two-state and ten-state, dominated all points on the static system frontier. The five-state scheme did not show the same improvement over the static system. In terms of mitigating the effects of variations, the two- and ten-state schemes were both at least as effective as the static benchmark system at producing consistent systems. Additionally, these two schemes increased the performance of the system over the static benchmarks. Therefore, these results suggest that the performance sacrifice often made to increase consistency can be negated by reconfigurability.

While this study has demonstrated that reconfigurability can be used to mitigate variations, the disparity among the different reconfiguration schemes implies that the decision of how much reconfigurability to implement is an important one. This becomes the foundation for another trade-off decision relating to performance and system complexity. This tradeoff is explored in the next chapter which explores the second research question posed in Chapter 1.

CHAPTER 4

Design Choice Sensitivity to Designer Preferences

4.1 Discussion of designer choice

In the previous chapter, implementation schemes designed to control how often the system was capable of changing were developed and analyzed. Performance analysis focused on improving system performance while minimizing performance variation. Epoch-Era analysis was used to establish limitations on how often the reconfigurable system was allowed to change. Three reconfiguration scheme were identified and a case study problem was developed around the design of a Formula 1 style race car. This results from this chapter revealed that two of the schemes completely dominated the static system, and the third produced designs that were nearly equivalent.

This investigation supports the hypothesis that varying levels of reconfigurability can improve both performance and consistency. It also introduced two problem formulation components where designer preferences concerning trade-offs would affect the choice of a reconfigurable system: the performance versus consistency trade-off, i.e. choosing a point on the Pareto Frontier, and the performance versus complexity trade-off, i.e. choosing an appropriate reconfiguration scheme. The first involves choosing from a Pareto frontier of possibilities that explores the trade-off decision between performance gain and performance consistency. The second involves navigating the tradeoffs that occur when considering system complexity and possible performance gains. Both of these decisions are influenced by the notion of "preference," whereby a designer assigns priorities to a set of design choices. These decisions require an understanding of the system in question, the performance advantages of reconfigurability, and the designer's preferences towards complexity [47].

The objective of this chapter is to explore how a designer's choice of reconfigurable system is influenced by his/her preferences for system attributes. This investigation is carried out in two stages. The first stage identifies the aspects of the system where a designer would express preference, as well as how appropriate quantitative measures are defined. For this work, the main considerations for a designer are the performance increase offered by a reconfigurable system and the added complexity introduced by the configuration changes. These two aspects must be weighed against each other to determine the level of reconfigurability to be implemented in the system. The second stage involves determining how different preference structures might influence the reconfigurable system that is chosen.

4.2 Defining appropriate complexity measures

System complexity often results in increased cost which can affect system value and the likelihood of implementing system reconfigurations [8], [13]. However, not all forms of complexity contribute equally to the value of a system. For example, incorporating reconfigurability could require specialized materials and computational complexity associated with advanced control algorithms. A designer would likely be concerned with a variety of complexity sources when weighing the advantages and disadvantages of a system. To this end, this section introduces several high-level complexity measures that can be used in the early stages of design to compare reconfigurable system concepts. The complexity measures considered in this work include the:

- number of configuration changes,
- frequency of the configuration changes,
- magnitude of the configuration changes.

The first complexity measure is the total number of times a system undergoes a configuration change during a specified operating cycle. The assumption behind this measure is that the more the system changes, the more difficult the system will be to design. For the race car example, the number of configuration changes for the two state vehicle may be more easily visualized by Equation 4.1. Here, the vehicle switches every time the vehicle transitions from a turn to a straight, and vice versa.

$$NC = (S + T) * L \tag{4.1}$$

In this equation, *NC* represents the number of changes during a race. Since the two state scheme transitions between states every turn and straight, these two values can be added to find the changes that occur in a single lap. These are represented as *S* for a straight and *T* for a turn, respectively. Finally, the race consists of a number of laps, *L*.

A second complexity measure corresponds to the frequency at which the changes occur. The implication for this measure is that a system is more complex if the frequency of change is higher. Causes of this increased complexity could include higher resolution sensors, more powerful control schemes, and more advanced actuation technologies. In this work, measuring the frequency of change is done by calculating change velocity. Change velocity is found by dividing the total number of configuration changes during a period of operation by the total time of the operation period. The formulation for this complexity measure is shown by Equation 4.2. Here, *CV* represents change velocity, *NC* represents the total number of changes, and *TT* represents the total time of the operating period. This formulation leads to a result with units of changes per second, and it is assumed that a larger value of *CV* corresponds to a more complex system.

$$CV = \frac{NC}{TT} \tag{4.2}$$

Finally, the third complexity measure accounts for the magnitude of motion within the design space. It is assumed that large design variable changes would be more complex to implement than small changes. To measure this factor, the maximum range of change for each design variable is calculated, as shown in Equation 4.3. For this measure, the scope of investigation requires assimilating information from multiple disciplines, reflecting the importance of treating complex system design as a multi-disciplinary endeavor.

$$\overrightarrow{DVC} = Max(\overrightarrow{DV}) - Min(\overrightarrow{DV}) \tag{4.3}$$

In this equation, $\overrightarrow{DVC}$ is the range of change for the design variable in question. $Max(\overrightarrow{DV})$ and $Min(\overrightarrow{DV})$ are the maximum and minimum numerical value for each design variable when the entire set of configuration changes are considered. The difference between these values represents the magnitude of change each design variable undergoes. As each design variable may be changed independently using a different solution principle, these measures are not aggregated.

The three complexity measures defined above represent important tradeoffs a designer must make when selecting an appropriate reconfiguration scheme. Tables 4.1, 4.2a and 4.2b list three of the reconfiguration schemes from the previous chapter and their associated complexity measures.

Table 4.1: Schemes and Results for the First Two Complexity Measures

Reconfiguration Scheme	*NC*	*CV*
2 State	30	0.1996
5 State	5	0.0312
10 State	30	0.2066

Table 4.2a: Maximum Variable Change Complexity Measure

Reconfiguration Scheme	*Fuel Tank Position*	*Roll Stiffness*	*Front AOA*	*Front Max Camber*	*Front Camber Dist.*	*Front Max Thickness*
Two State	0.899	0.111	0.819	0.764	0.416	0.567
Five State	0.463	0.408	0.776	0.539	0.559	0.828
Ten State	0.918	0.250	0.971	0.780	0.956	0.778

Table 4.2b: Maximum Variable Change Complexity Measure Continued

Reconfiguration Scheme	*Rear AOA*	*Rear Max Camber*	*Rear Camber Dist.*	*Rear Max Thickness*
Two State	0.328	0.322	0.083	0.381
Five State	0.344	0.271	0.671	0.734
Ten State	0.344	0.322	0.146	0.599

Once values are defined for the identified complexity measures, focus can turn to selecting the most appropriate reconfiguration scheme. To do this, designer preferences will be simulated and solution sensitivity will be explored. The simulation process is outlined in the next section.

4.3 Simulating designer preferences

To determine how preferences relating to the complexity measures influence the choice of reconfiguration scheme, different combinations of designer preferences need to be explored. These preferences can be combined with measures of system performance and complexity to assess the utility of a proposed design [29]**.** A design decision can then be made by choosing the concept with the largest utility (or value).

Research in the engineering design community has previously identified approaches that successfully capture design utility curves and variable weighting schemes [49], [51], [56], [57]. The research in this work does not focus on the method for obtaining such data, but instead focuses on how this information influences concept selection. Utility curves can be constructed to represent three different risk profiles: risk averse, risk neutral and risk prone.

Assuming that a smaller value of the independent variable is more preferred, and that the dependent value is normalized between 0 and 100, a risk averse utility curve starts at 100 and decreases at a faster rate initially as the independent variable increases. A risk neutral curve is linear as it decreases. Finally, the risk prone curve decreases at a slower rate initially as the independent variable increases. The equations used in this work for the three curves are given by Equations 4.4 – 4.6:

$$\text{Risk Averse:} \quad US = 100 * e^{\left(\frac{-4.6}{V_{max} - V_{min}}\right) * (AV - V_{min})} \tag{4.4}$$

$$\text{Risk Neutral:} \quad US = \left(\frac{-100}{V_{max} - V_{min}}\right) * (AV - V_{min}) + 100 \tag{4.5}$$

$$\text{Risk Prone:} \quad US = \left(\frac{-100}{(V_{max} - V_{min})^2}\right) * (AV - V_{min})^2 + 100 \tag{4.6}$$

For these equations, US is the utility score for each attribute as obtained from the utility curve. V_{max} and V_{min} are the upper and lower bounds of the attribute in question, and AV is the value of the attribute being examined. The equations are set up such that for each attribute the curve is bounded by the attributes maximum and minimum values. Designs that perform the worst for an attribute are given a value of 0; designs that perform the best are given a value of 100. This format is effective because the designer intends to minimize the attributes. For example the values of mean lap time, standard deviation, or complexity, are all small in an ideal system. Examples of these three curves can be seen in Figure 4.1.

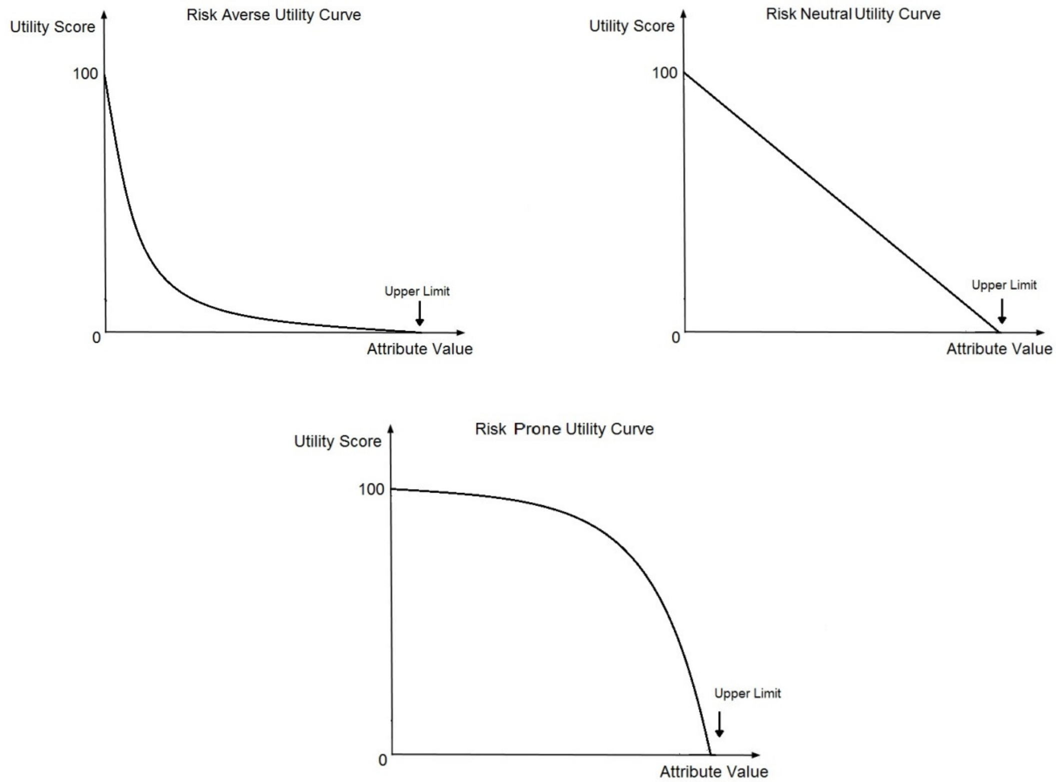

Figure 4.1: Examples of Three Utility Curves

Overall utility can then be found by aggregating the individual utilities using a weighting scheme. This value can then be compared to determine the reconfiguration scheme the designer would choose. For the race car example, this process is described by Equation 4.7.

$$OS = W_1 * US_P + W_2 * US_C + W_3 * US_{CM1} \ldots + W_7 * US_{CM5} \qquad (4.7)$$

In this equation, *OS* represents the overall utility score for a concept. The weighting scheme is represented by the variables W_1 through W_7, which are the individual weight values applied to the different attributes: lap time performance, consistency, and measures of system complexity. The weighting scheme is applied after the attributes have been converted into utility scores, represented by US_x. US_P applies to the performance of the vehicle as illustrated by mean lap time, US_C is the utility score for consistency, and US_{CM1} through US_{CM5} represent the utility scores of the complexity measures. The aggregation of these values creates a score that can be used to compare the reconfiguration schemes directly. For this study, the sum of all weights is always one. Therefore, because the utility curves return a maximum value of 100, the highest utility score for each design is a 100.

To accomplish the task of comparing designer preferences, an algorithm is used that generates a set of simulated utility curves and weighting schemes. The flowchart for the algorithm can be seen in Figure 4.2. The figure illustrates the general flow of the algorithm used to create each virtual designer. It starts with results from the multiobjective optimization completed in Chapter 3. The optimum designs are then assessed using utility curves for both the complexity and performance measures. The attribute value defines a corresponding utility score on the y-axis between 0 and 100. For each attribute, a higher utility score corresponds to a more desirable system.

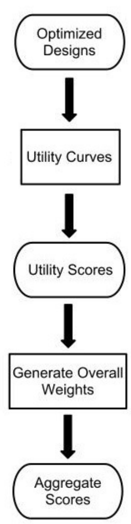

Figure 4.2: Virtual Designer Algorithm

In this work, utilities for $\overrightarrow{DVC}$ are considered at the discipline level. To obtain a single $\overrightarrow{DVC}$ score for each discipline, the values for all of the aerodynamic attributes must be combined into a single aerodynamics discipline score. This is only done for the aerodynamic discipline because the chassis and suspension disciplines each contain only one changeable variable. Figure 4.3 illustrates the sub-calculation needed to accommodate this calculation.

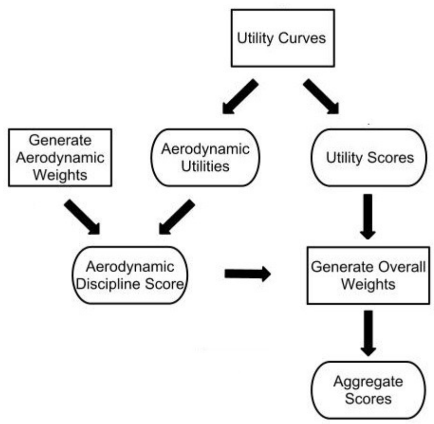

Figure 4.3: Calculating an Aerodynamic Discipline Utility Score

As shown in Equation 4.8, the aerodynamic scores are combined using a weighting scheme. Each utility score of $\overrightarrow{DVC}$ for the aerodynamic attributes is multiplied by a weight and the results are summed. Finally, this score is normalized on a scale of 0 to 100 so that it remains in the same magnitude as all of the other attributes.

$$AS = W_1 * US_{CV1} + W_2 * US_{CV2} + W_3 * US_{CV3} \ldots W_8 * US_{CV8} \qquad (4.8)$$

In this equation, *AS* represents the non-normalized aggregated aerodynamic maximum change variable score. Each variable W_1 through W_8 is a weight value. Additionally, the sum of these weights is one. Finally, the variables US_{CV1} through US_{CV8} represent the utility scores attached to the maximum change for each of the eight aerodynamic variables. These include three variables for the shape of each airfoil, front and back, and a variable for the angle of attack for each airfoil. The combined aerodynamic variable change score can then be placed with the scores from the other disciplines to represent the complexity measure maximum variable change.

The next step in the process is to use weights to aggregate the measures into a single total score for each reconfiguration scheme. A weight is applied to one of seven utility scores: lap time performance, lap time consistency, number of changes, change velocity, and the three scores for maximum variable change broken up by discipline. From this, an aggregate utility score is obtained, and it is assumed that a designer chooses the design that maximizes overall utility.

4.4 Discovering trends in system choice

To explore system selection, designer preferences were applied to the multi-objective optimization carried out in the previous chapter. The goal here is to highlight the differing choices that occur as a result of differing preferences. Two distinct virtual designers are compared, where each designer has their own set of utility curves and their own weighting scheme.

Designer A is interested in a fast car but also wants minimal changes to the design variables. Therefore, the designer A profile includes risk averse utility curves for the magnitude of change complexity measures and the number of changes measure. This indicates that low values will get a high utility score, and the utility score quickly decreases as the values increase. The change velocity complexity measure has a risk neutral utility curve because designer A is not interested in the frequency of change. The lap time performance preference is also represented by the risk averse utility curve where low values which equate to fast lap times are given high utility scores. Finally, the consistency measure, standard deviation of lap times, is represented with the risk prone utility curve. This is because designer A is willing to sacrifice consistency to meet the other demands.

Designer B is focused on a consistent race car even at the cost of a highly complex system. For designer B, the magnitude of change measures are represented with risk prone utility curves to indicate that there is leniency when it comes to large changes. This is also true of the number of changes complexity measure. The change velocity complexity measure has a risk neutral utility curve because the designer does not want to limit it too strictly, but is willing to include some higher values. The lap time performance is also represented by a risk neutral utility curve. This is because the designer is interested in retaining speed, but is willing to sacrifice some for consistency. Finally, consistency is associated with a risk averse utility curve because the designer would like the most consistent design possible.

4.5 Different utility curves but same weight scheme

Each designer profile is used to evaluate the designs from the multi-objective Pareto frontiers created in the previous chapter. This represents the process in which the designers investigate each design and rank their desirability in order to choose a final design. In this section, a common weighting scheme is used for both designers. The weighting scheme is randomly generated such that the sum of the values in each portion of the scheme is one. The weighting scheme is displayed in Tables 4.3 and 4.4.

Table 4.3: Aerodynamic Weights

Variable	Weight
Front AOA	0.576722
Rear AOA	0.233038
Front Camber	0.000855
Front Camber Distance	0.002482
Front Thickness	0.077427
Rear Camber	0.012037
Rear Camber Distance	0.014196
Rear Thickness	0.082981

Table 4.4: Overall Weights

Attribute	Weight
Fuel Position	0.005164
Roll Stiffness	0.001382
Aero. Discipline	0.00138
Number of Changes	0.02397
Change Velocity	0.001248
Lap Time	0.818149
Standard Deviation	0.148672

The weighting scheme is broken down into two parts. The first part is the aerodynamic specific weighting scheme. These weight values are used to make a cumulative aerodynamic discipline score with the aerodynamic magnitude of change measures. The second set of weight values represent the values used to create the overall aggregate score for each design. Each individual set of weights sums to a value of one. The results from the investigation are illustrated in the following figures.

4.5.1 Two state

The first figures, Figure 4.4a and 4.4b, show the results of the designers' preferences as applied to the two state multi-objective Pareto Frontier.

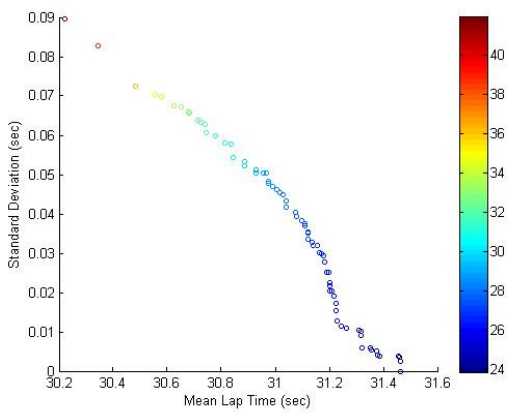

Figure 4.4a: Designer A 2 State

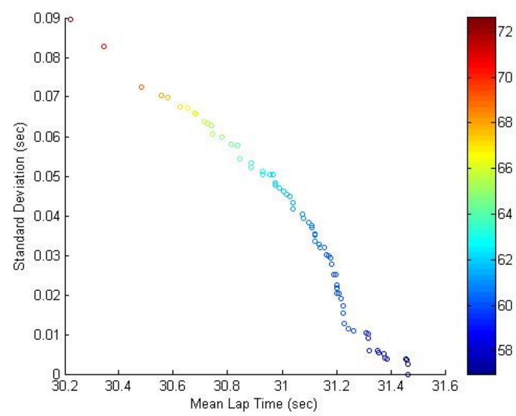

Figure 4.4b: Designer B 2 State

The results are displayed as a heat map with red representing high aggregate scores and blue representing low aggregate scores. The x-axis is the mean lap time for the race, and the y-axis is the standard deviation of the lap times. It can be seen from these figures that both designers have higher aggregate scores for the designs on the left of the frontier. These designs have higher standard deviations and lower mean lap times. One difference that can be noted from the figures is that designer B maintains a higher relative score as the mean lap times increases.

As stated earlier, the highest possible utility for any design is 100. Related to this, the graphs show that Designer B had higher utility scores overall than Designer A. This suggests that the designs on the two state frontier fit the ideal system preferences of Designer B better than Designer A. The data in Table 4.5 supports this assertion. It can be seen that the same design, which was scored highest by both designers, received a utility score approximately 30 points higher from Designer B than Designer A.

Tables 4.5a and 4.5b contain the design with the highest utility score, the lowest utility score and the median utility score for each designer. The first column is the aggregate utility score for each designer. The remainder of the tables contains the complexity measures, lap time performance and standard deviation. After the score column, the next ten columns contain the maximum change for each of the design variables.

Table 4.5a: Two State Preferred Designs (Utility Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	41.93	0.056	0.163	0.672	0.559	0.431	0.107	0.106
B	72.60	0.056	0.163	0.672	0.559	0.431	0.107	0.106
				Median Score				
A	27.15	0.064	0.153	0.638	0.558	0.435	0.129	0.081
B	60.66	0.035	0.186	0.638	0.559	0.432	0.129	0.068
				Low Score				
A	23.89	0.052	0.233	0.598	0.544	0.318	0.162	0.068
B	56.94	0.055	0.216	0.613	0.545	0.325	0.166	0.013

Table 4.5b Two State Preferred Designs (Utility Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
				High Score			
A	0.641	0.016	0.056	30	0.993	30.22	0.090
B	0.641	0.016	0.056	30	0.993	30.22	0.090
				Median Score			
A	0.648	0.000	0.073	30	0.964	31.11	0.037
B	0.644	0.006	0.068	30	0.964	31.11	0.037
				Low Score			
A	0.636	0.018	0.071	30	0.953	31.46	0.000
B	0.631	0.018	0.054	30	0.954	31.46	0.004

The most favored design for each designer is the same. A similar result is seen for the median score design. Each design has the same mean lap time and standard deviation, and many of the design variables have equivalent ranges of change. Finally, the least preferred design for each designer matches the trend of the previous two designs. Each has the same mean lap time as well as very similar maximum design variable changes.

Some of the trends seen in the figures and the table can be explained by taking a closer look at the utility values and aggregate score of each design. For example, the median design for both designers has a lap time of 31.11 seconds and a standard deviation of 0.037 seconds. Designer A has strict requirements for lap time because a fast design is desired. Therefore, Designer A gives the lap time a score of 14.35 utils. On the other hand, designer B is willing to accept a wide array of lap times, and this designer gives the same lap time a score of 57.8 utils. Additionally, the weighting scheme for this situation weighs lap time the highest among the attributes, at 0.818. For both designers then, the lap time score plays a large role in determining the final aggregate score. This is manifested in the fact that Designer B maintains a higher relative score moving down the Pareto frontier. Put another way, Designer A has very high standards for a fast design, and therefore favors only the fastest designs on the Pareto frontier.

Also, the data shows that Designer B maintained high scores across the Pareto frontier. One reason for this is Designer B's willingness to accept a variety of mean lap times. The risk prone utility curve assigned to lap time for Designer B provides high scores for a larger range of lap times than the risk averse curve assigned to Designer A. Since the lap time is the highest weight in this weighting scheme, the combination of high utility scores and high weight give Designer B higher overall scores.

4.5.2 Five state

Figures 4.5a and 4.5b represent the results for the five state Pareto frontier. These figures are presented in a similar form to the two state frontier - red represents higher scores and blue represents lower scores. It can be seen from these figures that there is a more dramatic difference between the two designers' design choices. For designer A, the lowest scores occur for designs at the left of the frontier, and the scores increase as they move right along the frontier. Conversely, for designer B, the highest scores are located at the left of the frontier, and become progressively lower moving right along the frontier. This is an example of how the different designer preferences target different sections of the Pareto frontier.

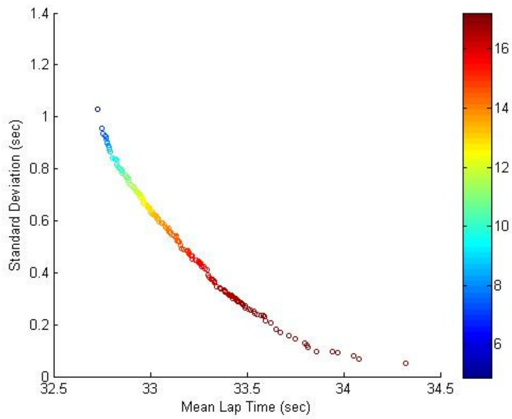

Figure 4.5a: Designer A 5 State

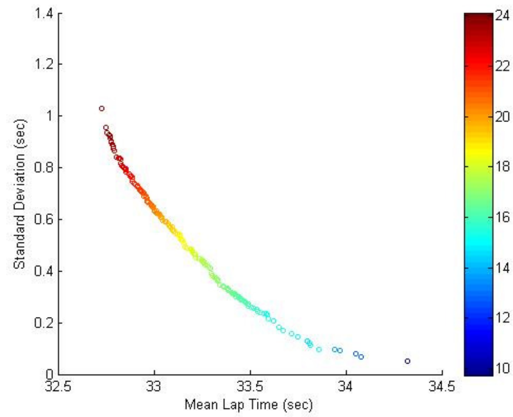

Figure 4.5b Designer B 5 State

Tables 4.6a and 4.6b compare three designs from each designer. Each design is represented by the complexity measures, mean lap time and standard deviation.

Table 4.6a Five State Preferred Designs (Utility Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	17.22	0.107	0.164	0.152	0.189	0.137	0.327	0.143
B	24.12	0.077	0.143	0.096	0.100	0.109	0.260	0.378
				Median Score				
A	14.43	0.087	0.122	0.163	0.107	0.074	0.252	0.098
B	18.71	0.075	0.140	0.165	0.109	0.076	0.239	0.081
				Low Score				
A	4.87	0.077	0.143	0.096	0.100	0.109	0.260	0.378
B	9.72	0.224	0.362	0.133	0.158	0.130	0.304	0.219

Table 4.6b Five State Preferred Designs (Utility Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
			High Score				
A	0.158	0.128	0.088	5	0.148	33.72	0.156
B	0.269	0.104	0.147	5	0.153	32.73	1.031
			Median Score				
A	0.145	0.045	0.109	5	0.151	33.13	0.539
B	0.141	0.058	0.042	5	0.151	33.13	0.537
			Low Score				
A	0.269	0.104	0.147	5	0.153	32.73	1.031
B	0.133	0.231	0.364	5	0.146	34.32	0.050

When the designs are examined, it can be seen that while different utility curve profiles change the favored designs for each designer, they do not change in an expected way. Designer A's most favored design has a worse lap time than Designer B, and seven of the ten maximum change measures have a greater magnitude than those from Designer B. Both of these facts are contrary to the predicted outcome based on the preferences of each designer. The median designs for each designer were very similar with identical mean lap times, and similar complexity measure values. Finally, the lowest scored design for each designer present a picture very much like the most favored design. The results for each designer are contrary to the predicted outcome based on the designer preferences.

The reasons for this can be seen when examining the aggregate utility equation and its values. First, for this weighting scheme, the highest weight goes to performance at 0.818. This means that after the utility scores are calculated, both designers value fast designs. Second, the utility curve associated with Designer A's performance preference is the risk averse or exponential curve. This means that designer A is focused on only the fastest designs, and any slower than the fastest quickly lose value. For example, the fastest time on the chart, 32.73 seconds returns a utility score of just 3.23 out of 100. Combined with the 0.818 weight, the contribution of performance to the aggregate score is 2.64.

The second highest weight in this scheme is the standard deviation at 0.149. Designer A has associated the risk prone curve or inverse quadratic to this attribute, which implies that even a sharp decrease in the value will maintain a high utility score. The combination of high utility scores and the second highest weight in the scheme means that the standard deviation plays a significant role in the determination of the designs utility. For example, the standard deviation associated with the fastest car is 1.031 seconds which equates to a utility of 3.586, and a contribution to aggregate utility of 0.533. On the other hand, the standard deviation of Designer A's highest scored design is 0.156. This gives a utility of 97.79 and contribution to aggregate utility of 14.54.

The contribution to the aggregate utility of the mean lap time and standard deviation is an order of magnitude different. This comparison shows how the standard deviation contributes more to Designer A's utility causing the focus on the lower end of the Pareto frontier. In contrast, Designer B assigns a risk prone utility curve to the mean lap time. This means that a wide range of race times maintain a high utility score. When this fact is combined with the high weight placed on the performance in this weighting scheme, it is easy to see how Designer B focuses on the faster section of the frontier. For example, the fastest time, 32.73 seconds, received a 3.23 utility score in the eyes of Designer A. Designer B however, gave this same value a utility score of 25.4 which led to targeting the fast section of the frontier.

4.5.3 Ten state

The next set of figures, Figures 4.6a and 4.6b, show the results for the ten state frontier. The figures show that both designers place the highest utility scores with designs on the left side of the frontier, and the lowest scores are on the right side of the frontier. For designer A, the scores decrease moving from left to right and shift through the spectrum from red to light blue by the middle of the curve. In the frontier for designer B, the designs maintain a higher relative score further along the curve. The three designs for each designer are displayed in Tables 4.7a and 4.7b.

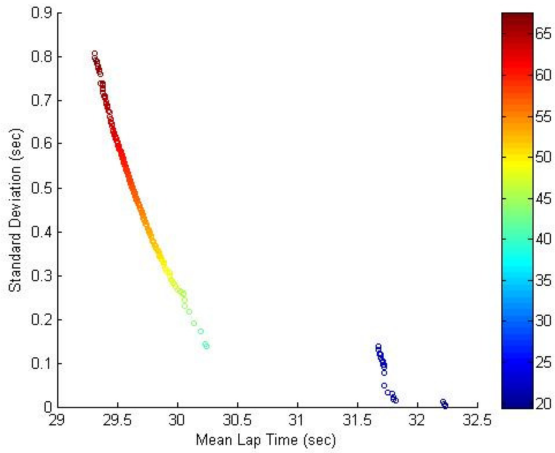

Figure 4.6a: Designer A 10 State

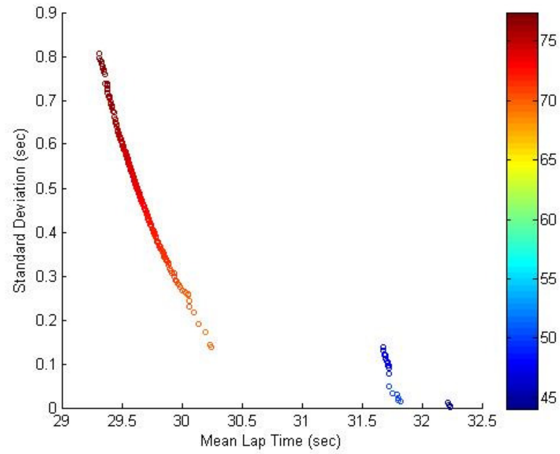

Figure 4.6b: Designer B 10 State

Table 4.7a Ten State Preferred Designs (Utility Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	67.66	0.843	0.607	0.892	0.652	0.705	0.689	0.776
B	77.42	0.853	0.592	0.892	0.652	0.705	0.532	0.780
				Median Score				
A	56.85	0.845	0.612	0.892	0.652	0.700	0.785	0.807
B	73.28	0.845	0.612	0.892	0.652	0.700	0.785	0.807
				Low Score				
A	19.45	0.189	0.084	0.208	0.415	0.221	0.159	0.142
B	44.06	0.172	0.086	0.206	0.437	0.221	0.152	0.143

Table 4.7b Ten State Preferred Designs (Utility Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
			High Score				
A	0.560	0.560	0.807	30	1.023	29.31	0.796
B	0.560	0.542	0.746	30	1.023	29.31	0.808
			Median Score				
A	0.556	0.584	0.774	30	1.012	29.65	0.484
B	0.556	0.584	0.774	30	1.012	29.65	0.484
			Low Score				
A	0.529	0.350	0.085	30	0.931	32.23	0.002
B	0.529	0.347	0.112	30	0.931	32.22	0.011

The design with the highest utility score for each designer is almost identical. They have the same mean lap time and there is very little difference in any of the complexity measures. Both designers chose the same median design. Finally, for the design with the lowest score the complexity measure values are very close, and the mean lap time is only one thousandth of a second different.

Once again, the influence of the different mean lap time utility curves can be seen in the ten state frontier. This time, Designer A found designs that meet the strict expectations conveyed by the risk averse utility curve. This can be seen in the higher overall scores, and the larger patch of red on the Pareto frontier. For example, the design with the highest score had a lap time of 29.31 seconds which corresponds to a utility score of 75.18 for Designer A. When this is compared with the scores from the fastest designs on the other frontiers, 14.75 and 3.23, it is clear why the scores went up for Designer A on the ten state frontier.

Additionally, Designer B maintains higher relative scores further along the frontier because of the risk prone utility curve associated with mean lap time. This provides higher scores for slower designs than those found with Designer A. Also, with the high weight of mean lap time with this weighting scheme, the lap time was able to dominate the aggregate utility score for Designer B.

4.5.4 Combined analysis

The final figures, Figures 4.7a and 4.7b, show the three frontiers on the same graph. For both designers, the far left side of the graph contains the points with the highest utility scores. These points are generally contained in the ten state and two state Pareto Frontiers, and they are characterized by having the lowest mean lap time. The main difference between the two designers is that one favors slightly slower designs that are more consistent. Designer A focuses on the fastest designs with little regard for consistency. Conversely, Designer B extends the focus further towards the bottom of the graph, to the more consistent designs.

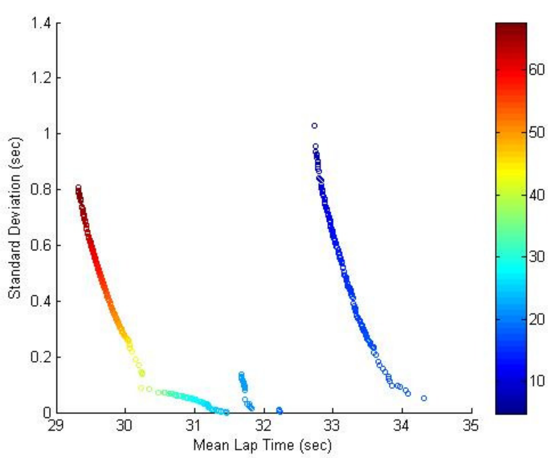

Figure 4.7a: Designer A All States

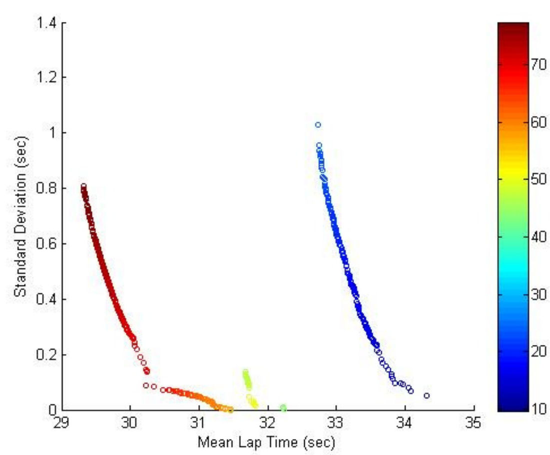

Figure 4.7b: Designer B All States

Tables 4.8a and 4.8b contain the designs with the highest scores from each frontier. This gives an idea about how the frontiers fit together in the eyes of each designer. The format for the table is the same as the previous tables.

Table 4.8a All Frontiers Preferred Designs (Utility Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				Two State				
A	41.93	0.056	0.163	0.672	0.559	0.431	0.107	0.106
B	72.60	0.056	0.163	0.672	0.559	0.431	0.107	0.106
				Five State				
A	17.22	0.107	0.164	0.152	0.189	0.137	0.327	0.143
B	24.12	0.077	0.143	0.096	0.100	0.109	0.260	0.378
				Ten State				
A	67.66	0.843	0.607	0.892	0.652	0.705	0.689	0.776
B	77.42	0.853	0.592	0.892	0.652	0.705	0.532	0.780

Table 4.8b All Frontiers Preferred Designs (Utility Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
				Two State			
A	0.641	0.016	0.056	30	0.993	30.22	0.090
B	0.641	0.016	0.056	30	0.993	30.22	0.090
				Five State			
A	0.158	0.128	0.088	5	0.148	33.72	0.156
B	0.269	0.104	0.147	5	0.153	32.73	1.031
				Ten State			
A	0.560	0.560	0.807	30	1.023	29.31	0.796
B	0.560	0.542	0.746	30	1.023	29.31	0.808

From the figures it can be seen that Designer A focuses mostly on the ten state frontier. The designs in the tables reinforce this notion with the ten state designs showing the lowest mean lap time. Both figures and tables show that Designer A's focus on minimizing change is outweighed by the desire for speed when it is applied solely through the utility curve profile.

For Designer B, the most favored designs include the two state frontier and the ten state frontier. The data in the table shows that the two state frontier has points with the lowest standard deviation, as well as comparably fast times. The ten state frontier also has fast times, and equivalent standard deviations to the five state frontier.

The same trends that were seen in the individual frontiers can be seen in this combined analysis. Designer A favors only the fastest designs which can get a high utility score with the risk averse curve. Therefore, the highest scores are seen on the ten state frontier, and little attention is paid to the other schemes. Additionally, Designer B targets the same region, but the two state frontier is also scored highly. This is due to the high weight associated with mean lap time, and the lenient scoring associated with Designer B's lap time utility curve.

4.6 Same utility curves and different weight scheme

In this section, the designers are again characterized by their interests in the ideal race car. This time the interests are represented by a unique weighting scheme for each designer. These weighting schemes are displayed in Table 4.9. The first three rows in the table contain the maximum change complexity measure. It is broken down by the three disciplines: chassis - fuel position, suspension - roll stiffness, and aerodynamics. Designer A is focused on designs that perform well with low mean lap times, but there is also an emphasis on reducing the amount of change in the design space. Consistency is given very little importance in order to obtain the desired results. Conversely, designer B is willing to choose a design that changes significantly to achieve a consistent design.

Table 4.9: Unique Weighting Schemes for Two Designers

Variable	Weighting Scheme Designer A	Weighting Scheme Designer B
Fuel Position	0.15	0.08
Roll Stiffness	0.15	0.08
Aerodynamic Discipline	0.15	0.08
Number of Changes	0.12	0.08
Change Velocity	0.10	0.10
Lap Time	0.30	0.16
Standard Deviation	0.03	0.42

Each design is evaluated with a common utility curve profile, with a different weighting scheme representing each of the designers. This utility curve profile is generated by assigning a random utility curve, 1, 2, or 3, to each attribute affecting designer preference. The utility curve profile is shown in Table 4.10.

Table 4.10 Common Utility Curve Profile

Variable	Utility Curve
Fuel Position	3 (Risk Prone)
Roll Stiffness	3 (Risk Prone)
Front AOA	1 (Risk Averse)
Rear AOA	3 (Risk Prone)
Front Camber	2 (Risk Neutral)
Front Camber Dist.	1 (Risk Averse)
Front Max Thickness	1 (Risk Averse)
Rear Camber	2 (Risk Neutral)
Rear Camber Dist.	3 (Risk Prone)
Rear Max Thickness	3 (Risk Prone)
Number of Changes	1 (Risk Averse)
Change Velocity	3 (Risk Prone)
Lap Time	3 (Risk Prone)
Standard Deviation	2 (Risk Neutral)

The table shows the common utility curve profile that is used to evaluate all of the designs. The first curve is a risk averse utility curve, the second is a risk neutral utility curve, and the third is a risk prone utility curve. Once again, the equations for these curves can be found in Equations 4.4 through 4.6. The generation of utility scores from the equations is the same as described in the previous section. Each curve is unique based on the bounds of the attribute in question, but all curves of the same type retain the same shape and trend in utility score. Examples of the curves can be seen in Figure 4.1.

4.6.1 Two state

Figures 4.8a and 4.8b show that each designer targets a different portion of the frontier. Designer A focuses on the far left side of the graph, which is populated by designs with low mean lap times. Conversely, Designer B focuses on the lower right portion of the graph where lap times are higher, but the designs are more consistent.

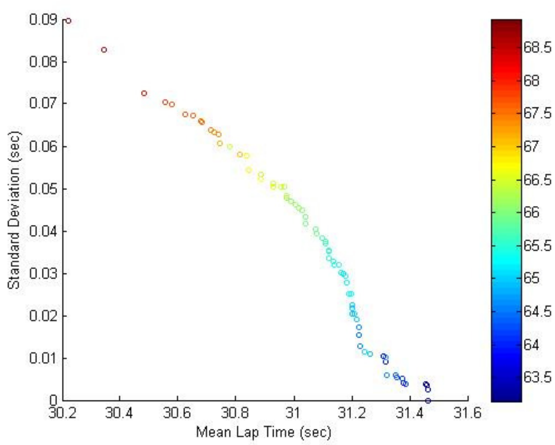

Figure 4.8a: Constant Utility Designer A 2 State

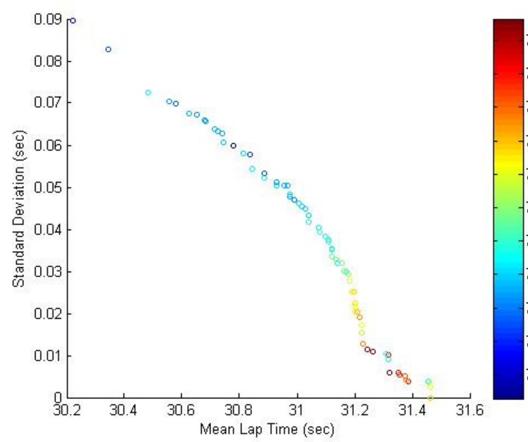

Figure 4.8b: Constant Utility Designer B 2 State

Tables 4.11a and 4.11b show a comparison of three designs for each designer. As in the previous section, the three designs are those with the highest, lowest and median scores for each designer. They are represented by the complexity measures, mean lap time, and standard deviation.

Table 4.11a Two State Preferred Designs (Weight Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	68.93	0.056	0.163	0.672	0.559	0.431	0.107	0.106
B	78.26	0.053	0.167	0.638	0.556	0.430	0.130	0.047
				Median Score				
A	65.59	0.035	0.186	0.638	0.559	0.432	0.129	0.068
B	77.58	0.013	0.174	0.640	0.558	0.433	0.113	0.076
				Low Score				
A	63.14	0.053	0.281	0.598	0.542	0.325	0.179	0.079
B	77.13	0.221	0.148	0.639	0.550	0.430	0.122	0.075

Table 4.11b Two State Preferred Designs (Weight Change)

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
				High Score			
A	0.641	0.016	0.056	30	0.993	30.22	0.090
B	0.639	0.005	0.046	30	0.960	31.24	0.012
				Median Score			
A	0.644	0.006	0.068	30	0.964	31.11	0.037
B	0.644	0.005	0.058	30	0.965	31.10	0.038
				Low Score			
A	0.632	0.010	0.060	30	0.954	31.46	0.004
B	0.644	0.014	0.060	30	0.975	30.78	0.060

The designs with the highest score highlight one of the differences between the two designers. The complexity measures for these two designs are very similar. The main differences between the two designs can be seen in the mean lap time and standard deviation. Designer A prefers a design with a lower lap time and higher standard deviation when compared to the design preferred by Designer B. The median designs for both designers are nearly identical with mean lap times and standard deviations that are only one hundredth and one thousandth of a second different, respectively. Finally, the design with the lowest utility score for Designer A has a high mean lap time, which is in contrast to Designer B's design with a low mean lap time. This is consistent with the fact that Designer A prefers a fast design and Designer B prefers a consistent design.

4.6.2 Five state

Figure 4.9a and 4.9b, illustrate the results for each designer as applied to the five state frontier. For Designer A, the highest scores are associated with the top left side of the frontier where the lap times are lowest. Alternately, Designer B focused on the bottom right of the frontier where the standard deviations are lowest. This is consistent with both the two state results and the weighting scheme where Designer A values speed and Designer B values consistency. Three designs for each designer are displayed in Tables 4.12a and 4.12b.

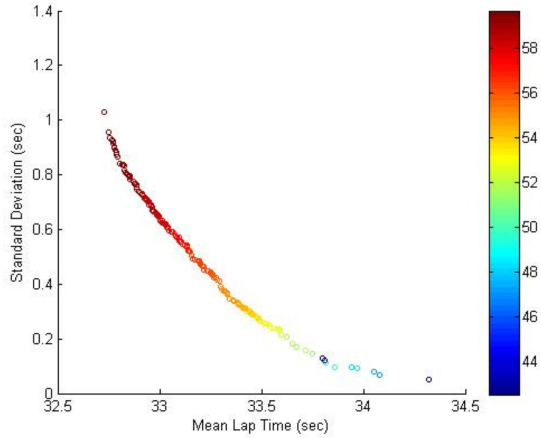

Figure 4.9a: Constant Utility Designer A 5 State

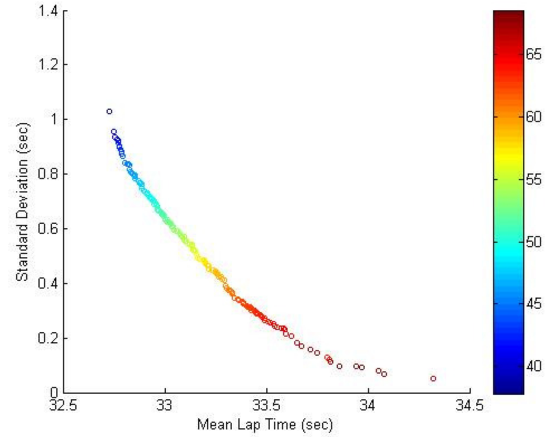

Figure 4.9b: Constant Utility Designer B 5 State

Table 4.12a Five State Preferred Designs (Weight Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	59.67	0.052	0.108	0.186	0.111	0.081	0.219	0.060
B	68.54	0.184	0.209	0.099	0.123	0.122	0.183	0.147
				Median Score				
A	57.09	0.075	0.140	0.165	0.109	0.076	0.239	0.081
B	55.50	0.087	0.122	0.163	0.107	0.074	0.252	0.098
				Low Score				
A	42.55	0.224	0.362	0.133	0.158	0.130	0.304	0.219
B	37.82	0.077	0.143	0.096	0.100	0.109	0.260	0.378

Table 4.12b Five State Preferred Designs (Weight Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
			High Score				
A	0.153	0.081	0.435	5	0.153	32.75	0.936
B	0.146	0.202	0.218	5	0.148	33.86	0.098
			Median Score				
A	0.141	0.058	0.042	5	0.151	33.13	0.537
B	0.145	0.045	0.109	5	0.151	33.13	0.539
			Low Score				
A	0.133	0.231	0.364	5	0.146	34.32	0.050
B	0.269	0.104	0.147	5	0.153	32.73	1.031

Designer A's most preferred design is one with a comparably low mean lap time and high standard deviation. Conversely, designer B focuses on the design with a low standard deviation and high lap time. Seven of the ten maximum change measures are lower for Designer A than for Designer B. This is consistent with the fact that Designer A is interested in fast systems with minimal change.

4.6.3 Ten state

Figures 4.10a and 4.10b, show the results for the ten state frontier. For previous frontiers designer A valued the left side of the frontiers and designer B sought designs on the bottom right. These figures show that both designers focus on the bottom right of the frontiers. The difference between the two is that Designer B values designs farther up the frontier than Designer A. One explanation for this is that, in addition to valuing lap time, Designer A is interested in designs that change very little. Due to the fact that the ten state designs are allowed to change more than the other two reconfiguration schemes, the fastest designs were not as highly desired because they include a significant amount of change. Designer B is focused most highly on consistency which leads to the bottom right of the frontier being the targeted section.

A comparison of three designs from each designer is displayed in Tables 4.13a and 4.13b. These are the designs with the highest, lowest and median utility scores. The tables display the complexity measures, mean lap time, and standard deviation for each design.

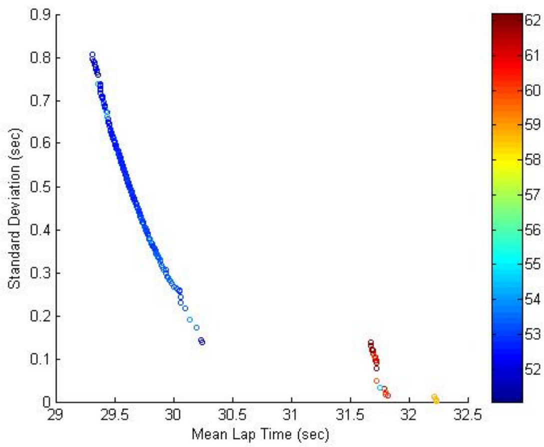

Figure 4.10a: Constant Utility Designer A 10 State

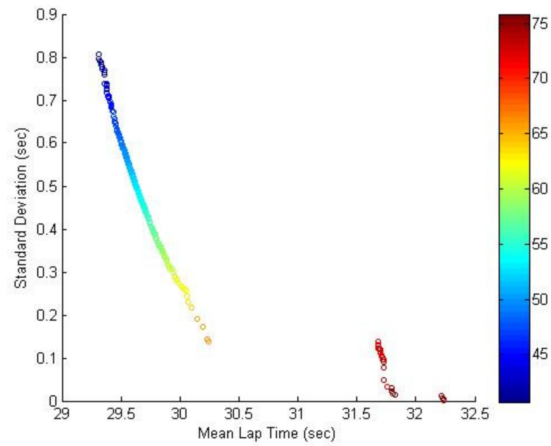

Figure 4.10b: Constant Utility Designer B 10 State

Table 4.13a Ten State Preferred Designs (Weight Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	62.18	0.132	0.087	0.271	0.634	0.408	0.158	0.219
B	75.83	0.286	0.089	0.330	0.635	0.215	0.158	0.153
				Median Score				
A	52.81	0.843	0.614	0.892	0.652	0.697	0.755	0.831
B	53.38	0.835	0.602	0.892	0.652	0.705	0.765	0.847
				Low Score				
A	51.03	0.892	0.609	0.892	0.652	0.706	0.663	0.839
B	40.69	0.853	0.592	0.892	0.652	0.705	0.532	0.780

Table 4.13b Ten State Preferred Designs (Weight Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
				High Score			
A	0.523	0.563	0.309	30	0.947	31.69	0.120
B	0.525	0.559	0.140	30	0.943	31.82	0.015
				Median Score			
A	0.555	0.570	0.835	30	1.006	29.81	0.370
B	0.556	0.575	0.794	30	1.012	29.64	0.490
				Low Score			
A	0.562	0.777	0.798	30	1.022	29.36	0.759
B	0.560	0.542	0.746	30	1.023	29.31	0.808

The design most favored by Designer B has a higher lap time and lower standard deviation than that of designer A. The mean lap time for Designer A though is not the fastest on the frontier, and it is only slightly lower, 0.13 seconds, than that for design B. Additionally, only five of the ten maximum change measures are lower for design A. The design with the lowest score shows the complexity measure preferences of Designer A more than the other designs. For this design, eight of the ten maximum variable change values are higher for design A. This is in line with Designer A disliking designs that change more, and it appears that the desire to avoid change manifests more in generating low utility scores for design with significant change than in generating high scores for simple designs.

This trend is born out in the aggregate utility equation as well. For Designer A's most preferred design, the ranges of change attributes contribute 14.74, 14.89, and 6.38 utils respectively out of a total score of 62.18 utils. On the other hand, for the least preferred design, these contributions drop to 3.06, 9.44, and 2.80 utils out of a score of 51.03 utils. This holds true with the idea that for Designer A the complexity measures contribute to a large portion of the total aggregate score.

4.6.4 Combined analysis

Finally, Figures 4.11a and 4.11b display the results for all three frontiers concurrently. Designer A focuses on the two state frontier, and the fastest designs from the five state frontier. This is consistent with the fact that designer A is interested in speed and minimal changes to the design. The fastest ten state designs changes too much to be valued by designer A, and the slowest five state designs do not perform well enough. Designer B also focuses on the two state frontier. The difference is that designer B also desires other designs at the bottom of the graph, representing very consistent designs. As the frontiers are traced further up towards less consistent designs, they receive lower utility scores from designer B.

Tables 4.14a and 4.14b show that designer A favors designs on the two state frontier and with slightly lower scores, the five state frontier. This is the case even though the ten state frontier has lower mean lap times. The reasoning for this is clearly illustrated when the five state point and the ten state point are examined. The five state design has a higher lap time, but nine of the twelve total complexity measures are lower than the ten state design. This is an example of a situation where, in designer A's estimation, the simple nature of the five state design outweighs the benefits gained with the faster highly complex design.

For designer B, the focus was on consistency, and therefore, designs with low standard deviations are highly desired. This can be seen when comparing the five and ten state designs. The ten state point, which comes from the lower left of the frontier, has a lower standard deviation than the five state point. This demonstrates why the lower section of the frontiers is most favored by designer B.

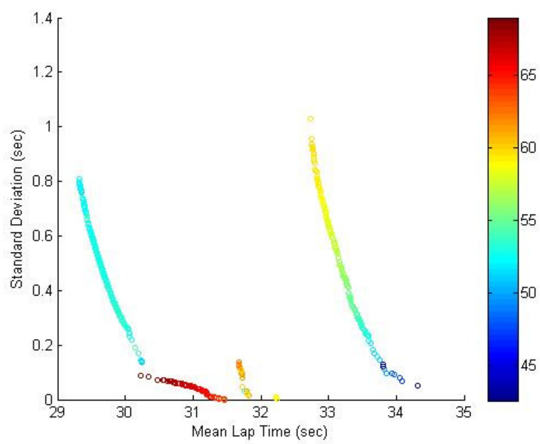

Figure 4.11a: Constant Utility All States Designer A

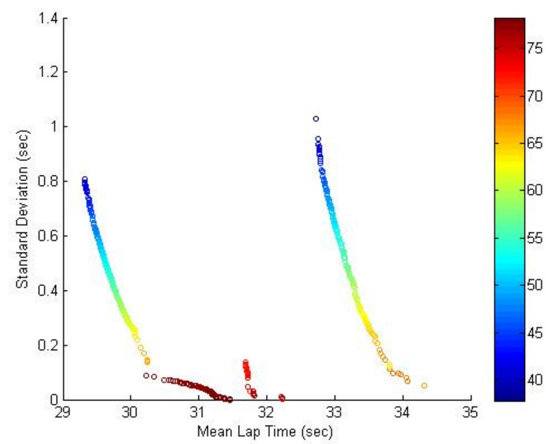

Figure 4.11b: Constant Utility All States Designer B

Table 4.14a All Frontiers Preferred Designs (Weight Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				Two State				
A	68.93	0.056	0.163	0.672	0.559	0.431	0.107	0.106
B	78.26	0.053	0.167	0.638	0.556	0.430	0.130	0.047
				Five State				
A	59.67	0.052	0.108	0.186	0.111	0.081	0.219	0.060
B	68.54	0.184	0.209	0.099	0.123	0.122	0.183	0.147
				Ten State				
A	62.18	0.132	0.087	0.271	0.634	0.408	0.158	0.219
B	75.83	0.286	0.089	0.330	0.635	0.215	0.158	0.153

Table 4.14b All Frontiers Preferred Designs (Weight Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
				Two State			
A	0.641	0.016	0.056	30	0.993	30.22	0.090
B	0.639	0.005	0.046	30	0.960	31.24	0.012
				Five State			
A	0.153	0.081	0.435	5	0.153	32.75	0.936
B	0.146	0.202	0.218	5	0.148	33.86	0.098
				Ten State			
A	0.523	0.563	0.309	30	0.947	31.69	0.120
B	0.525	0.559	0.140	30	0.943	31.82	0.015

The results displayed in these sections demonstrate that designer preferences in the form of weighting schemes have a noticeable effect on what designs might be selected. In addition, it can be seen that the weighting schemes provide a more predictable and consistent difference between the two designers than the differing utility curve profiles.

4.7 Different utility curves and different weighting scheme

In the previous sections, either the utility curve profile or the weighting scheme was allowed to change between designers while the other was held constant. The first section varied the utility curve profile, and the second varied the weighting scheme. In this section, both of these are varied, and a unique utility curve profile and weighting scheme is used for each designer. As in the previous sections, each designer's preferences are captured by the utility curve profile and weighting scheme. Designer A is interested in a fast car, but at the same time a design with limited changes is also desirable. Designer B is willing to allow changes in order to obtain a consistently performing vehicle. This section combines the unique utility curve profiles and weighting schemes that were introduced in previous sections. The utility curve profiles are described in section 4.4, and the weighting schemes can be seen in Table 4.4.

In the same fashion as the previous sections, each designer preference profile is applied to the three Pareto Frontiers created in Chapter 3. The same algorithm is used to combine the utility scores for each attribute with the weighting scheme to create an aggregate utility score for each design along the frontier. The difference for this section is that the two designers do not share either utility profiles or weighting schemes. The results from the investigation are displayed in the following figures and tables.

4.7.1 Two state

Figures 4.12a and 4.12b display the results for the two state Pareto Frontier. On the left, Figure 4.12a, are the results for Designer A, and on the right, Figure 4.12b, are the results for Designer B. Additionally, three designs from each designer are displayed in Tables 4.15a and 4.15b. From the figures, it can be seen that Designer A favors designs at the upper left end of the Pareto Frontier. These are designs that have low mean lap times but high standard deviations. This focus is consistent with the idea that Designer A favors fast vehicles that change very little. Designer B on the other hand favors designs found on the lower right of the frontier which are characterized by high mean lap times and low standard deviation. Once again, this is consistent with the idea that Designer B's main focus is on a consistent vehicle.

One interesting thing to note though is that there are a few blue, low score, points on the upper left of the Pareto frontier. The points are interspersed with the red, high score, points in that area. This signifies that something about those specific systems does not match the preferences of Designer A. The aggregate utility equation can be used to analyze these points in more detail. When the anomalous points are compared to their neighboring points, it can be seen that the biggest difference is the utility of the maximum change measure for the fuel position. For the lower score points, the utility of this measure was between 20 to 30 points lower than the other designs in the same area. This indicates that for these points the fuel position had a higher range of change when compared to the neighboring points. When this is combined with the weighting scheme it translates to a 4 to 6 point drop in overall utility. The scores for this frontier max at 37.08 so this drop is between 10 and 16 percent of the total score. This is an example of Designer A's preference for minimal change affecting the utility of some designs.

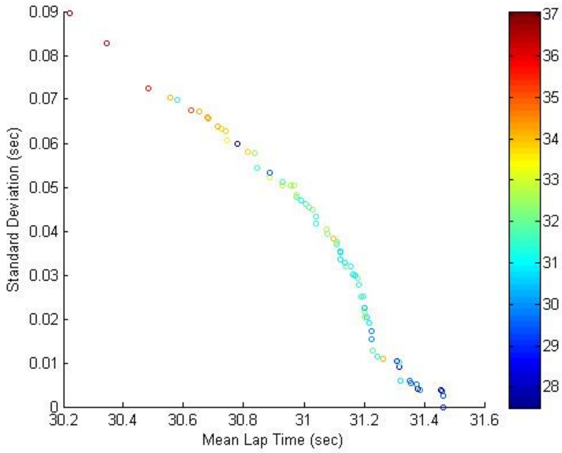

Figure 4.12a: Both Changed Designer A 2 State

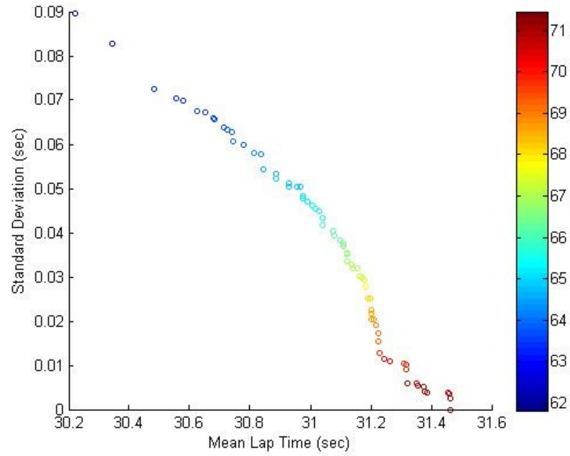

Figure 4.12b: Both Changed Designer B 2 State

Table 4.15a Two State Preferred Designs (Both Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	37.08	0.056	0.163	0.672	0.559	0.431	0.107	0.106
B	71.46	0.052	0.233	0.598	0.544	0.318	0.162	0.068
				Median Score				
A	31.70	0.064	0.153	0.638	0.558	0.435	0.129	0.081
B	66.49	0.064	0.153	0.638	0.558	0.435	0.129	0.081
				Low Score				
A	27.49	0.088	0.227	0.613	0.552	0.345	0.157	0.162
B	61.82	0.056	0.163	0.672	0.559	0.431	0.107	0.106

Table 4.15b Two State Preferred Designs (Both Change) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
			High Score				
A	0.641	0.016	0.056	30	0.993	30.22	0.090
B	0.636	0.018	0.071	30	0.953	31.46	0.000
			Median Score				
A	0.648	0.000	0.073	30	0.964	31.11	0.037
B	0.648	0.000	0.073	30	0.964	31.11	0.037
			Low Score				
A	0.625	0.036	0.022	30	0.956	31.38	0.004
B	0.641	0.016	0.056	30	0.993	30.22	0.090

The data in Tables 4.15a and 4.15b follow the trend illustrated in the figures. The most highly preferred vehicle for Designer A has a lower lap time and higher standard deviation than that of Designer B. Additionally, the maximum change complexity measure is split with six of the ten values being higher in design A. This follows with the weighting scheme where mean lap time is given a higher weight than variable change. The designers chose the same median design. Finally, the designs with the lowest score follow the trend with Designer A's least preferred design having a higher mean lap time and lower standard deviation than Designer B's.

4.7.2 Five state

Figures 4.13a and 4.13b display the five state Pareto Frontier results for Designers A and B respectively, and Tables 4.16a and 4.16b show three designs from each designer. From the figures, Designer A favors the upper left of the Pareto Frontier which contains designs with low mean lap time. Designer B focuses most on the lower right of the frontier where low standard deviations can be found. Both of these are consistent with the stated preferences of each designer.

Additionally, it can be seen that Designer A maintains high relative utility score throughout most of the frontier. This is in stark contrast to Designer B where only the very lowest section of the frontier received high scores. For Designer A, this is a manifestation of the desire to maintain low complexity. The designs on this frontier have relatively slow times when compared with the other frontiers. This translates to very low scores for Designer A. Therefore, the complexity measures are allowed to dominate the aggregate utility equation, and influence the overall utility of the designs. Since these designs maintain low complexity as measured by the maximum variable change, number of changes, and change velocity measures, the high utility scores extend throughout a majority of the frontier.

The behavior of Designer B is due to the premium placed on low standard deviation. The utility curve associated with this measure is the risk averse curve, and the weight is the highest in this weighting scheme at 0.42. Therefore, very low standard deviations provide high utility scores and dominate the aggregate utility equation, but the high standard deviations are associated with very low scores. For example, the highest rated design on this frontier has a standard deviation of 0.05 seconds which translates to a utility of 80.33. In contrast, the highest standard deviation 1.031 seconds has a utility score of 1.09. This is nearly a 79 point difference in utility scores, and when combined with the weight these become 33.74 and 0.459 respectively. It can be seen then that the low standard deviations are very highly favored, but slight increases quickly lower the overall utility of the design.

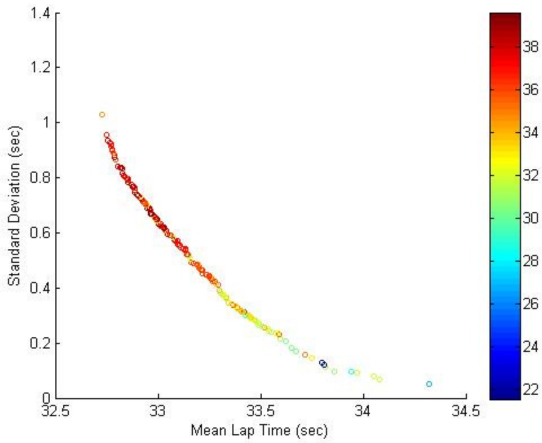

Figure 4.13a: Both Changed Designer A 5 State

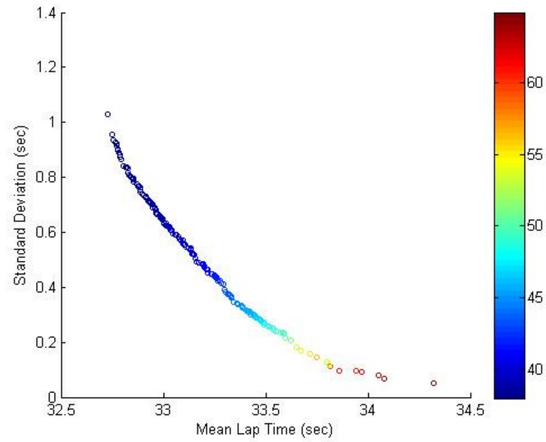

Figure 4.13b: Both Changed Designer B 5 State

Table 4.16a Five State Preferred Designs (Both Change)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
	High Score							
A	39.60	0.035	0.117	0.179	0.104	0.074	0.232	0.070
B	64.92	0.224	0.362	0.133	0.158	0.130	0.304	0.219
	Median Score							
A	35.83	0.065	0.189	0.489	0.068	0.105	0.251	0.186
B	40.22	0.079	0.117	0.162	0.119	0.086	0.249	0.094
	Low Score							
A	21.57	0.630	0.416	0.242	0.280	0.233	0.152	0.379
B	37.98	0.077	0.143	0.096	0.100	0.109	0.260	0.378

Table 4.16b Five State Preferred Designs (Both Changes) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
			High Score				
A	0.140	0.073	0.165	5	0.152	32.96	0.684
B	0.133	0.231	0.364	5	0.146	34.32	0.050
			Median Score				
A	0.108	0.233	0.161	5	0.151	33.20	0.473
B	0.153	0.041	0.062	5	0.151	33.11	0.545
			Low Score				
A	0.478	0.221	0.240	5	0.148	33.80	0.130
B	0.269	0.104	0.147	5	0.153	32.73	1.031

In Tables 4.16a and 4.16b designer A's most favored design has a low lap time and high standard deviation when compared with designer B. Designer B instead focused on the design with the low standard deviation. The median designs for each are very similar with the lap time and standard deviation each less than a tenth of a second different. Finally, the lowest scoring design reinforces the fact that each designer is consistent with the preference description. The design that A least preferred is characterized by a slow mean lap time with a low standard deviation, and designer B’s choice is a fast design with a high standard deviation. One thing to note is that designer B's least favored design is actually slightly faster than designer A's preferred design. The draw to the slightly slower design can be seen by looking at the complexity measures where seven of the ten maximum variable changes and the change velocity are all lower on the slightly slower design. Because designer A values low changes along with speed, the slightly slower design is given a higher score.

4.7.3 Ten state

The results for the ten state frontier are shown in Figures 4.14a and 4.14b as well as three designs from each designer in Tables 4.17a and 4.17b. Designer A, on the left, can be seen to favor the upper left of the Pareto Frontier where the fastest designs can be found. Also, some of the designs at the bottom right of the frontier are favored with high utility scores. Conversely, Designer B focuses on the lower right of the frontier where designs are characterized by low standard deviations. As with the previous frontiers, these trends are consistent with the description of each designers preferences.

For Designer A, the split in favored designs between the upper left and lower right is an example of the preference for minimal changes manifesting in the utility of the designs. At the upper left, the aggregate utility function is dominated by the mean lap time. The risk averse utility curve and high preference weight associated with mean lap time combine to give these points high scores. For example, the fastest point has a mean lap time of 29.31 seconds which has a utility of 74.96. When this is combined with the 0.3 weight, the contribution to aggregate utility is 22.49. This can be compared to the maximum range of change measures which all contribute less than 1 to the aggregate utility. This is in contrast to the second cluster of high scoring points. In this cluster, the mean lap time contributes very little to the utility, 2.53 utils for the fastest of those points. Instead, more than half of the score is made up of the maximum range of change complexity measures. These contribute 8.16, 10.05, and 3.44 utils to the overall utility.

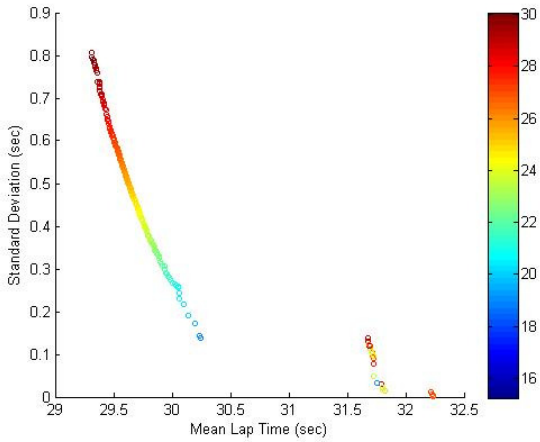

Figure 4.14a: Both Changed Designer A 10 State

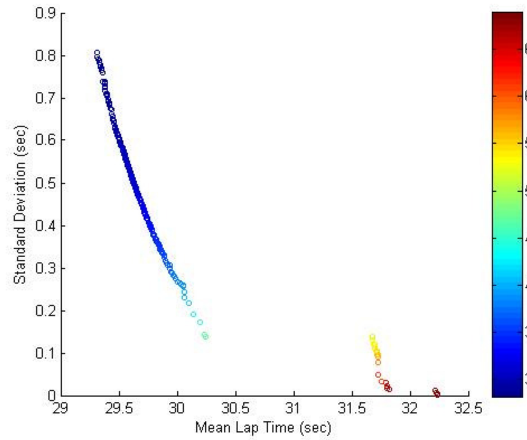

Figure 4.14b: Both Changed Designer B 10 State

Tables 4.17a Ten State Preferred Designs (Both Changes)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				High Score				
A	30.065	0.853	0.592	0.892	0.652	0.705	0.532	0.780
B	68.865	0.189	0.084	0.208	0.415	0.221	0.159	0.142
				Median Score				
A	25.494	0.839	0.609	0.892	0.652	0.697	0.760	0.862
B	31.489	0.853	0.609	0.892	0.652	0.702	0.727	0.782
				Low Score				
A	15.245	0.392	0.474	0.414	0.637	0.514	0.156	0.145
B	28.216	0.892	0.609	0.892	0.652	0.706	0.663	0.839

Table 4.17b Ten State Preferred Designs (Both Changes) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
			High Score				
A	0.560	0.542	0.746	30	1.023	29.31	0.808
B	0.529	0.350	0.085	30	0.931	32.23	0.002
			Median Score				
A	0.557	0.611	0.765	30	1.013	29.63	0.501
B	0.559	0.571	0.762	30	1.012	29.65	0.481
			Low Score				
A	0.566	0.563	0.306	30	0.946	31.73	0.090
B	0.562	0.777	0.798	30	1.022	29.36	0.759

In Tables 4.17a and 4.17b, the favored design for Designer A is characterized by a low mean lap time and high standard deviation. Also consistently, Designer B prefers a slower vehicle with a low standard deviation. Once again, the median designs are nearly the same between the two designers. Finally, the least favorable designs for each designer continue the trend of being consistent with preference description. Design A has a high mean lap time and low standard deviation while design B has the opposite for both categories.

4.7.4 Combined analysis

The final set of figures, Figures 4.15a and 4.15b, presents the data for all three Pareto Frontiers compiled into one figure, and Tables 4.18a and 4.18b show a design from each frontier for each designer. The data for Designer A is on the left where it can be seen that the fastest designs from both the two state and five state frontiers are the most favored. The least favored designs are found at the slow end of the ten state and five state frontiers. The information for Designer B, on the right, shows that the designer focuses on the bottom section of all of the frontiers. This is consistent with the fact that Designer B favors consistent vehicles with low standard deviations.

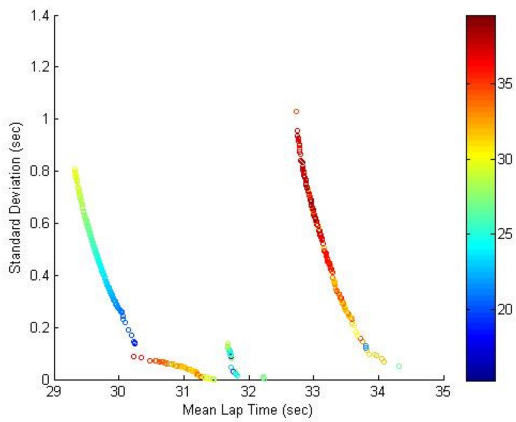

Figure 4.15a: Both Changed Designer A All States

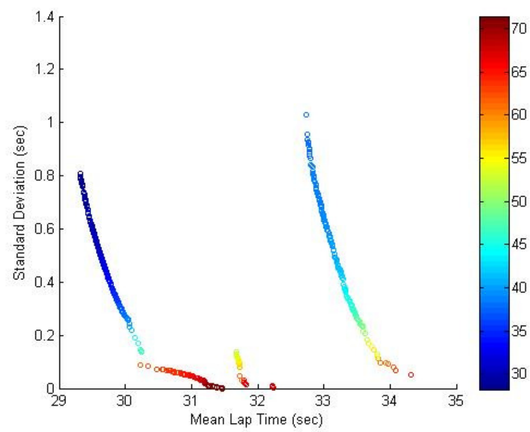

Figure 4.15b: Both Changed Designer B All States

Table 4.18a All Frontiers Preferred Designs (Both Changes)

	Score	Fuel Position	Roll Stiffness	F. AOA	R. AOA	F. Max Camber	F. Camber Distance	F. Max Thickness
				Two State				
A	37.08	0.056	0.163	0.672	0.559	0.431	0.107	0.106
B	71.46	0.052	0.233	0.598	0.544	0.318	0.162	0.068
				Five State				
A	39.60	0.035	0.117	0.179	0.104	0.074	0.232	0.070
B	64.92	0.224	0.362	0.133	0.158	0.130	0.304	0.219
				Ten State				
A	30.065	0.853	0.592	0.892	0.652	0.705	0.532	0.780
B	68.865	0.189	0.084	0.208	0.415	0.221	0.159	0.142

Table 4.18b All Frontiers Preferred Designs (Both Changes) Continued

	R. Max Camber	R. Camber Distance	R. Max Thickness	Num. of Changes	Change Velocity	Mean Lap Time	Stand. Dev.
				Two State			
A	0.641	0.016	0.056	30	0.993	30.22	0.090
B	0.636	0.018	0.071	30	0.953	31.46	0.000
				Five State			
A	0.140	0.073	0.165	5	0.152	32.96	0.684
B	0.133	0.231	0.364	5	0.146	34.32	0.050
				Ten State			
A	0.560	0.542	0.746	30	1.023	29.31	0.808
B	0.529	0.350	0.085	30	0.931	32.23	0.002

Tables 4.18a and 4.18b show that Designer A favors designs from the two state Pareto Frontier even though the ten state frontier contains designs with lower lap times. The table shows the most highly favored two state and ten state designs for Designer A. While the ten state design has a lower mean lap time, the two state design has lower values for nine of the ten maximum variable change measures as well as the change velocity measure. Therefore, it can be seen that because Designer A is interested in not only speed but also minimal changes, the focus is shifted to the simpler designs on the two state frontier. Similarly, while the ten state point has a better lap time than the five state, the values for all of the complexity measures on the five state design are lower.

For Designer B, the focus was placed on consistency at the cost of changes and speed. Therefore, the lowest standard deviations were the most highly favored without variation.

4.8 Conclusions

This chapter addressed the second research question posed in Chapter 1. To answer this question, two aspects of choosing a design are examined. The first addresses the trade-off between performance and consistency along a Pareto Frontier, and the second addresses the trade-offs between performance and complexity in choosing the level of reconfiguration in a design. These are two important choices facing a designer wishing to implement a reconfigurable system.

To answer both aspects of the question a two step process was followed. The first step involved determining which aspects of the system would affect the trade-off decision being made. This first step also involved quantifying the aspects of the system which affect the decision. At this point the complexity measures were introduced to quantify a number of sources of complexity that may be important to the designer.

The second step of the process involved comparing the final reconfiguration scheme and system decisions across designers with varied preferences. Instead of surveying a large number of individuals, the preference portraits in question were simulated using an algorithm explained in the chapter. This created a number of so called virtual designers whose preferences were applied to the trade-off decisions in question.

The impact of designer preferences was seen on the choice of specific designs when presented with a Pareto Frontier. Two virtual designers were presented first with different utility curve profiles, second with different weighting schemes, and third with unique utility curve profiles and weighting schemes. Each designer was presented with all three of the Pareto Frontiers, and each design was given a score based on the aggregate utility.

The section with varied utility curve profiles showed that the changes in utility curves could have a large effect on the Pareto Frontier region choice. For example, on the five state scheme the designers with different utility curve profiles targeted completely opposite sections of the Pareto Frontier. In contrast, on the two state and ten state frontiers, the designers targeted nearly identical sections of the frontiers. In addition, it could be seen that the effects of the utility curve profiles did not always match the predictions based on stated designer preferences.

This is due to a combination of effects. The first is the high weight of mean lap time in the randomized shared weighting scheme. This allowed high mean lap time utility scores from Designer

B to dominate the aggregate utility equation. The effect is enhanced by the risk prone utility curve Designer B assigns to mean lap time. On this curve, the utility decreases slowly as the attribute value moves away from the ideal, which results in slow lap times receiving relatively high scores. The result is that on some frontiers, Designer B targets fast designs while the preferences suggest more consistent designs should be favored.

The second driving factor in the results is the risk averse utility curve associated with mean lap time for Designer A. This curve is characterized by a sharp decrease in utility with small movements away from the optimal value. Therefore, when the mean lap times are high they receive very low utilities from Designer A. This is evidenced on the five state frontier where all of the lap times are relatively high, and the corresponding utilities are all very low. This causes the second highest weight in the weighting scheme, standard deviation, to dominate the aggregate utility equation. The end result of this is that Designer A targets the bottom left of the frontier with slow consistent designs when preferences suggest fast designs should be favored.

The weighting scheme variation offered large differences in the targeted location on the Pareto Frontier for each designer. For two of the three frontiers, the designers targeted opposite ends of the frontiers, and the third showed a difference in the range of desired designs. These differences were more consistent with the stated designer preferences than the changes seen with the utility curve profile variation. Generally, designer A focused on designs with low mean lap times while designer B targeted designs with low standard deviation. The only deviation from this trend came on the ten state frontier where other preferences asserted more influence on the design choice of designer A. In this instance, the desire to have minimal changes in the final design, drove designer A to target slower systems to avoid large changes. While this does not fit the trend shown in the other frontiers, it does maintain consistency with the stated designer preferences.

Finally, the combined variation, both utility curves and weighting scheme, showed the same large differences as seen with the previous iterations of the investigation. The designers targeted opposite ends of the frontier, keeping with their preferences on speed and consistency.

The other aspect of the question, performance versus complexity, is addressed in the sections that compare all three frontiers simultaneously. This analysis revealed each designers' desire for designs on different frontiers. As with the individual frontiers, when the utility curve profile is varied between designers, there is relatively little difference in their frontier preferences. This changes when the weighting scheme for each designer is varied. With weighting scheme variation, each designer targets the two state frontier. This frontier contains designs that are fast, consistent, and simple. Designer B also targets the lowest sections of each of the other frontiers because they offer low standard deviations. Designer A, in addition to the two state frontier, targets the fastest designs on the five state frontier. These designs offered some speed, but are also simple. The ten state frontier is not targeted because of its complexity. Similar results can be seen when examining the section where both utility curve profile and weighting scheme are changed. Once again, both designers heavily favor the two state frontier. Designer B maintains the desire for low standard deviations, with a focus on the lower end of the other two frontiers. Designer A focuses on fast and simple designs. This leads the focus to the top of the five state frontier, the two state frontier, and the top of the ten state frontier. The fastest designs on the ten state frontier contribute enough of a speed increase to overcome the desire for simple designs. Depending on the importance of the attributes, all three of the schemes examined here are viable choices, but if emphasis is placed on the reduction of complexity, the ten state scheme is rarely chosen.

It can be seen from the simulation of virtual designer preferences that there is a large effect on both the choice of design on a Pareto frontier and the choice of reconfiguration scheme. Each of the trade-off decisions examined here is sensitive to the individual designer preferences which cause changes in both frontier location and scheme choice. The investigation showed the importance of presenting designers with multiple choices because, depending on preferences, designers my target very different final designs.

CHAPTER 5

Conclusions

5.1 Research question outline

Chapter 2 of this work illustrated that reconfigurability has been applied to many areas, and it has been shown to increase system performance and efficiency which makes it an attractive area for further study [8], [14], [20]. The goal of this research is to answer two overarching research questions, both of which pertain to the idea of reconfigurable systems. The first research question is:

How effective is reconfigurability at mitigating the performance impact of uncontrollable system variations?

This question is intended to address the ability of reconfigurable systems to negate the sacrifices caused by a trade-off between performance and consistency. To answer this question, a variety of reconfiguration schemes are introduced with the main goal of overcoming the performance sacrifice necessary to create a consistent system.

The introduction of reconfigurability necessitates another trade-off decision. The reconfigurable systems introduce increased complexity when compared to a static system. Therefore, if reconfigurability is implemented, a designer must decide on an acceptable level of complexity to allow in order to attain a given increase in performance. These trade-off decisions are addressed by the second question:

How sensitive to designer preference are the tradeoff decisions associated with choosing a reconfigurable system?

In order to answer the questions, a race car model case study problem is used which allows the simulation of both reconfigurable systems and robust systems. It also provides a benchmark system with which to compare the other ideas.

5.2 Research question 1

The first research question is answered in chapter 3. This chapter outlines an approach to answer the research question applying it to a racecar case study model to answer the question. The approach includes identifying the problem, identifying reconfigurable variables, establishing limits to reconfigurabitlity, designing the reconfigurable system, and finally comparing it to benchmark systems. In the case study model, the uncontrollable variations cause performance fluctuations and inconsistency that both the robust and reconfigurable systems are attempting to mitigate. The variations prompt the need for the trade-off decision between performance and consistency introduced by Robust design. For the race car model, these include changing fuel levels which cause a dynamic center of gravity and tire wear which decreases the tire effectiveness over the course of a race.

For limiting the reconfigurability of the system, Epoch-Era analysis is introduced as a means of discretizing the lifecycle of the system. The discrete time intervals serve as moments of consistent conditions, and the transition points between Epochs serve as transition points between configurations. Therefore, a unique end state is assigned to each Epoch, and each reconfiguration scheme is characterized by its number of end states.

Three different limitation strategies are introduced which produce three reconfiguration schemes to investigate the trade-off between performance and consistency. The two state scheme relies on an Epoch transition every time the race track changes objectives from a turn to a straight and vice versa. The unique end states are a turning state and a straight state. The next scheme is determined by an Epoch transition every time the uncontrollable variations update. This occurs once per lap for five laps which produces five snapshots of race conditions and performance translating to a five state scheme. The final scheme is the ten state scheme which combines the Epoch transition criteria of the previous two. For this scheme, the configuration changes between turning and straight, but there is a unique turning and straight configuration for each lap.

The focus of the first research question is to address consistency as well as performance. This involves a multi-objective optimization approach where the two objectives are system performance measured as mean lap time, and system robustness measured as standard deviation of lap times. This tactic produces a Pareto frontier of designs ranging from faster configurations with more variation to slower configurations with low variations. The frontier shows the trade-off decision and the sacrifice associated with making it. This approach produced three Pareto frontiers of reconfigurable vehicles.

The next step is to compare the reconfigurable systems to a benchmark, a static system designed using Robust design. The static race car is optimized in the presence of the uncontrollable variations. Additionally, the multi-objective robust design method is applied to the static system to produce a Pareto frontier of design choices. From the results, it is clear that two of the reconfigurable systems, two state and ten state, improve on the performance of the robust Pareto frontier. They dominate all other designs on the mean lap time axis, and they dominate the some of the static system Pareto frontier on the standard deviation axis. The five state reconfiguration scheme did not show improvement over the static Pareto frontier.

With respect to the first research question, these simulations demonstrate that reconfigurable systems can compete with the system created using robust design strategies in both performance and the mitigation of performance variations. Two of the reconfigurable systems show 8% to 12% improvements over the lap times of the static system with similar standard deviations. Therefore, the answer to the question is that the reconfigurable systems can be comparable to the robust system in performance variation mitigation, and superior in maximum lap time performance. The reconfigurable systems not only matched the robust system in variation mitigation, but they negated the performance sacrifice associated with this trade-off.

Some additional conclusions can be drawn from the research in Chapter 3. First, the simulations demonstrated that reconfigurability is not a one size fits all solution to trade-off decisions, and not all forms of reconfiguration are equally suited to negating sacrifices due trade-offs. The three different implementations target different locations in the design space. This suggests that the most effective applications of reconfigurability will be those tailored specifically for the system in question, but also tailored for the particular desired results. Second, it can be seen that the choice of how to limit the reconfigurability in a system can greatly impact the final results for the design. This is illustrated by the fact that the three schemes have varying levels of success addressing the sacrifices associated with the trade-off decision. The five state scheme is only as effective at variation mitigation as a static system while the two and ten state schemes improved both variation mitigation and system performance. This reinforces the notion that effective reconfigurability will be tailored for specific systems and results. Also, it is important for designers to carefully choose the implementation methods of reconfigurability for their system.

5.3 Research question 2

One of the greatest drawbacks to a reconfigurable system is increased complexity both physically and computationally [11]. The second research question stems from this fact. It has been shown that fully reconfigurable systems and even the simplified reconfigurable system seen in the previous simulations improve system performance. Balancing the “cost” of reconfigurability with the performance benefits added to the system introduces an additional trade-off decision performance versus complexity. Because of the complication associated with balancing factors in complex design, this decision outcome depends highly on designer preferences. The second research question relates to this fact.

To answer the question, two virtual designers are created with differing preferences concerning an ideal design. An algorithm is used to apply their preferences to the Pareto frontiers created in Chapter 3, and the results are displayed as a heat map. In the heat map figures, the design space locations that each designer targets can be visualized.

The attributes where designer preferences are manifested include several different measures of complexity such as total design variable change, number of states, number of changes, and change velocity. The attributes also include system performance as measured by mean lap time, and robustness as measured by standard deviation of lap times. Each virtual designer is constructed with a set of preferences which represent their attitudes towards each attribute. For this investigation, Designer A prefers fast racecars with minimal complexity, and Designer B is willing to include complexity but prefers consistent systems. The preference profile of each designer is a combination of utility curves and weighting schemes that are applied to each attribute and aggregated to determine a utility score for each design. The preferences are applied to each design in each reconfiguration scheme, and a total utility score is determined with an aggregate utility equation.

The preference profiles of the designers are made up of two parts, utility curve profile and weighting scheme. For this investigation, each part is tested individually followed by a combined simulation. First, the weighting scheme is held constant between the two designers, and the utility curve profile is varied to reflect the different perspectives. This is followed with a constant utility curve profile and a varied weighting scheme. Finally, each designer is given a unique utility curve profile and weighting scheme.

All three sections of the investigation show an often divergent focus between the two designers. Each targets an opposite end of the Pareto frontier. This implies that the preferences have a noticeable impact on the performance versus robustness trade-off decision. Additionally, the designers target different reconfiguration schemes when all three are compared together. This implies that the preferences also affect the performance versus complexity trade-off decision.

As a general rule, each designer targets locations consistent with the obvious difference in preferences. Designer A mostly prefers fast designs while Designer B maintains interest in consistent designs. There are several instances however that do not follow this trend. Close inspection shows that these deviations from the trend are due to subtle effects of either the structure of the preference profiles or the preferences concerning attributes beyond the performance versus robustness trade-off. For example, when relatively slow designs are evaluated, Designer A's interest in minimal complexity overrides the interest in fast systems. This results in a focus on slow simple designs. Another example can be seen when a single weight is much larger than the others, or a utility curve is especially forgiving. For the first section, the performance weight is significantly larger than the other weights, and Designer B applies a risk prone utility curve to this attribute. The combination of the high scores given by the curve and the high weight on those scores causes Designer B to focus on fast designs.

The results show that the designer preferences have a large impact on the performance versus robustness trade-off decision, causing designers to choose opposite ends of the frontiers. It can also be seen, that the preferences affect the performance versus complexity trade-off decision as well. Both of the designers value the two state scheme frontier because it has a mix of fast and consistent designs. The difference between the designers is manifested in their assessments of the other two schemes. Designer B focuses solely on the lower sections of the five state and ten state frontiers. These sections are populated by the most consistent designs. Designer A targets the top of the five state scheme. These designs offer low mean lap times while maintaining simple designs as measured by the complexity measures.

In regards to the second research question, it can be seen that the trade-off decisions are heavily dependent on the designer preferences. Additionally, the decisions are even dependent on nuances in the particular preference scheme structure being used. These results suggest that it is beneficial to offer designers of complex systems a variety of choices when considering trade-off decisions.

These investigations demonstrate the importance of choice in the design process. The addition of reconfigurability gives designers a means of mitigating performance variations outside of traditional Robust design, while improving overall system performance. Additionally, the different reconfiguration schemes inhabit a variety of locations in the design space, and introduce varying levels of complexity allowing designers greater flexibility when attempting to balance factors in a complex system. This fact is bolstered by the use of multi-objective optimization which provides an set of optimal systems that span the spectrum of the trade-off decision. For the second research question, it is shown that different designer preferences can greatly affect the outcome of trade-off decisions. The inclusion of an expanded array of choices would ensure that designer preferences are more likely to be realized in the final system decision.

The research presented here has led to a variety of avenues for future research. For example, the introduction of the uncontrollable variations highlighted the fact that for this particular model track, geometries had a greater effect than variations on system performance. This can be seen by the fact that the two schemes that address track geometry, two and ten state, outperformed the scheme that addresses variations alone. This could change for other track conditions or case study models. Therefore a possible topic of future research would be to explore the relationship of uncontrollable variations and track geometry, and the affects they have on reconfigurable systems. Also, the method introduced for limiting the complexity in the system focuses on end states of the reconfiguration process. There are a variety of other system properties that could be used to limit reconfigurability and complexity. Some of these include the frequency at which the system changes, the range that the design variables are allowed to change, and which design variables are allowed to change. Any of these can perhaps also reduce the complexity of the reconfigurable system while maintaining its performance, and would therefore offer promising avenues for future research.

REFERENCES

[1] Simon, Herbert A. "The architecture of complexity." *Proceedings of the American philosophical society* 106, no. 6 (1962): 467-482.

[2] Privošnik, M., and Marolt, M., "The Development of Emergent Properties in Massive Multi-Agent Systems." *Advances in Automation, Multimedia and Video Systems and Modern Computer Science*. (2001): 240-242. Print.

[3] Kalsi, M., Hacker, K., and Lewis, L., "A comprehensive robust design approach for decision trade-offs in complex systems design." *Journal of Mechanical Design* 123 (2001): 1.

[4] Jarratt, T., Eckert, C., and Clarkson, P. J., "Pitfalls of engineering change." In *Advances in Design*, pp. 413-423. Springer London, 2006.

[5] Clarkson, P. John, Caroline Simons, and Claudia Eckert. "Predicting change propagation in complex design." *Journal of Mechanical Design* 126 (2004): 788.

[6] Jarratt, Timothy, C. M. Eckert, and P. John Clarkson. "Development of a product model to support engineering change management." *Proceedings of the TCME* (2004): 12-16.

[7] Ethiraj, S., and Levinthal, D., "Modularity and innovation in complex systems." *Management Science* 50, no. 2 (2004): 159-173.

[8] Ferguson, S., and Lewis, K., “Effective Development of Reconfigurable Systems Using Linear State-Feedback Control,” *AIAA Journal*, 2006, 44(4):868-878.

[9] Fox, M., Finger, S., Gardner, E., Navin Chandra, D., Safier, A., and Shaw, M., "Design Fusion: An Architecture for Concurrent Design." *Knowledge Aided Design* (1992): 157-195.

[10] Pareto, V., 1906, *Manuale di Econòmica Polìttica,* Società Editrice Libràia, Milan, Italy; translated into English by A. S. Schwier, as Manual of Political Economy, Macmillan, New York, 1971.

[11] Ferguson, S., Lewis, K., de Weck, O., and Siddiqi, A., “Flexible and Reconfigurable Systems: Nomenclature and Review,” *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Las Vegas, NV,DETC2007/DAC-35745, 2007.

[12] Tilstra, A., Ferguson, S., Seepersad, C., Wood, K., “Development of a Changeable Airfoil Optimization Model for use in the Multidisciplinary Design of Unmanned Aerial Vehicles,” *ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, San Diego, CA, DETC2009-87482, 2009.

[13] Olewnik, A., and Lewis, K., “A Decision Support Framework for Flexible System Design,” *Journal of Engineering Design,* 2006 17(1): 75-97.

[14] Ferguson, S., Kasprzak, E., and Lewis, K., “Designing a family of reconfigurable vehicles using multilevel multidisciplinary design optimization,” *Structural and Multidisciplinary Optimization*, vol. 39, no. 2, pp. 171–186, 2009.

[15] Martin, E., and Crossley, W., “Multiobjective Aircraft Design to Investigate Potential Geometric Morphing Features,” 2nd AIAA ATIO Forum, Los Angeles, CA., AIAA Paper AIAA-2002-5859, 2002.

[16] Roth, B., and Crossley, W., “Application of Optimization Techniques in the Conceptual Design of Morphing Aircraft,” 3rd Annual ATIO Forum, Denver, CO., AIAA 2003-6733, 2003.

[17] Mourelatos, Z., and Liang, J., "An efficient unified approach for reliability and robustness in engineering design." In *NSF Workshop on Reliable Engineering Computing*, pp. 127-138. 2004.

[18] Hacker, K., Lewis, K., “Robust Design Through the use of a Hybrid Genetic Algorithm,” *ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, Montreal, Canada, DETC2002/DAC-34108, 2002.

[19] Kumar, A., Keane, A., Nair, P., and Shahpar, S., "Robust design of compressor fan blades against erosion." *Journal of Mechanical Design* 128 (2006): 864.

[20] Siddiqi, A., de Weck, O., and Iagnemma, K., “Reconfigurability in Planetary Surface Vehicles: Modeling Approaches and Case Study,” *Journal of the British Interplanetary Society*, 59, 2006.

[21] Siddiqi, A., and de Weck, O., “Modeling Methods and Conceptual Design Principles for Reconfigurable Systems,” *Journal of Mechanical Design*, 2008, **130**(10): 101102**.**

[22] de Weck, O., and Suk Suh, E, “Flexible Product Platforms: Framework and Case Study,” *ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, Philadelphia, PA, DETC2006-99163, 2006.

[23] Lewis, P., Mattson, C., and Murray, V., “An engineering design strategy for reconfigurable products that support poverty alleviation,” *ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference* , Montreal, Quebec, Canada, DETC2010-28739, 2010.

[24] Sullivan, E., Tortorice, M., and Ferguson, S., “Using Design Reconfigurability to Mitigate the Effects of Uncontrolled System Variations,” presented at the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, Fort Worth, TX, 2010.

[25] Singh, V., et al. "Design for transformation: Theory, method and application." *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Las Vegas, NV, DETC2007-34876 2007.

[26] Singh, V., Skiles, S., Krager, J., Wood, K., Jensen, D., and Sierakowski, R., “Innovations in design through transformation: A fundamental study of transformation principles”. Journal of Mechanical Design, 131 August, 2009.

[27] Parkinson, M., Haldaman, J. “Reconfigurable products and their means of reconfiguration,” *ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, Montreal, Quebec, Canada, DETC2010-28528, 2010.

[28] Kuhr, R., Wood, K., Jensen, D., and Crawford, R., “Concept Opportunity Diagrams: A Visual Modeling Method to Find Multifunctional Design Concepts,” *ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, Montreal, Quebec, Canada, DETC2012-29068, 2010.

[29] Hazelrigg, G., “A Framework for Decision-Based Engineering Design,” *ASME Journal of Mechanical Design*, 1998, **120**: 653-658.**)**

[30] Khire, R., and Messac, A., “Selection-Integrated Optimization (SIO) Methodology for Optimal Design of Adaptive Systems,” *Journal of Mechanical Design*, 2008, **130**(10): 101401.

[31] Ferguson, S., and Lewis, K., “Investigating the Interaction between Reconfigurability and System Mass using Multidisciplinary Design Optimization,” *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, Schaumburg, IL, AIAA-2008-1803, 2008.

[32] Chmarra, M., Verriet, J., Waarsing, R., and Tomiyama, T., “State Transition in Reconfigurable Systems,” *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Montreal, Quebec, Canada, DETC2010-28723, 2010

[33] Taguchi G. *Introduction to Quality Engineering—Designing Quality into Products and Processes*. Asian Productivity Organization: Tokyo, 1986. **(Introduction to Quality)**

[34] Park SH. *Robust Design and Analysis for Quality Engineering*. Chapman & Hall: London, 1996.

[35] Ross PJ. *Taguchi Techniques for Quality Engineering*. McGraw-Hill: New York, 1996.

[36] Clausing D. *Total Quality Development—A Step-By-Step Guide to World-Class Concurrent Engineering*. ASME Press: New York, 1994.

[37] Phadke, M., *Quality Engineering Using Robust Design*, Prentice-Hall, Englewood Cliff, NJ, 1989.

[38] Taguchi G, Wu Y. *Introduction to Off-Line Quality Control*. Central Japan Quality Control Association: Nagoya, Japan, 1979.

[39] Hunter JS. Statistical design applied to product design. *Journal of Quality Technology* 1985; **17**:210–221.

[40] Kackar RN. Off-line quality control, parameter design, and the Taguchi method (with discussion). *Journal of Quality Technology* 1985; **17**:176–188.

[41] Goh TN. Taguchi methods: Some technical, cultural and pedagogical perspectives. *Quality and Reliability Engineering International* 1993; **9**:185–202.

[42] Deb, K., and Gupta, H., "Introducing robustness in multi-objective optimization," *Evolutionary Computation* 14, no. 4 (2006): 463-494.

[43] Kang, J., Lee, T., and Lee, D., "Robust optimization for engineering design," *Engineering Optimization* 44, no. 2 (2012): 175-194.

[44] Jin, Y., and Sendhoff, B., "Trade-off between performance and robustness: an evolutionary multiobjective approach," *Evolutionary Multi-Criterion Optimization*, pp. 237-251. Springer Berlin Heidelberg, 2003.

[45] Ross, A., and Rhodes, D., "Using natural value-centric time scales for conceptualizing system timelines through epoch-era analysis." *INCOSE International Symposium*. 2008.

[46] McManus, H., Richards, M., Ross, A., and Hastings, D., "A Framework for Incorporating' ilities' in Tradespace Studies," in *AIAA Space*, 2007.

[47] Ji, H., Yang, M., Honda, T., "A Probabilistic Approach for Extracting Design Preferences from Design Team Discussion," *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference,* Las Vegas, NV, DETC2007-35455, 2007.

[48] Von Neumann, J., and Morgenstern, O., *Theory of games and economic behavior (commemorative edition)*. Princeton university press, 2007.

[49] Scott, M., and Antonsson, E., "Compensation and weights for trade-offs in engineering design: beyond the weighted sum." *Journal of Mechanical Design* 127 (2005): 1045.

[50] See, T., and Lewis, K., "Multiattribute Decision Making Using Hypothetical Equivalents," *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference,* Montreal, Ontario, Canada, DETC2002/DAC-34079, 2002.

[51] Kulok, M., and Lewis, K., "A method to ensure preference consistency in multi-attribute selection decisions." *Journal of Mechanical Design* 129 (2007): 1002.

[52] Hacker, K., Kasprzak, E. M., and Lewis, K., "Racecar Optimization and Tradeoff Analysis in a Parallel Computing Environment," SAE Motorsports Engineering Conference & Exposition, Dearborn, MI., SAE. Paper 2000-01-3564, 2000.

[53] Kasprzak, E., "Modeling race car performance variations," E-mail to Scott Ferguson, June 11, 2010.

[54] Gumasta, K., Gupta, S., Benyoucef, L., and Tiwari, M., "Developing a reconfigurability index using multi-attribute utility theory," *International Journal of Production Research*, vol. 49, no. 6, pp. 1669–1683, 2011.

[55] 2013 IZOD Indycar series Rulebook

[56] Wassenaar,H., and Chen, W., "An approach to decision-based design with discrete choice analysis for demand modeling." *Journal of Mechanical Design* 125 (2003): 490.

[57] Gurnani, A., See, T., Lewis, K., "An Approach to Robust Multiattribute Concept Selection," *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference,* Chicago, IL, DETC2003/DAC-49707, 2003.