

DRAFT DETC 2008-49359

A RISK-INFORMED DECISION MAKING METHODOLOGY FOR EVALUATING FAILURE IMPACT OF EARLY SYSTEM DESIGNS

Tolga Kurtoglu

Mission Critical Technologies
NASA Ames Research Center
Intelligent Systems Division
Moffett Field, California, 94035
tolga.kurtoglu@nasa.gov

Irem Y. Tumer

Associate Professor
Complex Engineered Systems Design Lab
Department of Mechanical Engineering
Oregon State University, Corvallis, Oregon 97331
irem.tumer@oregonstate.edu

ABSTRACT

In this paper, we introduce a new risk-informed decision-making methodology for use during early design of complex systems. The proposed approach is based on the notion that a failure happens when a functional element in the system does not perform its intended task. Accordingly, risk is defined depending on the role of functionality in accomplishing designed tasks. A simulation-based failure analysis tool is used to analyze functional failures and their impact on overall system functionality. The analysis results are then integrated into a decision-making framework that relates the impact of functional failures and their propagation to decision making in order to guide system level design decisions. With the help of the proposed methodology, a multitude of failure scenarios can be quickly analyzed to determine the effects of decisions on overall system risk. Using this decision-making approach, design teams can systematically explore risks and vulnerabilities during early, functional stage of system development prior to the selection of specific components. Application of the presented method to a reservoir system design demonstrates these capabilities.

Keywords: Risk-Based Design, Decision-Based Design, Functional Modeling, Simulation-Based Design.

1 INTRODUCTION

The identification of risks of losing functionality during the earliest stages of designing complex systems is of growing importance for risk sensitive industries. Early stage design provides the greatest opportunities to explore design alternatives and perform trade studies before costly decisions are made. The goal of this research is to develop a formal

framework that enables risk-informed analysis of complex system design decisions during the conceptual design phase. The analysis of potential failures and associated risks of functional losses performed at this earliest stage of design will facilitate better design decision making, and thus the development of more robust and reliable system architectures [1-3].

Many methods have been introduced in recent years to move risk based analyses and decisions into the early stages of design. The intended goal of what we generally call Risk Based Design (RBD) is to use formal methods to understand and characterize risk drivers as the design develops and incorporates this information into principles, tools, or methodologies. The methods are then intended to assist designers in making design decisions that reduce risk while meeting overall system goals. However, a majority current risk-based design techniques are reliability analysis and optimization methods applied to system design. While of great merit, these techniques remain difficult to apply during the earliest, functional stage of design, requiring data and models about a design at a fidelity level that is not typically available during early design stages. As a result, this research is motivated by the need to have formal processes and tools to quantify risk as early as *functional* design and to guide design decision making accordingly.

To achieve this goal, one must identify functions, risks, and failure modes related to design decisions and enable making design decisions and choices based on risk and failure information. One way of doing that is by understanding the nature of the failure and its impact on the functionality of the system. This kind of impact assessment requires establishing the relationship between components and their failure modes, the

functionality of components, and the propagation of failure effects.

In prior work, we have introduced the Functional Failure Identification and Propagation (FFIP) analysis framework that integrates all these aspects into a formal framework to enable the analysis of functional failures and their impact on overall system functionality [3]. In this paper, we extend the analysis capabilities of the FFIP framework by: 1) integrating quantifiable measures that define risks based on the role of functionality in accomplishing design goals, and, 2) relating impact analysis results to decision making in order to guide system level design decisions based on functional failure potential. Specifically, the FFIP analysis results are integrated into a decision-making framework that relates the impact of functional failures and their propagation to decision making in order to guide system level design decisions. The proposed risk informed decision making methodology offers opportunities for significant reduction in cost, and increases in system safety and reliability by enabling early development of preventive measures that can effectively and efficiently guard against system failures.

An important aspect of such an impact analysis is the determination of redundancy and criticality following failures. The level of redundancy of critical systems and the criticality of the functions affected by the failure are two crucial pieces of information necessary to determine best design decisions and thus are modeled in the method presented here. Two examples are shown in this paper illustrating how the relationship between system redundancy and criticality of functional failures can be explored to analyze different design decisions using the FFIP-based failure-informed decision making framework.

2 RELATED WORK

Risk, reliability, failure analysis, and decision-making under uncertainty have received much attention in industry and the research community alike. This section presents a brief survey of three fields of direct relevance, namely: Risk Assessment, Reliability Based Design, and Decision Based Design. This review will be followed by a brief chronological summary of the research in function-based failure analysis in order to motivate the methodology introduced in this paper.

2.1 Review of Risk Assessment Methods

A review of various risk and reliability based assessment techniques was presented in prior work [3], and is repeated here for completeness. Risk-sensitive industries designing complex systems currently employ three major techniques to assess the reliability of their products: FMEA, FTA, and PRA. Failure Modes and Effects Analysis (FMEA) [4] is a method that systematically examines individual system components and their failure mode characteristics to assess risk and reliability. The FMEA analysis starts with decomposition of the system into subsystems and finally into individual components. Ways

in which each component can potentially fail are then recorded and evaluated separately to determine what effect they have at the component level, and then at the system level. It is a widely used method that is easy to understand and implement. However, the analysis requires a detailed level of system design, and thus is not optimal to be used during conceptual design [3]. Moreover, FMEA does not capture component interactions explicitly, and relies heavily on expert knowledge to assess failure consequences and their criticality [3]. As a result, it is often considered to be a highly subjective method.

Fault Tree Analysis (FTA) [5] is performed to capture event paths from failure root causes to top-level consequences. Using this approach, possible event paths from failure root causes to top-level consequences can be captured. When conducted properly, it is likely to identify more possible failure causes than single-component oriented FMEA. However, FTA also relies greatly on expert input and shares similar criticism as FMEA [3]. Moreover, since the failure domain is represented using events in FTA, low-level component interactions and dynamics leading to failure are only considered informally, during expert identification of event-consequence relationships. Formally capturing component interactions and system dynamics of complex systems is however crucial for supporting design decisions during early concept development of complex systems.

Probabilistic Risk Assessment (PRA) [6] is a method used for quantification of failure risk [7]. PRA combines a number of fault/event modeling techniques such as master logic diagrams, event sequence diagrams and fault trees, and integrates them into a probabilistic framework to guide decision-making during design. Recently, PRA has been extended to include event/behavior simulation into the analysis as demonstrated by the SIMPRA tool developed by Mosleh et al. [8], but this extension demands a fully-specified system model as part of the analysis. Such detailed, high-fidelity models of complex systems, however, are not available during conceptual design.

In addition, a variety of diagnostic reasoning tools have been proposed for fault assessment and diagnosis of fault propagation. Diagnostic reasoning approaches share a common process in which a system is monitored and a comparison is performed of observed and expected behavior of the system to detect anomalous conditions [9]. Model-based approaches to diagnosis, on the other hand, rely mostly on qualitative knowledge to predict the behavior of a system [10]. When observations disagree with the predicted behavior, some diagnostic technique is initiated to identify the faults. The broadest category for diagnostic reasoning is model-based diagnosis (MBD) [11-14]. Among those, directed graphs [15] are one of the techniques used to analyze component dependencies and fault propagation [16, 17]. Multi-Signal Flow Graphs developed by Deb et al. [18] is another comprehensive methodology to model cause-effect dependencies of complex

systems. Other methods include statistical and probabilistic classification methods [19, 20]. As was discussed by Kurtoglu and Tumer in [3], while fault propagation analysis tools exist, they require designers to explicitly formulate a fault propagation model by specifying paths of causal relationships, which is not feasible during the early stages of design where information about the system specifics is scarce.

Finally, historical anomaly and lessons learned types of databases (LLIS, PRACA, ASIAs, etc.) also provide a way of documenting risk and failure information [21]. However, the information is system specific (does not capture design context), and the methods using these databases do not provide analysis capabilities and hence cannot directly be applied design decision making.

Current risk assessment methods share an after-the-fact approach that looks at effects and traces them back to the causes of those effects. Using the risk-informed decision making method presented here, we aim to eliminate or reduce the likelihood of reaching certain possible futures by formal analysis of risk of failures early in the design process and by proper guidance of decisions before the design becomes solidified.

2.2 Review of Reliability-Based Design Methods

To move risk and failure analysis into the early stages of design, reliability based methods have been introduced in recent years in various forms. Many of these efforts have focused on using robust design, uncertainty estimation and reliability based optimization methods. Examples can be found in [22-33]. Robust design techniques focus on system quality by minimizing performance variance, whereas reliability-based design methods seek to design systems that achieve an invariably small, targeted probability of failure, ensuring proper system functioning. Among these, reliability based design optimization techniques have successfully been used in various engineering fields including aerospace engineering [34].

The intended goal of these methods is to use formal methods to understand and characterize risk drivers as the design develops, and incorporate this information into principles, tools, or methodologies. The methods are then intended to assist designers in making design decisions that reduce risk while meeting overall system goals. However, most current techniques are reliability analysis methods proposed for system design, and hence suffer from the same issues as the current reliability assessment methods, in that, they require more information and higher-fidelity models than is available during functional design. In particular, this research focuses on analyzing the propagation of potential failures and the resulting functional losses during the early stages of functional design, which are not directly obtainable from such techniques.

2.3 Review of Decision-Based Design Methods

Decision-making has been recognized as an integral part of the engineering design process in all phases of design. In particular, the uncertainty that is associated with the decisions made during the early design stages has been acknowledged as a major source of risk to the design. To address this issue, Decision Based Design (DBD) has emerged as a potential solution to optimize decisions made during design [35], and hence improve the decision making process through the application of rigorous mathematical principles from decision theory [36, 37]. The various methods proposed are largely based on concepts from game theory, utility theory, voting, and preference modeling, and has its roots in decision science, economics, and operations research [38-41]. Many design researchers have worked on finding utility functions and preferences that work for the engineering design process [42-45]. Furthermore, in a real design environment, numerous decisions have to be made with multiple and potentially conflicting criteria or attributes of the product [37]. Many of the later efforts have addressed this problem by formulating multi-criteria decision making approaches and incorporating the decision theory concepts into multiple-objective optimization schemes to help explore the alternative design trade space [23, 46-48].

Decision based design techniques can be used to help with objective and structured decision making processes using decision-theoretic interpretations of risk and uncertainty management in the context of design [49, 50]. However, these methods have not seen acceptance in the early stages of conceptual design, largely due to the black-box nature of the analysis, where the models of connections between functions and components are not made explicit. Of particular interest to this research, it is impossible to analyze failure propagation paths using DBD methods, making the analysis of potential functional losses difficult in the early design stages.

2.4 Review of Function-Based Analysis Methods

To address the need to evaluate the potential of failures and the resulting functional losses during conceptual design, a body of work has emerged that use function-based approaches to bridge the gap between risk analysis and conceptual design. The emphasis of this body of work is on the use of functional descriptions to describe early concepts, leading to various function-based analysis approaches. Functional design or function based modeling is a natural language for expressing designs at the early stages. Early risk analyses start from these function-based descriptions to provide insight on risk of functional failures. In this subsection, we provide a brief chronological summary of the evolution of these approaches.

The first effort in the identification of failure modes during conceptual design was made possible through the function-failure design method (FFDM), developed by Tumer and Stone [51-53]. This method used a functional model for a system in

combination with historical failure information to map the functionality of a system to potential failure modes. A standard taxonomy to describe functionality, namely the Functional Basis [54], was used to model systems and components at the highest (functional) level, with the intent of providing generic and reusable templates for spacecraft. The method then collected failure data from historical databases and expert elicitation, and mapped these failures onto function, hence building a knowledge base relating failure modes directly to functionality, bypassing the need to know the details of the design form or solutions. FFDM was successfully applied to the Bell 206 rotorcraft and spacecraft systems [21, 55, 56]. Inspired by the FFDM method, Hutcheson et al. [57-59] later sought to enable the design of health monitoring modules concurrently with system conceptual design in order to reveal, model, and eliminate associated risks and failures. These successful applications led NASA to sponsor a research team to conduct functional failure analysis for extending these methods to the design of prognostic and health management systems for the shuttle replacement Crew Launch Vehicle. To add a means of quantifying the risk via FFDM, Grantham-Lough et al. [60] developed the Risk in Early Design (RED) method that formulated a functional-failure likelihood and consequence based risk assessment. This approach classified high-risk to low-risk function-failure combinations to provide designers with a tool that can be used to qualitatively rank/order functional failures and their consequences during conceptual design.

A necessary extension of these matrix-based approaches was to enable decision-making based on risk during the conceptual design of space systems. A risk based decision making method was developed Mehr and Tumer [1] to address this need, namely, the Risk and Uncertainty Based Integrated and Concurrent (RUBIC) design methodology, fueled by the need to assess the risk of integrating prognostic and health management capabilities in large aerospace systems, at the system design stage. In this work, risk was defined by a triplet of fault type, fault probability, and fault consequential cost, and used to determine optimal resource allocation for the detailed design phase; however, risk mitigation attributed to the prognostic and health management capabilities was not explicitly quantified in this formulation. An extension to RUBIC was introduced by the same authors in [2] to enable a cost-benefit analysis (CBA) of integrating new technologies to large complex systems. The CBA framework provided an optimization framework for the allocation and cost justification during functional design, based on a formulation using probabilistic reliability metrics such as system availability, cost of detection, etc.

Last year, the FFIP method was introduced by Kurtoglu and Tumer [3] to significantly extend on the work started with these methods, with a number of additional, novel features: 1) enabling the computation of component interactions that are likely to result in functional failures; 2) allowing the

identification of not only the functional failures but also their propagation paths that are derived from the functional and structural topology of a system; 3) enabling use with a variety of systems without being constrained by a database of documented, historical failure data. In FFIP, we combined decision-making and automated reasoning driven by functional failure analysis. RUBIC, for example, only identified the most critical functions to allocate resources to and was not geared towards the analysis of design decisions regarding system configuration/redundancy. FFDM, and RED were primarily driven by historical data and did not necessarily capture the propagation aspects of functional failures.

The FFIP-based *failure-informed decision making framework*, introduced in this paper, does both simulation-based analysis of functional failure propagation, and association of that analysis with the criticality of functional losses to guide specific design decisions regarding system configuration. Note that, only redundancy decisions are targeted in this paper, however, the method is applicable to other design decisions governing the configuration of a system. FFIP presents a conceptual design tool that enables robust and reliable system design and development of complex systems during the stages of design where only functionality and basic (generic) configuration information is available.

3 SUMMARY: THE FFIP ANALYSIS FRAMEWORK

In prior work, we introduced the Functional Failure Identification and Propagation (FFIP) framework [3], as a significant addition to our prior function-based failure analysis work [1, 2, 51-53, 58]. FFIP brought a much-needed formalism to effectively analyze the effect of the combination of function and configuration on the failure propagation and resulting functional losses. There are three major modules to the proposed FFIP analysis framework: the system model, the behavioral simulation, and the functional-failure logic (FFL) reasoner [3]. This section presents these three modules, summarized from Kurtoglu and Tumer [3].

3.1 System Modeling

As the first module of the FFIP framework, we represent system function, configuration, and behavior by an interrelated array of graph-based, elemental component models. The graph-based modeling approach provides a coherent, consistent, and formal schema to capture function-configuration-behavior architecture of a system at an abstract level.

Functional Representation:

System *function* is represented using function structures [61-63], establishing a formal *function-based design paradigm* based on the concept of *functional modeling* [52, 54, 63, 64]. Functions and flows are represented as verbs and nouns respectively (e.g., transfer gas, mix liquid, open gate, display warning, record data, etc.). The flows are broken down into

three categories: energy, material, and signal. The Functional Basis (FB) taxonomy, with its hierarchical set of flows and functions [52-54], and the functional modeling processes proposed in the literature [52, 53, 62, 64] are used to develop the functional models for the systems under study.

Configurational Representation:

The *structure*, on the other hand, is captured using configuration flow graphs (CFGs) [65]. A CFG strictly follows the functional topology of a system and maps the desired functionality into the component configuration domain. In a CFG, nodes of the graph represent system components, whereas arcs represent energy, material or signal flows between them. For flow naming, the Functional Basis terminology is adopted, while the components of the graph are named using a taxonomy of standard components [66]. The component types in a CFG can be thought of as generic abstractions of common component concepts.

Note that, the construction of a functional model (FM) and the corresponding configuration flow graph (CFG) captures a direct mapping between the functional and the structural architecture of a system. Each mapping represents a transformation that shows how a functional requirement was addressed in the actual design by the use of a specific component concept. To extract these mappings in a consistent manner, the flow information, i.e., the fact that the two graphs share the same flow types, is utilized. Accordingly, by following the “flow paths”, one can define boundaries that isolate the mapping between functional nodes of a function structure and component nodes of a configuration flow graph [3].

Behavioral Representation:

The behavior of the system is represented using a component-oriented modeling approach. The approach involves the development of high-level, qualitative behavior models of system components in various discrete nominal and faulty modes. The transitions between these discrete modes are defined by mode transition diagrams. The component behavior in each mode is derived from input-output relations and underlying first principles. These modular, reusable component behavior models follow the form of configuration flow graphs. Accordingly, state variables critical to the system behavior are incorporated into the representation by associating them with their respective (CFG) flows [3].

3.2 Behavioral Simulation

As the second module of the FFIP framework, we present a simulator that determines the system behavior under certain specified conditions. These conditions are represented by the occurrence of events that cause specific component mode transitions. During the simulation, both the discrete component modes and the set of system state variables need to be tracked.

Accordingly, the overall system state $X(t)$ at time t is described by: $X(t) = \Phi(c(t), v(t))$ (Eqn. 1), where,

$c(t) = [c_1, c_2, c_3, \dots, c_N]$ is a vector of discrete component modes where each component $c = 1, \dots, N$ assumes a discrete mode from its own set of M modes $c_i = (c_{i1}, c_{i2}, c_{i3}, \dots, c_{iM})$, and, $v(t) = [v_1, v_2, v_3, \dots, v_K]$ is a vector of system state variables.

During conceptual design, the system state variables are not known quantitatively. Therefore, these continuous variables are discretized into a set of qualitative values. The vector $v(t)$ then defines these qualitative values for each state variable v_i from a set of P possible values $v_i = (v_{i1}, v_{i2}, v_{i3}, \dots, v_{iP})$. For example, a liquid flow rate variable may take on values from the set of {zero, low, nominal, high}. Similarly, a control signal variable may have values of {nosignal, on, off}, etc.

To start the simulation, the modes of individual components (nodes) in the CFG are initialized along with the values of system state variables associated with input flows (arcs). Then, the state of the system is simulated by solving the continuous-time system in the intervals between discrete events. Each time step propagates values of certain state variables depending on the mode of components, the behavioral models in that particular mode, and the defined component constraint relations. When an event occurs, the continuous-time simulation is stopped, and the corresponding component mode transition is executed. Using this scheme, critical events, consequences of which are investigated, can be inserted into the simulation at any time step. Following this approach, the simulation may be run over a certain number of time steps, or until the system reaches a prescribed end state [3].

3.3 Reasoning via the Function-Failure Logic

The last module of the FFIP framework uses a function-failure logic (FFL) reasoner to determine the state of each system function (i.e., whether it is operational, degraded, or lost) at any time t given the state of the system $X(t) = \Phi(c(t), v(t))$. The simulation feeds the state of the system to the FFL reasoner at the end of each time step and the state of each system function is evaluated at these discrete points. The FFL reasoner translates the dynamics of the system into functional failure identifiers and facilitates the assessment of potential functional failures and resulting fault propagation paths.

Note that, FFL allows the assessment of the operability of a function to be made based on the values of the input and output state variables of the CFG that corresponds to the component by which the function is realized. Therefore, capturing the mapping between the functional model (function) and the configuration

flow graph (behavior) is fundamental to the employment of the function failure logic. The reasoner uses a set of form-independent system function models that describe conditions under which functions deviate from their intended operation [3].

4 RISK-INFORMED DECISION MAKING USING THE FFIP FRAMEWORK

In this paper, the FFIP framework is extended to enable decision-making during early system design based on functional failure potential. This section describes the basic steps of making design decisions using the proposed method and the associated functional risk analysis. This new approach is aimed to offer two immediate advantages:

- It accounts for individual risks in a system derived from basic functional elements as well as the combined risk in a system resulting from the propagation of functional failures,
- Using the approach, the level of risk mitigation based on specific design decisions can be determined by computing a decisions' direct effect on functional failures and the impact on the overall system safety.

The basis for these analyses is a four-step process, which is explained next.

4.1 Step 1: Estimate the “Criticality” of Each System Function

The objective of the first step is to estimate how critical each system function is for the system’s operation. These criticalities are determined by conducting a “Functional FMEA” analysis. Functional FMEA is similar to a traditional FMEA [4], however, it is conducted at a functional level as opposed to the component level that requires detailed component information, which is unavailable during conceptual design.

Figure 1 shows an example of a Functional FMEA Table performed on an Electrical Power System (EPS) of an exploration vehicle [67]. The first column of the table in Figure 1 lists each functional element in the system and the second column depicts the component addressing the listed functionality. For example, the component “solid-state relay” addresses the function “actuate electrical energy”, whereas; the component “wire” provides “transfer electrical energy” functionality in the system. Note that, each of the functional elements may correspond to one or more physical components in the system. Similarly, a single component may address one or more functional requirements. As described in Section 2, building the system functional model and the corresponding configuration flow graph enables one to capture the mapping between the functional and configurational architecture of the system at an abstract, conceptual level. Capturing this mapping is critical for accurately reasoning about a system’s potential faults since it allows the designers and analysts to establish the

relationship between functionality, components, and the system configuration.

Listed next in Figure 1 are failure modes of components, which capture how components may fail in the system. For example, a solid-state relay may fail to open or close, or may overheat. The cause, the effect, and the criticality of a failure in each mode may potentially be different and are therefore defined separately. These are listed in the fourth, fifth, and sixth columns.

A 1-4 ordinal scale is used to assess the criticality of each failure mode in the system. The definition of these criticality ratings are summarized below in Table 1:

Table 1: Criticality Definitions used for Functional Failure Modes and Effects Analysis

Criticalities	
4	Potential for immediate physical harm
3	Potential dangerous mode
2	Reduced mission performance
1	Common failures

4.2 Step 2: Determine the “Functional Risk Factor (FRF)” of Each System Function

The objective of the second step is to estimate the distribution of risk over functional elements of a system using the functional FMEA analysis. Accordingly, the *Functional Risk Factor (FRF)* for each system sub-function is determined by comparing the criticality of individual system functions and by converting the criticality ratings into a coefficient that is normalized based on the combined criticality of all system functions. For simplicity, the probability of each functional failure mode is assumed to be the same. For example using the Functional FMEA Table of Figure 1, the likelihood of ‘Actuate Electrical Energy’ function failing under one of the four failure modes listed is assumed to be equal to 1/4=0.250. If necessary, these probabilities can be estimated using the FFDM method described in Section 2.3, or by using an appropriate historical database.

The Functional Risk Factor (FRF) for each system sub-function is then calculated in two steps:

1) Calculate the cumulative criticality of each sub-function using:

$$\text{Cumulative criticality of function } j, CC_j = \sum p_i \times Cr_i$$

where,

p_i is the probability of failure mode i

Cr_i is the criticality assigned to failure mode i

Summed over the failure modes of function j

Functional FMEA
Sub-system Name:

Electrical Power System (EPS)

Function	Component	Failure Mode	Cause	Effect on system	Criticality
Actuate Electrical Energy	<i>Solid-State Relay</i>	Actuator Fail open	Multiple	Loss of current in branch	2
		Actuator Fail close	Fault in coil side	Loss of ability to break circuit	2
		Overheating	Overcurrent	Eventual loss of relay	2
Regulate Electrical Energy	<i>AC/DC Charger</i>	Uncontrolled state	Noise, improper wiring, damage to coil side	Loss of control or current	2
		Trip/fuse blown	Overcurrent	Loss of charging current	1
		Overheating	Overcurrent/prolonged charging	Damage to charger	2
		Isolation failure	Damage/internal shorts	120VAC at output	4
Import Electrical Energy	<i>Wall Outlet</i>	Breakdown	Multiple	Loss of charging current	2
		No current	Unplugged	Loss of charging	1
Transmit Electrical Energy	<i>Wire</i>	Rupture	Circuit fails open	Loss of current in branch	2
		Current fluctuations	Damage	Fluctuations in current through branch	2
		Short	Circuit is shorted	Current surge, damage, overheating	3

Figure 1: An excerpt of a Functional FMEA Table conducted on an Electrical Power Supply (EPS) sub-system.

Using this formula, the cumulative criticality of the system sub-functions shown in Figure 1 become,

Actuate Electrical Energy:	2.00
Regulate Electrical Energy:	2.25
Import Electrical Energy:	1.00
Transfer Electrical Energy:	2.33

2) Convert the cumulative criticality of each function into a functional risk factor using:

$$\text{Functional risk factor of function } j, \text{FRF}_j = \text{CC}_j / \sum \text{CC}_{1..j}$$

where,

CC_j is the cumulative criticality of function j

Using this formula, the functional risk factors of the system sub-functions for the same example become,

Actuate Electrical Energy:	0.264
Regulate Electrical Energy:	0.297
Import Electrical Energy:	0.132
Transfer Electrical Energy:	0.307

Basically, the FRF is calculated by normalizing the cumulative criticality ratings such that the sum of all functional risk factors equals to unity as shown by the electrical power system example.

The individual FRF ratings constitute the relative weight of each system function based on overall system risk and provide an expected distribution of risk over functional elements. In

other words, the higher the FRF of a function, the more valuable is maintaining that functionality during system operations. For example, loosing the ‘regulate electrical energy’ function carries higher risk (0.297) then losing the ‘import electrical energy’ function (0.132) because when lost, its impact on the system is far more critical.

4.3 Step 3: Determining the “Functional Failure Impact (FFI)” of a Critical Scenario using the FFIP

The objective of third step is to quantify the overall impact of functional failures and their propagation on system functionality. The basis of this step is to calculate the “consequential cost” of functional failures under a critical scenario using the FFIP framework described in Section 3. This is accomplished by:

1) Modeling a system of interest using the FFIP framework:

Any complex electro-mechanical system can be modeled using the FFIP framework, which represents system function, configuration, and behavior, by an interrelated array of graph-based models. This graph-based system model constitutes an environment where knowledge about system function, behavior, and control is integrated and used to automatically predict functional failures [3].

2) Selecting a scenario of interest:

These scenarios are determined based on the concept of operations of a particular system. The FFIP framework is developed to analyze the consequences of critical what-if scenarios in a system governed by the occurrence of specific

component failures [3]. For example, for the EPS system, the FFIP can help answer questions like “How does functional failures propagate if a wire shorts in the system?” or “What is the impact of a AC/DC charger breakdown on overall system functionality?”

3) Running the FFIP analysis using the system model and the selected scenario:

The task of the FFIP framework is to estimate potential functional failures and their propagation under critical event scenarios using behavioral simulation. This is accomplished through a reasoner that translates changes in the system behavior into an assessment of the operability of system functions. Accordingly, system functions are classified as ‘operating’, ‘degraded’, or ‘lost’ [3]. Using the simulation scheme of the FFIP, functions that can potentially be lost under the selected scenario can be computed.

4) Calculating the “Functional Failure Impact (FFI)” of the selected scenario:

Finally, after the FFIP analysis is run, the Functional Failure Impact of the selected scenario can be calculated by simply summing over the “Functional Risk Factors (FRF)” of all functions that are classified as “lost” during the simulation. Naturally, the estimated loss of functions with higher risk factors will result in higher functional failure impact for the system (i.e. losing the ‘regulate electrical energy’ function has a higher impact on the overall functionality of the system when compared to ‘import electrical energy’ function.)

As stated earlier, this process allows the system designers and risk analysts to quantify the overall impact of functional failures and their propagation on the functional operability of the system. This quantification of the functional failure impact is crucial for design decision-making as it constitutes a formal, mathematical basis for exploring design decisions relevant to risk management in general and for risk mitigation in particular. The way in which these design decisions are explored is summarized next in Step 4.

4.4 Step 4: Determining the “Reduction in Risk (RIR)” for each Design Decision

The proposed design methodology is based on the assumption that one can reduce the severity of consequences of failures by making design decisions to mitigate risks associated with certain functional elements in the system. This can be done, for example, by placing more sensors in a sub-system, designing more redundancy, changing the architecture of the sub-system by the addition or removal of certain components, or by introducing new technologies.

The objective of this fourth step is to quantify the level of mitigation a designer can achieve by making such design decisions. This is accomplished by first calculating the consequential cost of functional failures for a *modified design*

under the same critical scenario used in Step 3. Accordingly, the “Functional Failure Impact of the modified design (FFIm)” is computed by making the necessary modeling changes, and by running the FFIP analysis under the same scenario for the modified design. Finally, one can calculate the “Reduction in Risk (RIR)” expressed in percentage by using:

$$\text{Reduction in Risk, RIR} = (\text{FFIm}-\text{FFI})/\text{FFI}$$

The RIR value formally quantifies the amount in risk reduction based on a specific design decision. The RIR value can be used to determine proper decisions to most efficiently mitigate risks associated with functional elements in a design. Moreover, it allows system designers to assess system safety beginning from very early stages of design, and to explore various conceptual design alternatives guided by safety and reliability requirements. The next section demonstrates this by the application of the proposed approach to the design of a hydraulic system.

5 A CASE STUDY: THE DESIGN OF A RESERVOIR SYSTEM

Figure 2 shows the functional model of a design example that will be used in the remainder of this paper to demonstrate the application of the proposed approach. This design problem involves a hold-up tank example, which is used to regulate the liquid amount in an open tank. The hydraulic system consists of seven components: a tank, two valves, two pipes, a controller, and a level sensor. One of the valves, the outlet valve, is manually controlled by an operator. The inlet valve, on the other hand, is actuated through a controller based on sensor measurements from the level sensor installed on the tank. The flow rate of both valves is assumed to be the same. To simplify the analysis it is also assumed that the liquid supply to the system is uninterrupted. Moreover, control laws are designed such that the controller shuts off the inlet valve if the liquid level reaches an overflow threshold to prevent a potentially hazardous tank overflow. Similarly, the operator is expected to shut off the outlet valve if the liquid level reduces below a hazardous dry-out threshold. Figure 2 also depicts the schematic and the configuration flow graph of the hold-up tank.

As is shown on the system configuration flow graph of Figure 2, there are ten state variables (attached to the arcs of the CFG) of the hold-up tank design. Also, the seven components of the system have a total of thirteen distinct modes: six nominal, and seven fault modes, which are detailed in [3]. Note that we will reuse the two critical scenarios from last year’s paper. These two scenarios involve malfunctioning of critical system components as well as an operator error. In the first scenario, the effects of a clogged pipe and a valve failure are investigated, whereas in the second scenario, the consequences of a sensor failure and an operator error are examined.

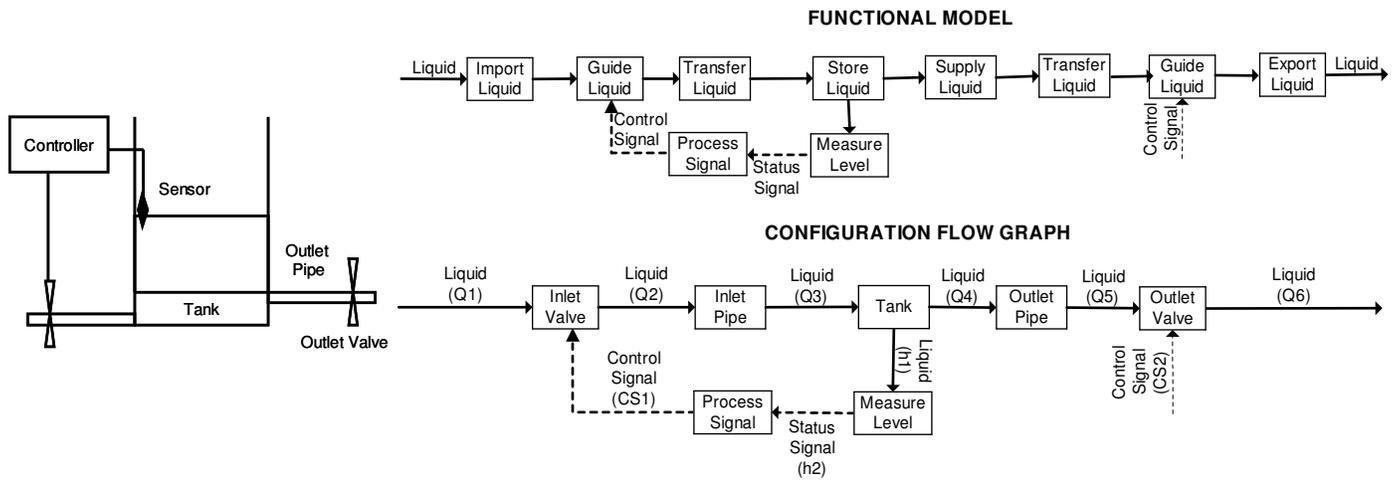


Figure 2: A high-level functional model and a configuration flow graph of a hold-up tank example at some point during its conceptual design. The schematic of the system is shown on the left. The tank system is used to regulate the liquid amount in an open reservoir.

For illustration purposes, we will analyze two modified designs representing different selections for the location of necessary safeguards, and the level of system redundancy. The schematics corresponding to these modified designs are shown in Figure 3. In the first modified design, a decision has been made to add a redundant level sensor to the reservoir system. In the second modified design, system designers have decided to employ a second outlet valve in a series configuration. The

analysis of these two design decisions for determining how well they mitigate functional risks in the system is described next.

5 RESULTS

Before we present the results of the simulations on the modified designs, we first establish the functional failure impact of the two scenarios on the baseline (i.e. original) reservoir design shown in Figure 2.

By following the steps outlined in Section 4, the functional risk factors of the ten system functions are calculated to be:

Import Liquid:	0.043
Guide Liquid:	0.086
Transfer Liquid:	0.109
Store Liquid:	0.131
Supply Liquid:	0.131
Transfer Liquid:	0.109
Guide Liquid:	0.086
Export Liquid:	0.043
Measure Level:	0.131
Process Signal:	0.131

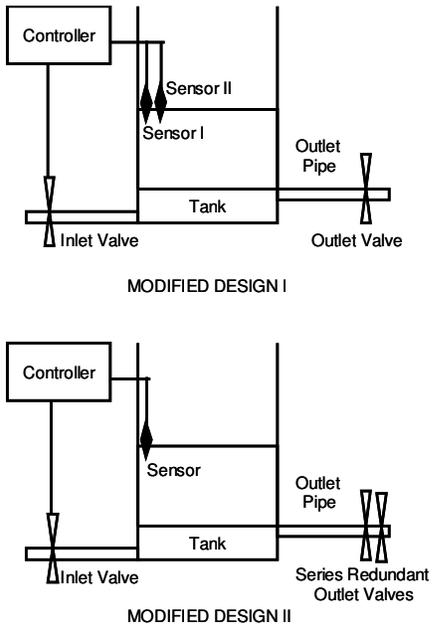
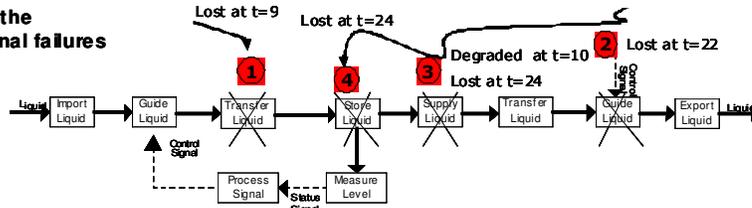


Figure 3: Schematics for the two modified designs. In modified design I, a redundant sensor is added to the reservoir system. In modified design II, a secondary outlet valve is employed in a series configuration.

We, then, determine the functional failure impact of the two critical scenarios. In the first scenario (detailed in [3]), two events are considered: the inlet pipe getting clogged, and the outlet pipe failing to close. These events are injected to the simulation in the specified order. According to this scenario, the system starts out as working nominally until the inlet pipe fails “clogged”. The system responds to this component failure through the designed control laws and the outlet valve is closed as a precautionary measure to prevent a potential dry-out. Later in the scenario, the outlet pipe fails to “close”. This causes the tank level to drop further resulting in a tank dry-out. The first

Pictorial illustration of the propagation of functional failures

SCENARIO I



SCENARIO II

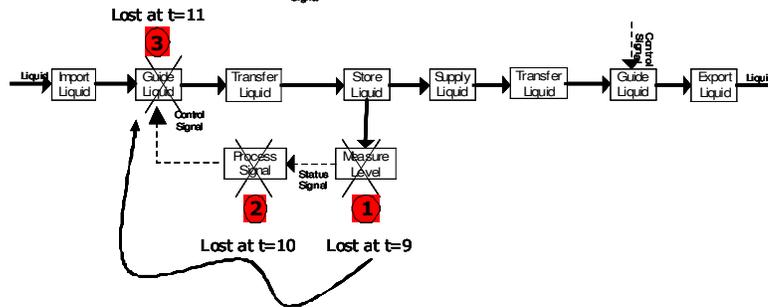


Figure 4: A pictorial illustration of functional failure propagation for both scenarios on the original design.

functional loss occurs for the “transfer liquid” function. After the valve failure, the “guide liquid” function is lost. Finally, the “supply” and “store liquid” functions are lost as a result of the dry-out. These estimates and the time-step of failures are illustrated in Figure 4.

In the second scenario (again described [3]), the system has the same initial conditions, however, a different set of critical events is considered: a sensor failure and an operator error. In this simulation, the system is fully functional until the level sensor fails. At this point, the system is working with the inlet valve under “nominal on” mode, and the absence of an “on” control signal to the valve does not have an immediate negative effect on the system. However, later, the operator mistakenly shuts off the outlet valve. This causes the liquid level to rise, eventually requiring an off signal to be issued for the inlet valve to prevent a potential tank overflow. However, since the sensor is burst, the rise of the liquid level cannot be detected and therefore the inlet valve cannot be shut off. The liquid level continues to rise and the tank overflows. In this scenario, the first functional loss occurs for the “measure level” function. Immediately following this, the “process signal” and the “guide liquid” functions are lost.

Using these results, the functional failure impact of each scenario is calculated to be:

FFI – Scenario I:	0.457
FFI – Scenario II:	0.348

After the functional failure impact of the scenarios on the baseline design is established, we can determine the reduction in risk (RIR) for the two modified designs. To accomplish this, the FFIP is run for the same set of scenarios on both modified designs and the results are tabulated in Figure 5.

For the first modified design, the decision to add a redundant sensor has no effect on the estimated functional failures under the first scenario, and hence the reduction in risk (RIR) for this scenario is zero. However, the addition of the redundant sensor proves to be extremely effective in mitigating the functional failure impact under the second scenario. In this scenario, the second sensor allows the ‘measure level’ function to remain operational even after the initial sensor failure. Moreover, it allows the system to issue a command to close the inlet valve. As a result, a potential tank overflow is safeguarded against without a single loss of functionality (only redundancy is lost in the system). Hence the reduction in risk (RIR) for the second scenario is a 100%.

For the second modified design, the design decision to employ a series redundant outlet valve has quite the opposite effect under the two scenarios. This time, the decision does not help to mitigate any functional risk under the second scenario. However, it improves the system safety under the first scenario. According to the simulation, the inlet valve fails ‘clogged’ and the ‘transfer liquid’ function fails just as it failed for the original design. After this, one of the outlet valves fails to close. The decision to switch to a series redundant outlet valve configuration pays off after this event. The existence of a redundant outlet valve allows the operator to shut-off the second outlet valve after the first one fails to close. As a result, further drop of the liquid level and a potential system dry-out is prevented. At the end, the ‘transfer liquid’ function remains to be the only functionality lost in the system. The RIR for this scenario becomes 76.149%.

7 SUMMARY AND CONCLUSIONS

In this paper, we introduced a new risk-informed decision-making methodology that can be used during early design of complex systems. The proposed approach is based on the notion

Design Version	Scenario	FFIP Analysis and Risk Estimates		
		Functions Estimated to 'Fail'	Functional Failure Impact (FFI)	Reduction in Risk (RIR in %)
Baseline Design				
Design Decision	N/A			
	Scenario I	Transfer Liquid Store Liquid Supply Liquid Guide Liquid	0.457	N/A
	Scenario II	Guide Liquid Process Signal Measure Level	0.348	N/A
Modified Design I				
Design Decision	redundant sensor			
	Scenario I	Transfer Liquid Store Liquid Supply Liquid Guide Liquid	0.457	0.000
	Scenario II	None	0.000	100.000
Modified Design II				
Design Decision	series outlet valve config			
	Scenario I	Transfer Liquid	0.109	76.149
	Scenario II	Guide Liquid Process Signal Measure Level	0.348	0.000

Figure 5: Tabulated results of the analysis of the two design decisions. The RIR percentages are used to express how well design decisions mitigate risks under critical scenarios.

that a failure happens when a functional element in the system does not perform its intended task. Accordingly, risk is defined depending on the role of functionality in accomplishing designed tasks.

A simulation-based failure analysis tool is used to analyze functional failures and their impact on overall system functionality. The analysis results are then integrated into a decision-making framework that relates the impact of functional failures and their propagation to decision making in order to guide system level design decisions. With the help of the proposed methodology, a multitude of failure scenarios can be quickly analyzed to determine the effects of decisions on overall system risk. Using this decision-making approach, design teams can systematically explore risks and vulnerabilities during early, functional stage of system development prior to the selection of specific components. Thus, the proposed method offers opportunities for significant reduction in cost, and increase in system safety and reliability by enabling early development of preventive measures that can effectively and efficiently guard against system failures.

There are several unique characteristics of the develop framework. First, it provides an analytical approach to quantify individual risk of basic functional elements in a system as well as the combined risk resulting from the propagation of functional failures. More importantly, this quantification is derived from system specific function-to-configuration relations integrating the knowledge of which components are to-be used in the system for addressing functional requirements. This is a significant extension to the existing function-based failure assessment techniques in the literature. Second, the developed framework provides the designers with a means to determine the level of risk mitigation based on specific design decisions. This is accomplished by computing a decision's direct effect on

functional failures and the impact on the overall system safety. As a result, it allows designers to make decisions about what components to use in the system, how to configure them, the types and locations of necessary safeguards, and the proper level of system redundancy, all guided by potential functional failures and their impact on overall system performance as determined by reliability and safety requirements.

There are also several assumptions that the presented method is based upon. These assumptions pose certain limitations that are left to be addressed in the future research. For example, only design decisions targeting system redundancy are tackled in this initial implementation. Such decisions allow same failure scenarios to be run on the original and modified designs. If, however, more complex design decisions are made governing the addition or removal of a huge number of system components, or the introduction of new technologies, the resulting configuration changes may force a failure scenario to be obsolete for the modified design. Secondly, the current implementation does not account for the likelihood of different failure scenarios. In the reservoir design, for example, it is inherently assumed that both scenarios have the same prior probability of occurrence, which may or may not be the case. Incorporating scenario probabilities will help the designers to more accurately assess the impact of their decisions by allowing them to analyze the combined effect of decisions under all potential scenarios. Third, the sequence of events to simulate is chosen by the designer. Unavoidably, a designer may miss certain sequence of events that could lead to failures. Exploring the event sequence space automatically for comprehensive coverage of potential failure scenarios is an open area of research left for future studies.

8 REFERENCES

- [1] Mehr, A.F. and Tumer, I.Y., 2006, "Risk based decision making for managing resources during the design of complex aerospace systems," *ASME Journal of Mechanical Design*, **128**(4): 1014-1022.
- [2] Hoyle, C., Mehr, A.F., Tumer, I.Y., and Chen, W., 2007, "Cost-benefit analysis of ISHM in aerospace systems," *International Design Engineering Technical Conferences; Computers in Engineering Conference (IDETC/CIE)*, **Accepted**, Las Vegas, NV.
- [3] Kurtoglu, T. and Tumer, I.Y., 2007, "A graph based fault identification and propagation framework for functional design of complex systems," *ASME Journal of Mechanical Design*, (**In print.**)
- [4] DoD, D.o.D. Procedures for performing failure mode, effects, and criticality analysis.
- [5] Vesely, W.E., Goldberg, F.F., Roberts, N.H., and Haasi, D.F. (1981), *The Fault Tree Handbook*, US Nuclear Regulatory Commission.

- [6] Greenfield, M.A., 2000, "NASA's Use of Quantitative Risk Assessment for Safety Upgrades," *IAAA Symposium*, Rio de Janeiro, Brazil.
- [7] Stamatelatos, M. and Apostolakis, G. (2002), Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners v1.1, NASA, Safety and Mission Assurance.
- [8] Mosleh, A., Groen, F., Hu, Y., Zhu, D., Najad, H., and Piers, T. (2004), Simulation-Based Probabilistic Risk Analysis Report, Center for Risk and Reliability, University of Maryland.
- [9] Giarratano, J.C. and Riley, G.D., 2004, *Expert Systems: Principles and Programming*, 4th, Boston, MA, PWS Publishing Company.
- [10] deKleer, J. and Williams, B.C., 1987, "Diagnosing multiple faults," *AI*, **32**: 97-130.
- [11] Kurien, J. and Nayak, P., 2000, "Back to the Future with Consistency-based Trajectory Tracking," *AAAI*.
- [12] Williams, B.C. and Nayak, P.P., 1996, "A Model-based Approach to Reactive Self-Configuring Systems," *AAAI*.
- [13] Dvorak, D. and Kuipers, B.J., 1989, "Model Based Monitoring of Dynamic Systems," *IJCAI*.
- [14] Chen, J. and Patton, R.J., 1998, *Robust Model-Based Fault Diagnosis for Dynamic Systems*, Kluwer Academic Publishers.
- [15] Sacks, I.J., 1985, "Digraph Matrix Analysis," *IEEE Transactions on Reliability*, **R-34**(5): 437-446.
- [16] Abdelwahed, S., Karsai, G., and Biswas, G. (2003), System Diagnosis using Hybrid Failure Propagation Graphs, Vanderbilt University.
- [17] Kapadia, R., 2003, "SymCure: A Model-Based Approach for Fault Management with Causal Directed Graphs," *IEA/AIE 2003*, **LNAI 2718**.
- [18] Deb, S., Pattipati, K.R., Raghavan, V., Shakeri, M., and Shrestha, R. (1995). *Multisignal flow graphs: a novel approach for system testability analysis and fault diagnosis*. *IEEE Aerospace and Electronics Systems Magazine*, **10**: 14-25.
- [19] Berenji, H., Ametha, J., and Vengerov, D., 2003, "Inductive Learning For Fault Diagnosis," *12th IEEE International Conference on Fuzzy Systems*.
- [20] Yairi, T., Kato, Y., and Hori, K., 2001, "Fault Detection by Mining Association Rules from House-keeping Data," *SAIRAS*.
- [21] Tumer, I.Y., Stone, R.B., and Roberts, R.A., 2003, "Analysis of JPL's Problem and Failure Reporting Database," Submitted to *ASME Design Engineering Technical Conference, Design for Manufacturing Conference*, Chicago, IL.
- [22] Swaminathan, S. and Smidts, C.S., 1999, "Framework for assessing confidence in simulation-based design under uncertainty," *Annual Reliability and Maintainability Symposium*.
- [23] Li, H. and Azarm, S., 2002, "An approach to product line design selection under uncertainty and with competitive advantage," *Journal of Mechanical Design*, **122**(4): 411-418.
- [24] Grote, G., 2004, "Uncertainty management at the core of system design," *Annual Reviews in Control*, **28**(2): 267-274.
- [25] Thunnissen, D.P., 2004, "Method for determining margins in conceptual designs," *Journal of spacecraft and rockets*, **41**(1): 85-92.
- [26] Martin, J.D. and Simpson, T.W., 2006, "A methodology to manage system level uncertainty during conceptual design," *Journal of Mechanical Design*, **128**: 959-968.
- [27] Aughenbaugh, J.M. and Paredis, C.J., 2006, "The value of using imprecise probabilities in engineering design," *Journal of Mechanical Design*, **128**(July): 969-979.
- [28] Gu, X., Renaud, J.E., and Penniger, C.L., 2006, "Implicit uncertainty propagation for robust collaborative optimization," *Journal of Mechanical Design*, **128**: 1001-1013.
- [29] Padhke, M.S., 1989, *Quality engineering using robust design*, Englewood Cliffs, Prentice Hall PTR.
- [30] Parkinson, A., Sorensen, C., and Pourhassan, N., 1993, "A General Approach for Robust Optimal Design," *Journal of Mechanical Design*, **115**(1): 74-80.
- [31] Melchers, R.E., 1999, *Structural Reliability Analysis and Prediction*, Chichester, England, John Wiley & Sons.
- [32] Du, X., Sudjianto, A., and Huang, B., 2004, "Reliability-Based Design under the Mixture of Random and Interval Variables," *Journal of Mechanical Design*, **127**(6): 1068-1076.
- [33] Youn, B.D. and Choi, K.K., 2004, "Selecting Probabilistic Approaches for Reliability-Based Design Optimization," *AIAA Journal*, **42**(1): 2154-2161.
- [34] Zang, T.A. (2002), Needs and Opportunities for Uncertainty-Based Multidisciplinary Design Methods for Aerospace Vehicles, NASA.

- [35] Hazelrigg, G.A., 1988, "A framework for decision based engineering design," *Journal of Mechanical Design*, **120**: 653-658.
- [36] Ullman, D.G., 2006, *Making Robust Decisions*, Trafford Publishing.
- [37] Lewis, K.E., Chen, W., and Schmidt, L.C., eds. *Decision making in engineering design*, ed. A. Press, 2006.
- [38] Saari, D.G., 2001, *Decisions and elections: explaining the unexpected*, New York, NY, Cambridge University Press.
- [39] Sidall, J.N., 1972, *Analytical decision-making in engineering design*, Englewood Cliffs, NJ, Prentice-Hall.
- [40] Hazelrigg, G.A., 1996, *System Engineering: An approach to information-based design*, Prentice-Hall.
- [41] French, S., 1986, *Decision theory: An introduction to the mathematics of rationality*, London, Wiley.
- [42] Jie, W. and Krishnamurty, S., 2001, "Learning based preference modeling in engineering design decision making," *Journal of Mechanical Design*, **123**(2): 191-198.
- [43] Wassenaar, H.J., Chen, W., Cheng, J., and Sudjianto, A., 2005, "Enhancing discrete choice demand modeling for decision-based design," *Journal of Mechanical Design*, **127**(4): 514-523.
- [44] Marston, M., Allen, J.K., and Mistree, F., 2000, "Decision Support Problem Technique: integrating descriptive and normative approaches in Decision Based Design," *Journal of Engineering Valuation and Cost Analysis*, **2000**(3): 2.
- [45] Allen, B., 2000, "Toolkit for decision-based design theory," *Journal of Engineering Valuation and Cost Analysis*, **3**(2): 85-105.
- [46] Tappeta, R.V. and Renaud, J.E., 1997, "Multiobjective collaborative optimization," *Journal of Mechanical Design*, **119**(3): 403-411.
- [47] Keeney, R.L. and Raiffa, H., 1993, *Decisions with multiple objectives: preferences and value tradeoffs*, New York, NY, Cambridge University Press.
- [48] Thurston, D.L., 1991, "A formal method for subjective design evaluation with multiple attributes," *Research in Engineering Design*, **3**(2).
- [49] Wood, W.H. and Agogino, A.M., 2005, "Decision based conceptual design: Modeling and navigating heterogeneous design spaces," *Journal of Mechanical Design*, **127**(1): 2-11.
- [50] Kalsi, N., Hacker, K., and Lewis, K., 2001, "A Comprehensive Robust Design Approach for Decision Trade-Offs in Complex Systems Design," *Journal of Mechanical Design*, **123**(1): 1-10.
- [51] Stone, R., Tumer, I., and Stock, M., 2005, "Linking Product Functionality to Historic Failures to Improve Failure Analysis in Design," *Research in Engineering Design*, **16**(2): 96-108.
- [52] Stone, R., Tumer, I.Y., and Van Wie, M., 2004, "The Function Failure Design Method," *Journal of Mechanical Design*, **127**(3): 397-407.
- [53] Tumer, I.Y. and Stone, R.B., 2003, "Mapping Function to Failure During High-Risk Component Development," *Research in Engineering Design*, **14**(1): 25-33.
- [54] Hirtz, J., Stone, R., McAdams, D., Szykman, S., and Wood, K., 2002, "A Functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts," *Research in Engineering Design*, **13**(2): 65-82.
- [55] Uder, S.J., Stone, R.B., and Tumer, I.Y., 2004, "Failure Analysis in Subsystem Design for Space Missions," *ASME Design Engineering Technical Conferences, Design Theory and Methodology*, **DETC2004/DTM-57338**, Salt Lake City, Utah.
- [56] Tumer, I.Y., Stone, R.B., and Bell, D.G., 2003, "Requirements for a Failure Mode Taxonomy for Use in Conceptual Design," *International Conference on Engineering Design*, Stockholm Sweden.
- [57] Hutcheson, R. and Tumer, I.Y., 2005, "Function based co-design paradigm for robust health management," *International workshop on structural health management*, Palo Alto, CA.
- [58] Hutcheson, R., McAdams, D., Stone, R., and Tumer, I., 2006, "A Function-Based Methodology for Analyzing Critical Events," *Proceedings of IDETC/CIE 2006 DETC2006-99535*, Philadelphia, PA.
- [59] Hutcheson, R. and Tumer, I.Y. (2005). *Function based design of a spacecraft power subsystem diagnostics testbed*. International Mechanical Engineering Congress and Exposition, Orlando, FL.
- [60] Grantham Lough, K., Stone, R., and Tumer, I., 2006, "The Risk in Early Design (RED) Method: Likelihood and Consequence Formulations," *Proceedings of DETC'06, DETC2006-99375*, Philadelphia, PA.
- [61] Pahl, G. and Beitz, W., 1996, *Engineering Design: A Systematic Approach*, Springer Verlag.

- [62] Otto, K. and Wood, K., 2001, *Product Design: Techniques in Reverse Engineering, Systematic Design, and New Product Development*, New York, Prentice Hall.
- [63] Stone, R. and Wood, K., 2000, "Development of a Functional Basis for Design," *Journal of Mechanical Design*, **122**(4): 359-370.
- [64] Stone, R., Wood, K., and Crawford, R., 2000, "Using Quantitative Functional Models to Develop Product Architectures," *Design Studies*, **21**(3): 239-260.
- [65] Kurtoglu, T., Campbell, M., Gonzalez, J., Bryant, C., Stone, R., and McAdams, D., 2005, "Capturing Empirically Derived Design Knowledge for Creating Conceptual Design Configurations," *Proceedings of IDETC/CIE 2005*, **DETC2005-84405**, Long Beach, CA.
- [66] Kurtoglu, T., Campbell, M., Bryant, C., Stone, R., and McAdams, D., 2005, "Deriving a Component Basis for Computational Functional Synthesis," *International Conference on Engineering Design, ICED'05*, Melbourne, Australia.
- [67] Poll, S., Patterson-Hine, A., Camisa, J., Garcia, D., Hall, D., Lee, C., Mengshoel, O., Neukom, C., Nishikawa, D., Ossenfort, J., Sweet, A., Yentus, S., Roychoudhury, I., Daigle, M., Biswas, G., & Koutsoukos, X. (2007 May). "Advanced Diagnostics and Prognostics Testbed", *18th International Workshop on Principles of Diagnosis*, Nashville, TN.