MAPPING FUNCTION TO FAILURE DURING HIGH-RISK COMPONENT DEVELOPMENT

Irem Y. Tumer, Ph.D. *
Research Scientist
Computational Sciences Division
NASA Ames Research Center
Moffett Field, California 94035
Phone: 650 604 2976
itumer@mail.arc.nasa.gov

Robert B. Stone, Ph.D.
Assistant Professor
Department of Basic Engineering
University of Missouri-Rolla
Rolla, MO 65409-0210
Phone: 573 341 4086
rstone@umr.edu

Abstract

When designing high-risk aerospace systems, it is essential to provide crucial failure information for failure prevention. Failure Modes and Effects types of analyses and prior engineering knowledge and experience are commonly used to to determine the potential modes of failures a product might encounter during its lifetime. When new products are being considered and designed, this knowledge and information is expanded upon to help designers extrapolate based on their similarity with existing products and the potential design tradeoffs. In this work, we aim to enhance this process by providing design-aid tools which derive similarities between functionality and failure modes. Specifically, this paper presents the theoretical foundations of a matrix-based approach to derive similarities that exist between different failure modes, by mapping failures to the functionality of each component, and applies it to a simple design example. The function-failure method is proposed to design new products with solutions for functions that eliminate or reduce the potential of a failure mode.

Keywords: Failure prevention in design; Function-failure similarity; Design for Failure; Design for Reliability; Risk-Based Design.

Background and Objectives

When designing rotating machinery components for high-risk aerospace applications, safety and performance problems become crucial elements. Failures are unacceptable and cost of maintenance is preferably low. As a result, most aerospace systems are implemented with thorough failure monitoring units, which often result in an overwhelming amount of information from which decisions have to be made real-time. In this work, the aim is to eliminate at least some of the potential failure modes early on at the design stage. In particular, methods to understand and predict the potential failure modes are viewed as essential to advancing the field of fault monitoring and failure prevention. With this goal in mind, a novel approach is presented here as a potential design-aid tool which explores the relationship between failure modes and the functionality of components (Tumer and Stone, 2001). The underlying premise of the research is that failure modes ultimately correlate back to the function that a particular component addresses. If the link between

---

*Corresponding Author. Accepted to *Research in Engineering Design.* Conference version published in ASME/DETC 2001.

failure mode and function can be established, then component solutions for each function can be designed to eliminate or significantly reduce a given failure mode.

Failure-Free Design

Feedback of crucial failure information into the design stage is essential in producing high-quality parts that must satisfy stringent performance and safety requirements. Such is the case with high-risk aerospace components. As shown in Figure 1, a typical feedback loop into design must consider all phases where failures and variations can be introduced, including design, manufacturing and assembly, tooling and fixture, and operational considerations. The focus in this work is on those considerations that lead to unacceptable failure modes when these components are placed in operation. This information is commonly gathered from experience and previous designs; their significance is typically re-evaluated for each application. When designing a new product, or modifying existing products for new specifications, it is often up to the designers to assess and draw conclusions about the similarity between different designs, components, and failure modes. To help with this daunting task, this work aims to provide a means to systematically and correctly identify and eliminate potential failure modes, based on the functionality of machinery components.

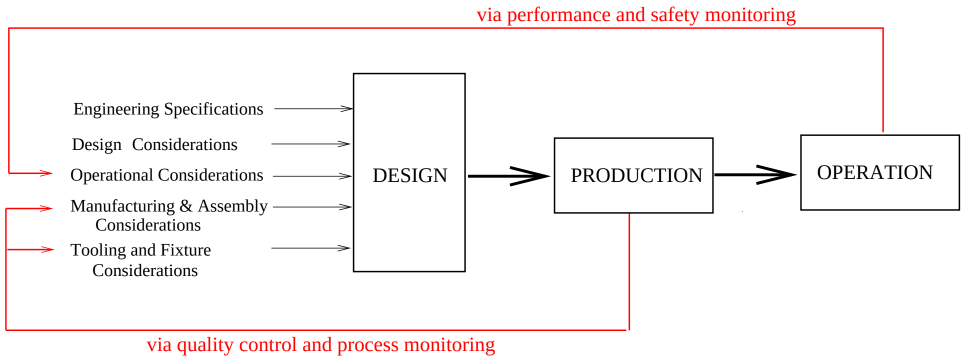

Figure 1: Information Feedback From Design to Operation.

The potential of mechanical failures is a crucial concern in design. Reliability, maintenance, and satisfactory performance of machines and systems depend heavily upon understanding, recognizing, and preventing/eliminating mechanical failures (Collins and Hagan, 1976; Mitchell, 1993; Smith, 1999). Mechanical failures may be defined as any change in size, shape, or material properties of a structure, machine, or machine component that renders it incapable of satisfactorily performing its intended function (Collins, 1993). Success in designing competitive products while preventing premature mechanical failures can be achieved only by recognizing and evaluating all potential failure modes, in the early stages of design. To this end, the designer must be acquainted with an array of failure modes observed in the field, and with the conditions leading to these failures. In this work, failures are defined in terms of a basic set of standard mechanical failure modes that all components will be subject to during their lifetime. To define this vector of failures, failure modes presented in Collins (Collins, 1993) are adopted, summarized in Table 1. All new systems will be mapped to match these standard modes.

To help with feedback from operation and production into design, it is crucial to provide designers and manufacturing engineers with techniques they can use to effectively account for the existing and potential failure modes and mechanisms. At the design and development stages, standard reliability tools are used for a thorough coverage and understanding of all possible and potential failure modes, lengthening the

Table 1: Elemental Failure Modes.

Main Category	Sub	Main Category	Sub
Elastic Deformation	force induced	Impact	fracture
	temperature induced		deformation
Yielding			wear
Brinnelling			fretting
Ductile rupture			fatigue
Brittle fracture		Fretting	fatigue
Fatigue	high-cycle		wear
	low-cycle		corrosion
	thermal	Creep	
	surface	Thermal relaxation	
	impact	Stress rupture	
	corrosion	Thermal shock	
	fretting	Galling and seizure	
Corrosion	direct chemical attack	Spalling	
	galvanic	Radiation damage	
	pitting	Buckling	
	intergranular	Creep buckling	
	selective leaching	Stress corrosion	
	erosion	Corrosion wear	
	cavitation	Corrosion fatigue	
	hydrogen damage	Creep and fatigue	
	biological		
	stress		
Wear	adhesive		
	abrasive		
	corrosive		
	surface fatigue		
	deformation		
	impact		
	fretting		

development time of such components considerably. At the manufacturing stage, quality control techniques are used to inspect components (some at a 100% rate) to assure satisfactory and safe operation, making the manufacturing of such components costly and time-consuming (Carter, 1997; Henley and Kumamoto, 1992; Phadke, 1989). Despite these lengthy and costly steps during production, failures still occur at an unacceptable rate when components are placed in their operational states. The increasing pressure in the aerospace industry to reduce the production and development cycle and increase the lifecycle of crucial aircraft components, while keeping safety the number one priority, requires more stringent steps during the development of high-risk components.

There are several supporting techniques that are often used by designers to account for potential failures (Carter, 1997). Examples (commonly used at NASA) are checklists, FMEA/FMECAs, and FTAs. Checklists are listings of all relevant failure modes and mechanisms. They act as reminders to ensure that the design has been assessed as adequate to meet all possible circumstances. Although often the only source of such information, checklists are typically incomplete and do not provide the complete picture of the mechanisms for failure. A systematic method for drawing up an exhaustive list is lacking from the literature (Carter, 1997). In other words, there is no “algorithm” that enables one to draw up a comprehensive checklist for a specified part. This results in checklists being unreliable design tools.

FMEA (failure modes and effects analysis) and FMECA (failure modes effects and criticality analysis) are tools used to first identify each failure mode at some designated level (e.g., component, sub-assembly, machine), and then trace the effect of the failure through all the higher levels of the hierarchy in turn (Carter, 1997). It is used to establish whether each failure mode has unacceptable consequences on the system as a whole. The problem with this method is that, contrary to what the name implies, FMEA does not tell the designers what to do at the lowest level, if the consequences are unacceptable. While these traditionally-used methods are effective for identifying failure modes related to components, a common complaint is the difficulty in identifying system-wide failure modes (Bowles, 1998; Eubanks et al., 1997; Henning and Paasch, 2000). Traditional FMEA needs a systematic approach capable of capturing a wider range of failure modes, applicable early in the design stage (Eubanks et al., 1997).

FTA (fault tree analysis) performs the reverse of FMEA. It starts with an undesirable top event and isolates possible causes at each successive lower level of the hierarchy in order to establish the prime cause(s). FTA is more powerful in the sense that it forces the designers to consider all the causes of unacceptable top events. However, the analysis is not pursued far enough, and the prime causes are not revealed (Carter, 1997). Although a well-accepted technique, large system-level fault trees are often difficult to understand, and difficult to build due to the complex logic involved (Henley and Kumamoto, 1992). The weakness of both FMEA and FTA is that the basic sources of unacceptable behavior cannot be identified (Carter, 1997).

Functional Modeling in Design

Functional modeling is a key step in the product design process, whether original or redesign. By developing a formal theory of functional modeling, the intent is to push functional modeling into the realm of repeatable, and even computable, engineering analysis. Stone et al. have had substantial success with their functional model derivation and common functional language as demonstrated by inter-institutional experimental results (Stone and Wood, 2000; Stone et al., 2000). In this work, their common functional language will be adopted for defining elemental functions.

All functional modeling begins by formulating the overall product function. By breaking the overall function of the device into small, easily solved sub-functions, the form of the device follows from the assembly of all sub-function solutions. The lack of a precise definition for small, easily solved sub-functions casts doubt on the effectiveness of prescriptive design methodologies (Pahl and Beitz, 1988; Ullman, 1997; Ulrich and Eppinger, 1995) among engineers in more analytical fields. For instance, within a given methodology how does one reconcile different functional models of a product generated by different designers? Typically, such differences arise from semantics or poor identification of product function. The development of a standard set of functions and flows, referred to here as a functional basis, and a systematic approach to functional modeling offer the best case to erase remaining doubt.

Much of the recent work on a functional basis stems from the results of value engineering research that began in the 1940s (Akiyama, 1991; Miles, 1972). Value analysis seeks to express the sub-functions of a product as an action verb-object pair and to assign a fraction of a product's cost to each sub-function. Sub-function costs then direct the design effort (specifically, the goal is to reduce the cost of high value sub-functions). However, there is no standard list of action verbs and objects. Recognizing that a common vocabulary for design was necessary to accurately communicate helicopter failure information, Collins et al. (Collins and Hagan, 1976) develop a list of 105 unique mechanical functions. Here, the mechanical functions are limited to helicopter systems and do not utilize any classification scheme.

Function-based design methodologies have also pushed the development of functional languages in order to provide a clear stopping point in the functional modeling process and a consistent level of detail. Pahl and Beitz (Pahl and Beitz, 1988) list five generally valid functions and three types of flows, but they are at a very high level of abstraction. Hundal (Hundal, 1990) formulates six function classes complete with more specific functions in each class in order to make function-based design computable. Another approach uses the 20 subsystem representations from living systems theory to represent mechanical design functions (Koch et al., 1994). Malmqvist et al. (Malmqvist et al., 1996) compare the Soviet Union era design methodology known as the Theory of Inventive Problem Solving (TIPS) with the Pahl and Beitz methodology. TIPS uses a set of 30 functional descriptions to describe all mechanical design functions (Altshhuller, 1984). Malmqvist et al. note that the detailed vocabulary of TIPS would benefit from a more carefully structured class hierarchy using the Pahl and Beitz functions at the highest level. Kirschman and Fadel (Kirschman and Fadel, 1998) propose four basic mechanical functions groups, but vary from the standard verb-object sub-function description popular with most methodologies. However, this work appears to be the first attempt at creating a common vocabulary of design that leads to common functional models of products.

Building on the above work, the concept of a functional basis is developed by Stone and Wood (Stone and Wood, 2000; Stone et al., 2000) which significantly extends previous research (Little et al., 1997; Otto and Wood, 1997). A functional basis is a standard set of functions and flows capable of describing the mechanical design space. The work expands the set of functions and groups them into eight classes. This initial functional basis subsumes all other classification schemes discussed above along with the 30 basic sub-functions found in TIPS. The standard list of functional descriptions is needed such that the matrices can be shared among different engineers. Summarized in Tables 2 and 3, the functional basis is a vocabulary of function and flow words which may be combined to form a functional description (Stone et al., 2000; Hirtz et al., 2002). A functional description has a verb-object format where the verb is selected from the function list in Table 3, and the object is selected from the flow lists in Tables 2. The function and flow sets are divided into different categorizations, i.e., class, basic, sub-basic (or flow-restricted). Each successive categorization allows greater levels of detail to be captured in the functional description. Typically, the basic level is sufficient to convey the elemental functions at the basic level.

Table 2: Classes, Flow Types, and Complements.

Class	**Secondary**	**Tertiary**	**Correspondents**
Material	Human		Hand, foot, head ,etc.
	Gas		Homogeneous
	Liquid		Incompressible, compressible, homogeneous
	Solid	Object	Rigid-body, elastic body, widget
			Particulate
			Composite
	Plasma		
	Mixture	Gas-gas	
		Liquid-liquid	
		Solid-solid	Aggregate
		Solid-liquid	
		Liquid-gas	
		Solid-gas	
		Solid-liquid-gas	
		Colloidal	Aerosol
Signal	Status	Auditory	Tone, Verbal
		Olfactory	
		Tactile	Temp, Pressure, Roughness
		Taste	
		Visual	Position, Displacement
	Control	Analog	Oscillatory
		Discrete	Binary
Energy	Human		
	Acoustic		
	Biological		
	Chemical		
	Electrical		
	Electromagn.	Optical	
		Solar	
	Hydraulic		
	Magnetic		
	Mechanical	Rotational	
		Translational	
		Vibrational	
	Pneumatic		
	Radioactive		
	Thermal		

Table 3: Function Classes, Basic Functions and Synonyms.

Class	**Secondary**	**Tertiary**	**Correspondents**
Branch	Separate		Isolate, sever, disjoin
		Divide	Detach, isolate, release, sort, split, disconnect, subtract
		Extract	Refine, filter, purify, percolate, strain, clear
		Remove	Cut, Polish, Sand, Drill, Lathe
	Distribute		Diverge, Scatter, Disperse, Diffuse, Empty, Absorb, Dampen, Dispel, Resist, Dissipate
Channel	Import		Input, Receive, Allow, Form Entrance, Capture
	Export		Discharge, Eject, Dispose, Remove
	Transfer	Transport	Lift, Move
		Transmit	Conduct, Convey
	Guide		Direct, shift, switch,Straighten, Steer
	Guide	Translate	Move, relocate
		Rotate	Turn, Spin
		Allow DOF	Constrain, Unlock, unfasten
Connect	Couple		Associate, connect
		Join	Assemble, fasten
		Link	Attach
	Mix		Combine, Blend, Add, Pack, Coalesce
Control	Actuate		Enable, Start, Initiate, Turn on
Magnitude	Regulate		Control, equalize, Limit, maintain
		Increase	Allow, open
		Decrease	Close, delay, interrupt
	Change		Adjust, modulate, clear, demodulate, invert, normalize, rectify, rest, scale, vary, modify
		Increment	Amplify, enhance, magnify, multiply
		Decrement	Attenuate, dampen, reduce
		Shape	Compact, Crush, Compress, Pierce, deform, form
		Condition	Prepare, adapt, treat
	Stop		End, halt, pause, interrrupt, restrain
		Prevent	Disable, turn off
		Inhibit	Shield, insulate, protect, resist
Convert	Convert		Transform, Liquefy, Solidify, Evaporate, Condense, Integrate, Differentiate, Process create, decode, encode, generate, digitize
Provision	Store		Accumulate
		Contain	Capture, enclose
		Collect	Absorb, consume, fill, reserve
	Supply		Provide, Replenish, retrieve
Signal	Sense		Feel, determine
		Detect	Discern, perceive, recognize
		Measure	Identify, locate
	Indicate		Announce, show, denote, record, register
		Track	Mark, time
		Display	Emit, expose, select
	Process		Compare, calculate, check
Support	Stabilize		Steady
	Secure		Attach, Mount, Lock, Fasten, Hold
	Position		Orient, Align, Locate

Current Focus

The goal in this work is to enhance failure prevention in design by incorporating functional modeling information. In this light, tools are sought to make use of known failure modes and the required functionality of the components, across components and systems. It is the authors' view that components have a "commonality" they share at some basic level in terms of their failure modes and functionality. This basic level of commonality is explored in this work by decomposing the knowledge about failures and functionality via matrix manipulations. Once the common modes of failures at the basic levels are determined, a larger family of components/systems can be considered. Using this generalization, this work proposes to formalize the process of feeding failure and reliability information into the design and manufacturing phases (Stone and Wood, 2000; Stone et al., 2000; Tumer and Huff, 2002). In this paper, the initial development of such a function-failure method is presented. The paper first presents the theoretical basis for the proposed method, followed by a detailed demonstration of the mechanics of the method by using a simple example in rotating machinery. Future work will establish this method as a design tool for typical applications for NASA missions, including the domains of rotorcraft transmission and aircraft engine failures.

Function-Failure Method: A Design-Aid Tool

The method proposed in this work is based on work that was presented by Stone et al. (Stone and Wood, 2000; Stone et al., 2000) to derive the similarity between different designs based on functionality, and used to provide a repository for designers; a brief background is presented next. In this paper, the idea of similarity is extended to failure detection for a family of aerospace components and products. The key idea is to prevent failures by means of tradeoff and/or redesign decisions. In the following sections, first a formal definition of the starting and derived matrices is presented. Then, a simple example problem using a rotating machinery simulator model is used in this paper to demonstrate how the method can be applied, including a discussion of the potential uses of the derived results in the early stages of design. Future work is currently in progress to apply this method to the domain of helicopter and aircraft engine failures and functions.

Theoretical Background

A methodology was developed by Stone et al., which provides a means to transform customer need rankings and function structures into quantitative models, offering designers a novel way to archive and communicate product design knowledge (Stone and Wood, 2000; Stone et al., 2000). Specifically, they use matrix manipulations to extract product similarity using a product repository which groups products together based on functionality and customer needs. Scaled customer need rankings are first mapped to sub-functions of the product function structure in the form of a product vector ϕ. An $m \times n$ product-function matrix Φ is then formed to create a product repository to archive product design knowledge. Each element of the product-function matrix, ϕ_{ij} is the cumulative customer need rating for the ith function of the jth product. To compensate for variations due to different sources of information, the product-function matrix is normalized across the entire product space. The normalized product-function matrix $\mathbf{N}$, has elements $\nu_{ij} = \phi_{ij}\frac{\bar{\eta}}{\eta_j}\frac{\mu_j}{\bar{\mu}}$. Here, $\bar{\eta}$ is the average customer need rating, η_j is the customer rating for the jth product, $\mu_j = \sum_{i=1}^{m} H(\phi_{ij})$ is the number of functions in the jth product (H is the Heaviside function), and $\bar{\mu}$ is the average number of functions (n is the number of products and m is the total number of sub-functions for all products.) The product repository can then be manipulated to identify groups of products sharing similar functions and customer needs (product families). Using such a method, a new product's functional model can be used to find similarities so that existing knowledge can guide its development. This is accomplished by computing the product-product matrix using the renormalized matrix $\hat{\mathbf{N}}$ (so that the norm is equal to 1), defined as $\hat{\Lambda} = \hat{\mathbf{N}}^T\hat{\mathbf{N}}$.

Preliminary Definitions

Consider m subsystems and/or components for the application domain under study (e.g., helicopters, airplanes, space station, mars rover, etc.). Let $\mathbf{F}$ be an $n \times 1$ vector of failures commonly found in that application domain. Let $\mathbf{E}$ be the $r \times 1$ vector containing all elemental functions for the components under study. To represent failure information, such individual vectors (containing information on failure modes and functionality) are transformed into a matrix of information. To begin, consider failure information that is typically recorded with respect to components or subsystems. This information can be arranged succinctly using a failure vector $\mathbf{F}$ with elements indicating the failure modes that can occur for the components. The n failure modes are aggregated together to form $\mathbf{CF}$, the $m \times n$ component-failure matrix, where n is the total number of failure modes occuring across all m components. In addition to the binary information of failure for a given component, likelihood or frequency of occurrence data can be encoded in $\mathbf{CF}$ as well. For instance, if multiple failures are observed for a component, their frequency can be entered in the matrix instead of simple binary data.

Similarly, components can be described in terms of their functionality. Here, an elemental function vector $\mathbf{E}$ is constructed for each component with elements that indicate the functionality of the component. Aggregating each vector of r functions, together for the m components (represented in the columns), creates the $r \times m$ function-component matrix $\mathbf{EC}$, where r is the total number of functions necessary to describe all of the m components. The function-component matrix is closely related to the product-function matrix Φ, reviewed above, though this time functionality of components rather than that of the entire product, is considered. Thus, the $\mathbf{EC}$ matrix may be constructed as a binary matrix with a 1 indicating the component solves a certain function and a 0 indicating the opposite, or the elements of $\mathbf{EC}$ may be weighted to include additional information. Examples include customer need importance correlated to functions (as in the $\mathbf{F}$ matrix reviewed above) or manufacturing cost associated with each component.

Function-Failure Relationship

Although component-failure and function-component matrices can be formed automatically using the knowlegde at hand, finding a correlation between functionality and potential failure modes is a non-trivial task. Intuition and prior experience can possibly be used in a similar way, but the risk of making an intuitive error is often too high to accept. In this work, we propose to use the more easily obtained information, described in terms of the $\mathbf{CF}$ and $\mathbf{EC}$ matrices, to derive the function-failure correlation. Once the component-failure and function-component matrices are computed, the relationship between function and failure can be computed as: $\mathbf{EF} = \mathbf{EC} \times \mathbf{CF}$. This $r \times n$ matrix, called the function-failure matrix, relates the failure modes to the elemental functions. Each element ij indicates whether any component solving function i has ever failed by failure mode j. This information is useful when designing or redesigning components, offering failure modes to guard against during the design phase. For example, a new design or redesign of an existing component might proceed as follows. A component's functional model is specified as a vector. That vector is multiplied by the function-failure matrix, $\mathbf{EF}$, to produce a component-failure mode vector. This vector then indicates potential failure modes the component could experience and the likelihood of occurrence for

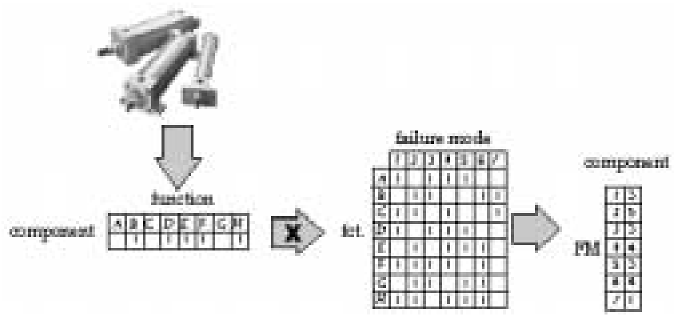

Figure 2: Using a Functional Model to Identify Potential Failure Modes.

Table 4: Component-Failure Matrix Example (**CF**).

	F1	**F2**	**F3**	**F4**	**F5**
C1 : *gear*	1	1	0	1	1
C2 : *bearing*	1	0	1	1	0
C3 : *shaft*	0	1	0	0	1

each failure mode (the larger the failure mode value, the more likely). The designer is then able to design out the identified failure modes during the conceptual design stage. This approach is shown schematically in Figure 2.

Application: Rotating Machinery Example

Consider the design of a simple rotating machinery system, consisting of a shaft attached to a motor by means of a coupling, supported by two sets of ball bearings, which drives a gearbox via two belts, which in turn drives a load. This machinery system will serve as a preliminary test bed to demonstrate how the function-failure matrix can work. More realistic applications are currently being attacked, starting with helicopters. In the case of a helicopter, the load would be equivalent to driving the rotor blades with an epicyclic transmission gearbox. The input to the transmission would be equivalent to a shaft, supported by bearings, and driven by the helicopter engine (Huff et al., 2002).

For this simple example, three types of components are considered: namely, the shaft, gears, and bearings. These components can be subject to elementary failure modes, described in Table 1, that need to be considered at the early design stages. Selecting a subset from these failure modes, these components are assumed to be subject to wear, fatigue, corrosion, fretting, and impact failure modes. Table 4 presents an aggregated matrix of failures and components, with 1's representing an occurrence of a failure for a given component, and 0's representing non-occurrence. The failure modes are labeled as follows: F_1 is wear, F_2 is fatigue, F_3 is corrosion, F_4 is fretting, and F_5 is impact. The components are labeled as follows: C_1 is a gear, C_2 is a bearing, and C_3 is the shaft. The failure modes represent the variables (columns) and the components represent the various observations (rows).

The basic functional descriptions are found using the functional basis of Tables 2 and 3. The function

Table 5: Function-Component Matrix Example (**EC**).

	C1 : *gear*	**C2** : *bearing*	**C3** : *shaft*
E1 : *change m.e.*	1	0	0
E2 : *guide m.e.*	1	0	1
E3 : *transfer m.e.*	1	0	1
E4 : *position m.e.*	0	1	0
E5 : *stabilize m.e.*	0	1	0

vectors for each component are aggregated together to form the function-component matrix **EC** (with $r = 5$ and $m = 3$) shown in Table 5. Once again, the components under consideration are the gear, C_1, bearing, C_2, and shaft, C_3. The elemental functions these components have to satisfy are selected as E_1: change mechanical energy, E_2: guide mechanical energy, E_3: transfer mechanical energy, E_4: position mechanical energy, and, E_5: stabilize mechanical energy (see Table 3 for basic function definitions.)

Capturing Similarity Information for Design and Redesign

The matrices described above represent convenient ways to mathematically capture failure mode and function data for components. Additional useful design information may be obtained through matrix manipulations of the data. The resulting similarity matrices (equivalent to covariance matrices from above) provide tools for designers to assess and design against the impact of potential failure modes.

Similarity matrices can be derived in several ways, depending on the purpose of the designer. For example, taking the transpose of the function-component matrix and post multiplying it by function-component matrix yields an $m \times m$ symmetric component-component matrix. Mathematically, the component-function similarity matrix is given by: $\hat{\Lambda}_{EC} = \overline{\mathbf{EC}}^T \times \overline{\mathbf{EC}}$, where $\overline{\mathbf{EC}}$ is the normalized function-component matrix with each column normalized to unity for convenience. Each element ij of the component-function matrix indicates the similarity between component i and component j based on elemental functions. That is, if component i is functionally similar to component j, then element λ_{ij} will have a value in $(0,1]$. Components that are completely similar with themselves have a similarity value of 1 due to the normalization of the function-component matrix. Likewise, components that share no functions in common will have a similarity value of 0. Similar derivations can be achieved using the remaining matrices, as demonstrated below.

Using the **CF** and **EC** matrices from above, the function-failure matrix can be computed as $\mathbf{EF} = \mathbf{EC} \times \mathbf{CF}$, which gives:

$$\mathbf{EF} = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 \\ 1 & 2 & 0 & 1 & 2 \\ 1 & 2 & 0 & 1 & 2 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \end{bmatrix},$$

where the rows represent the elemental functions E_i and the columns represent the failure modes F_j. Analyzing the function-failure matrix, one sees that function pairs *guide m.e.* & *transfer m.e.* and *position m.e.* & *stabilize m.e.* experience the same failure modes. Also, the failure modes *fatigue* and *impact* occur more frequently for the functions *guide m.e.* and *transfer m.e.*. Though this is a limited example, the function-failure data can be used to identify traditionally occurring failure modes when only a component's function is known and use that knowledge to design out the potential failures.

Additional design observations can be made by computing the similarity matrices. First, the component-function similarity $\hat{\Lambda}_{EC}$ is calculated from the function-component matrix after normalizing each column to unity as follows:

$$\overline{\mathbf{EC}} = \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ \frac{\sqrt{3}}{3} & 0 & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{3}}{3} & 0 & \frac{\sqrt{2}}{2} \\ 0 & \frac{\sqrt{2}}{2} & 0 \\ 0 & \frac{\sqrt{2}}{2} & 0 \end{bmatrix},$$

and,

$$\hat{\Lambda}_{EC} = \overline{\mathbf{EC}}^T \times \overline{\mathbf{EC}} = \begin{bmatrix} 1.000 & 0.000 & 0.816 \\ 0.000 & 1.000 & 0.000 \\ 0.816 & 0.000 & 1.000 \end{bmatrix}.$$

The component-function similarity matrix identifies that components 1 and 3 (i.e., the gear and the shaft) are similar in function (in terms of failure modes) when one is projected onto the other. This indicates that the gear could possibly be used as a replacement for the shaft (or vice versa) and that solution principles used in the gear could be used in a redesign of the shaft (again, the converse is also true).

Next, the component-failure similarity matrix is calculated from the component-failure matrix (non-normalized) as:

$$\Lambda_{CF} = \mathbf{CF} \times \mathbf{CF}^T = \begin{bmatrix} 4 & 2 & 2 \\ 2 & 3 & 0 \\ 2 & 0 & 2 \end{bmatrix}.$$

Note that the diagonal simply returns the count of failure modes each component experiences when **CF** is a binary matrix. Component 1 (the gear, from looking at column 1 or row 1) shares two failure modes in common with each of the other components, while components 2 and 3 (bearing and shaft) have no common failure modes (as indicated by the zeros in the off-diagonals). Consider components 1 and 3 which are functionally similar (with a similarity index of 0.816) and share two failure modes in common, as seen from the component-failure matrix. If a design solution for one component is found that eliminates the common failure modes, then that solution will most likely be applicable to the remaining components as well.

Finally, the failure-component similarity matrix is calculated as:

$$\Lambda_{FC} = \mathbf{CF}^T \times \mathbf{CF} = \begin{bmatrix} 2 & 1 & 1 & 2 & 1 \\ 1 & 2 & 0 & 1 & 2 \\ 1 & 0 & 1 & 1 & 0 \\ 2 & 1 & 1 & 2 & 1 \\ 1 & 2 & 0 & 1 & 2 \end{bmatrix}.$$

For this set of components and recorded failures, the failure modes F1-F4 (wear and fretting) and F2-F5 (fatigue and impact) tend to occur on the same component most frequently. Other combinations of failure modes are possible, but not as likely. Failure modes F2-F3 (fatigue and corrosion) and F3-F5 (corrosion and impact) have no incidence of occurring on the same component.

Conclusions and Future Work

In this paper, a function-failure method was introduced to take advantage of the link between failure modes and functionality of components. The method is meant to provide designers with an analytical means to make systematic tradeoff and design/redesign decisions based on similarities, to avoid potential failure modes. A crucial piece of the work is the inherent link between functionality and failure modes.

The matrices presented in this paper can be used to derive various types of information for designers. For example, the component-function similarity matrix provides designers with a tool to identify possible replacement components that solve similar functions. It also provides a way to search and rank component solutions that are similar in function and use design by analogy techniques to embody a design. One possible use for the component-function similarity matrices is to identify component solutions that prevent certain failure modes. If, between functionally-similar components A and B (as determined by $\hat{\Lambda}_{EC}$), component B does not experience all of the same failure modes as component A (as determined by Λ_{CF}), then there is some characteristic of component B that could be incorporated into A to improve its performance.

As another example, premultiplying the component-failure matrix by its transpose yields a symmetric matrix with elements indicating failure mode combinations which occur across components. A high value in element ij of the failure-component similarity matrix indicates that failure modes i and j affect many components jointly. Mathematically, the matrix is formed by: $\Lambda_{FC} = \mathbf{CF}^T \times \mathbf{CF}$. The failure-component similarity matrix (Λ_{FC}) yields insight into possible interactions of two or more failure modes, with elements indicating failure mode combinations which occur across components. It can be used to direct component remedies that will eliminate more than one failure mode. In terms of current FMEA and FTA techniques, knowledge of failure modes that often occur interactively would give designers a more complete list of possible product failures to investigate.

Finally, the relationships between component functionality and failure modes are revealed by analyzing the function-failure matrix **EF**. This information, not readily available to the designer, is obtained by multiplying the function-component matrix (derived from engineering specifications and schematics) with the component-failure matrix (derived from accident reports, maintenance guides, etc.) The real advantage of the **EF** matrix as a design tool is the early (i.e., at the conceptual design phase) identification of potential failure modes that commonly occur in components solving the known function. With only a functional description of a product, designers can identify the type of analysis required to embody a component solution.

The function-failure method is applied here to a simple example using a rotating machinery test rig, to illustrate its potential. The purpose of developing such analytical methods is to meet the tight performance and safety requirements imposed on designers for critical NASA applications. As an ongoing collaborative project between NASA Ames and The University of Missouri-Rolla, the function-failure method will be applied to a more realistic example using rotorcraft transmission and aircraft engine failure data and design specifications (Huff et al., 2002). This will involve a thorough analysis of actual failures collected from accident data reports (Harris et al., 2000). A mapping of the assigned functions onto the basic set of functions presented in this work has begun. This mapping, accompanied by the standard failure modes described in Table 1, will be used to start analyzing the failure data. Such an analysis is essential in establishing the function-failure method presented in this paper as a viable and useful design-aid tool.

References

Akiyama, K. (1991). *Function Analysis: Systematic Improvement of Quality Performance*. Productivity Press.

Altshhuller, G. (1984). *Creativity as an Exact Science*. Gordon and Branch Publishers.

Bowles, J. (1998). The new SAE FMECA standard. In *Proceedings of the annual reliability and maintainability symposium*.

Carter, A. (1997). *Mechanical Reliability and Design*. John Wiley & Sons.

Collins, J. (1993). *Failure of materials in mechanical design: analysis, prediction, prevention*. Wiley Interscience.

Collins, J. and Hagan, B. (1976). The failure-experience matrix: a useful design tool. *ASME Journal of Engineering for Industry*.

Eubanks, C., Kmenta, S., and Ishii, K. (1997). Advanced failure modes and effects analysis using behavior modeling. In *ASME Design Engineering Technical Conferences*, Sacramento, CA.

Harris, F. D., Kasper, E. F., and Iseler, L. E. (2000). U.S. civil rotorcraft accidents: 1963 through 1997. In *NASA Technical Report*, volume NASA/TM-2000-209597.

Henley, E. and Kumamoto, H. (1992). *Probabilistic Risk Assessment: Reliability Engineering, Design, and Analysis*. IEEE Press.

Henning, S. and Paasch, R. (2000). Diagnostic analysis for mechanical systems. In *ASME Design Theory and Methodology Conference*, volume DETC2000/DTM-14580, Baltimore, MD.

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*, Accepted for publication.

Huff, E., Tumer, I., Barszcz, E., Dzwonczyk, M., and McNames, J. (2002). Analysis of maneuvering effects on transmission vibrations in an AH-1 Cobra helicopter. *Journal of the American Helicopter Society*.

Hundal, M. (1990). A systematic method for developing function structures, solutions and concept variants. *Mechanism and Machine Theory*, 25(3):243–256.

Kirschman, C. and Fadel, G. (1998). Classifying functions for mechanical design. *Journal of Mechanical Design, Transactions of the ASME*, 120(3):475–482.

Koch, P., Peplinski, J., llen, J., and Mistree, F. (1994). A method for design using available assets: Identifying a feasible system configuration. *Behavioral Science*, 30:229–250.

Little, A., Wood, K., and McAdams, D. (1997). Functional analysis: A fundamental empirical study for reverse engineering, benchmarking and redesign. In *ASME Design Engineering Technical Conferences*, volume DETC97/DTM-3879, Sacramento, CA.

Malmqvist, J., Axelsson, R., and Johansson, M. (1996). A comparative analysis of the theory of inventive problem solving and the systematic approach of pahl and beitz. In *ASME Design Engineering Technical Conferences*, volume DETC96/DTM-1529, Irvine, CA.

Miles, L. (1972). *Techniques of Value Analysis Engineering*. McGraw-Hill.

Mitchell, J. S. (1993). *Introduction to machinery analysis and monitoring*. PennWell Books.

Otto, K. and Wood, K. (1997). Conceptual and configuration design of products and assemblies. *ASM Handbook, Materials Selection and Design, ASM International*, 20.

Pahl, G. and Beitz, W. (1988). *Engineering Design: A Systematic Approach*. Springer-Verlag.

Phadke, M. S. (1989). *Quality Engineering Using Robust Design*. Prentice Hall PTR, Englewood Cliffs, New Jersey.

Smith, J. D. (1999). *Gear Noise and Vibration*. Marcel Dekker.

Stone, R. and Wood, K. (2000). Development of a functional basis for design. *Journal of Mechanical Design*, 122(4):359–370.

Stone, R., Wood, K., and Crawford, R. (2000). Using quantitative functional models to develop product architectures. *Design Studies*, 21(3):239–260.

Tumer, I. and Huff, E. (2002). On the effects of production and maintenance variations on machinery performance. *Journal of Quality in Maintenance Engineering*, Accepted for publication.

Tumer, I. and Stone, R. (2001). Analytical method for mapping function to failure during high-risk component development. In *ASME Design for Manufacturing Conference*, volume DETC2001/DFM-21173, Pittsburgh, PA.

Ullman, D. (1997). *The Mechanical Design Process*. McGraw-Hill.

Ulrich, K. and Eppinger, S. (1995). *Product Design and Development*. McGraw-Hill.