INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'07
28 - 31 AUGUST 2007, CITÉ DES SCIENCES ET DE L'INDUSTRIE, PARIS, FRANCE

A MODELLING SCHEME FOR CAPTURING AND ANALYZING MULTI-DOMAIN DESIGN INFORMATION: A HAIR DRYER DESIGN EXAMPLE

Gregory M. Mocko[1], Joshua D. Summers[1], Georges M. Fadel[1], Sudhakar Teegavarapu[1], Jonathan R. A. Maier[1], and Thulasiram Ezhilan[1]
[1]Department of Mechanical Engineering, Clemson University, Clemson, SC 29631-0921

ABSTRACT
A matrix-based modelling method and system analyses are developed for describing design information associated with complex products including: requirements, functions, components, and engineering characteristics. The matrices provide a quantitative means for modelling the relationships between the four information domains. Additionally, matrices enable basic mathematical functions and matrix multiplication to be utilized to analyse systems. The modelling method is demonstrated in the design and analysis of a consumer hair dryer. The hair dryer is chosen because it is sufficiently complex and has been used as a demonstration/validation example by several other researchers. The modelling method fills the gaps of existing matrix-based approaches by enabling adjacent information domains to be described, relationships between these information domains to be captured, and analyses of the system to be performed using mathematical functions. Additionally, the modelling method provides a linkage for existing matrix-based methods including Design Structure Matrices (DSM) and the House of Quality (HoQ) matrices.

Keywords: House of Quality, Matrix-based, Design Information, Functional Modelling

1 INTRODUCTION
The design problems and decisions encountered in the early stages of design deal with information, including requirements, functions, components, and engineering characteristics that capture the performance measures of the system. As such, several design tools have been developed for structuring this conceptual design information using matrices. However, these existing tools do not provide algorithms for evaluating this conceptual design information. For example, the requirements list and function structures proposed by Pahl and Beitz [1] provide a means for organizing conceptual design information, but do not provide a means for capturing the relationships between requirements and functions or algorithms for analyzing the requirements and functions. Conversely, the House of Quality (HoQ) provides a mathematical means for modelling the relationships between customer requirements and engineering attributes and analysis algorithms. However, the HoQ and information flow proposed by Pahl and Beitz are inconsistent. For example, the HoQ models the relationships between customer requirements and engineering attributes in a single matrix. In the Pahl and Beitz method, customer requirements are mapped to functions, functions are mapped to components, and components are mapped to engineering attributes.
In this paper a matrix-based modelling scheme and analysis algorithms for evaluating information in the conceptual phase of design are presented. The modelling scheme provides a quantitative approach that links design information including requirements lists, function structures/trees, component and assembly hierarchies, and leverages matrix-based design tools such as the HoQ and the design structure matrix (DSM), and complements best-practice systematic design methods. System analyses methods are developed to identify potential areas of design improvement in terms of requirements, functionality, and components. The modelling scheme is illustrated through engineering and analysis of a consumer hair dryer. In the following sections, the modelling scheme is introduced and the demonstration example is discussed.

2 LITERATURE REVIEW

Developing tools, methods, and frameworks for modelling complex engineering systems is important to ensure that engineering designers understand the complex interactions between components and assemblies, the aggregate and individual functionality of the system, the allocation of requirements across the system architecture, and the verification of requirements in the context of system behaviour to name a few. As such, a number of researchers have developed tools for modelling the complex information and relationships in engineering systems. An ever-present consideration in developing support tools is balancing the level of detail required for modelling with the design insight gained through usage of the tool. Thus, several matrix-based methods have been developed to enable complex systems to be concisely represented. Johansson and Krus [2] argue that matrix-based modelling techniques provide an efficient way for displaying and interpreting relationships. Ghoniem and co-authors [3] conclude that matrix representations enable information and relationships to be quickly visualized over graph-based approaches. Finally, it is shown by Steward [4] that matrix-based modelling approaches of engineering activities provide a means for both modelling and analyzing complex development processes.

The House of Quality (HoQ) is a widely used graphical tool for capturing the relationships between customer requirements and engineering attributes. The HoQ is used to capture the relationships between customer requirements and quantitative measurable parameters that represent the customer needs. The HoQ is used to concisely describe the product specifications, engineering requirements, benchmarks, target values for the product, and their relationships. The customer requirements are captured as rows in the HoQ and can be modelled in a hierarchical representation. The engineering characteristics are captured as columns. The relationship matrix is where the mappings between customer requirements and engineering characteristics are modelled. Several different types of mapping schemes (i.e., binary (1/0); 9-3-1) may be used to describe the customer attribute-engineering measurable relationships. The House of Quality is an attention directing tool that enables design teams to focus attention on particular aspects or trouble areas within a product. For example, by examining the rows and columns of the HoQ, it is easy to determine how customer requirements are or are not being satisfied. Conversely, the engineering characteristics of the product can be rank-ordered. The HoQ is an effective tool for modelling the complex interactions between end-users (customers) and engineering designers [5-7].

The Design Structure Matrix (DSM) is a matrix-based tool for modelling and analyzing complex engineering systems. It has been applied in many different engineering and management domains for a variety of applications. The DSM was originally developed to manage the sequence of design activities in product development processes [4]. Eppinger [8] and Ulrich and Eppinger [9] have used the DSM to visualize and analyze relationships between design activities, components, and component parameters. Complex engineering systems are often represented using the DSM as a directed graph consisting of binary relationships between information elements. Cells in a DSM matrix are populated with ones (1's) indicating a relationship or with zeroes (0's) indicating no relationship between the corresponding row and column. The quantitative binary modelling scheme enables mathematical functions to be exploited for analyzing and organizing information in the DSM. The DSM is limited to a single information domain and is represented with a square, symmetric matrix.

Axiomatic design is a matrix-based design methodology that enables designers to systematically transform customer needs to functional requirements, functional requirements to design parameters, and design parameters to process variables. The Axiomatic Design process includes a set of activities and arrays and matrices for modelling the information in the domains and for creating the mappings between the information domains. Several different types of systems analysis can be performed. Horizontal decomposition is enabled through the mappings between information domains. Hierarchical decomposition is enabled by decomposing high-level functions into specific component-related functions. Coupled decomposition and system analysis is generated through a zigzagging process. Zigzagging is completed to iterate between the functional requirement and design variable domains until a "proper" decomposition results. Additionally, hybrid decomposition can be determined by using the horizontal and hierarchical structures. While not required, square matrices are encouraged to enable functionally decoupled design of minimum information content [10].

Matrix-based modelling techniques have been developed for evaluating and analyzing failure modes and failure diagnosis [11-13]. Arunajadai and co-authors [11] model the failure modes and corresponding components which are then manipulated using clustering algorithms to determine critical components. For example, in [12, 13] system components, failure modes, and their interactions are modelled using matrices and several analysis algorithms are used to compute the diagnosability of the system, components, and failure modes. While the failure modes and diagnosis application domain is not directly related to the domains modelled in this research, the matrix manipulation methods provide a means for analyzing complex systems. These methods provide the basis for the analysis completed in the research reported in this paper.

Leung and co-authors [14] utilize matrices to determine the work share analysis of product development processes. A customer requirement-to-measurable engineering metric mapping matrix, component-to-engineering metric mapping matrix, and a component-to-development location mapping matrix are created and multiplied using basic matrix mathematics to determine how the product development tasks are decomposed and allocated to different design locations. Existing matrices from the HoQ, the physical decomposition of mechanical systems into constitutive assemblies and components, and work allocation and distribution knowledge are used to create a model of the product and process to determine the risk associated with distributive and collaborative product development process.

As discussed there are many different matrix-based modelling tools and methods for structuring, organizing, and analyzing complex engineering design information. The DSM provides a generic set of tools and algorithms for analyzing tasks, component grouping, and information in design with a single information domain. Axiomatic design is a structured means for capturing similar information to the proposed method from functional requirements to process variables. However, a key limitation of axiomatic design is the focus on functionally decoupled designs. The HoQ enables designers to capture customer requirements to engineering characteristics. The modelling method and corresponding matrices proposed in this research provide several advantages over existing approaches and fill the following gaps: 1) modelling of adjacent information domains, 2) propagation and tracking of information across domains using, 3) information domains are based on well-accepted systematic design process, and 4) analysis methods for multi-domain design information.

3 MATRIX BASED MODELLING METHOD

3.1 Model Matrices

The proposed matrix-based modelling methodology is presented in Figure 1. The modelling scheme enables designers to capture customer requirements, functions, components, and engineering characteristics. Additionally, the method and matrices provide a mechanism for representing assembly models and component hierarchies, and functional hierarchies in a series of inter-related matrices. Additionally, the matrices provide a basis for analyzing the system using several analysis algorithms.

As illustrated in Figure 1, the information domains and the flow of information captured in the methodology complement commonly accepted systematic design methodologies [1]. The matrix-based modelling scheme consists of four information domains and three relationship matrices. Specifically, the modelling method is composed of a systematic design process, and three primary matrices (Level 0) that are populated by designers. Additionally, the primary matrices can be manipulated and multiplied for system design and analysis. The resulting matrices exist at Level 1 and Level 2. As illustrated in Figure 4, the computed matrix at Level 2 is similar to the HoQ.

3.2 Step 1. Requirement to function modelling

The first step in the modelling method is to identify the customer requirements, functions, and the corresponding mapping relationships. The requirements, functions, and mappings are represented in the requirements-to-functions (R-F) matrix. The R-F matrix uses a binary scale to describe existence or non-existence of a relationship between individual requirements and functions. Binary mappings between adjacent domains simplify the modelling process by requiring that designers identify if a relationship exists or not. Thus, designers do not need to model the strength of relationships between domains. The R-F matrix is an easy to visualize representation of customer requirements, functions and their relationships. The R-F matrix enables functional requirements and non-functional requirements to be identified through the existence relationships in the matrix.

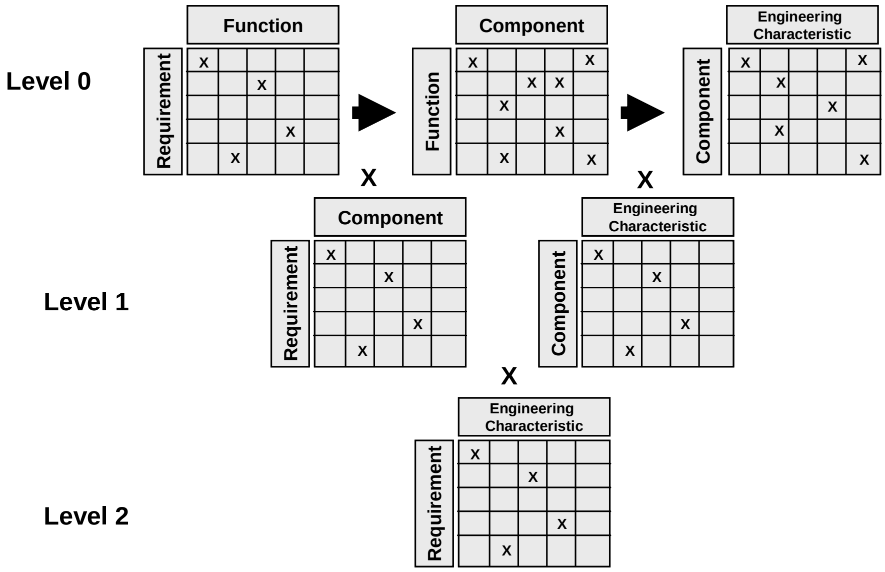

Figure 1 – Matrix-based modelling approach

3.3 Step 2. Function to component modelling

The second step in the method is the identification of components and how each component embodies a function of the system. The functions, identified in the R-F matrix from Step 1, the components, and the mapping relationships are captured in the function-to-component (F-C) matrix. The F-C matrix also uses a binary scale to describe if a component embodies or does not embody a function (an existence relationship). The binary representation enables designers to identify if a component embodies a function, but does not enable the "strength" of the mapping to be represented. The F-C matrix permits easy visualization of the complex interactions between components and their functionality. The matrix supports basic analysis of the system including high and low functionality components, non-functional components, and functionally coupled components.

Similar to the R-F matrix, the information in the F-C matrix must be captured at the same level of abstraction. It is natural to identify the functionality of individual components instead of identifying the aggregate functionality of assemblies and sub-assemblies in the system. With the exception of the information domains captured, the R-F matrix is populated in an identical manner to the R-F matrix.

3.4 Step 3. Component to engineering characteristic model

The third step in the methodology is the identification of the engineering characteristics. The engineering characteristics represent the behaviour characteristics of the system as a whole and each component individually. The assembly-component structure, from Step 2, engineering characteristics, and relationship mappings between individual components and engineering characteristics are captured in the component-to-engineering characteristic (C-EC) matrix. Like the previous two matrices, the C-EC uses a binary scale to describe the performance measures associated with a particular component (again, existence relationships). The C-EC matrix provides a summary of how the performance of the system is related to the components in the system. The engineering characteristics represented in the C-EC matrix are identical to the engineering characteristics in the HoQ. However, unlike the HoQ the engineering characteristics are dependent on the components in the system. Thus, the set of engineering characteristics represented in the C-EC matrix varies based on the system components.

3.5 Step 4. System analysis

Steps 1 through 3 describe the requirements, functions, components, and engineering characteristics of the system. These matrices are referred to as Level 0 in Figure 1. Designers can complete several analyses at this level to understand the system. However, additional analyses and deeper insight of the system are possible through matrix manipulation of the Level 0 matrices, resulting in the Level 1 and Level 2 matrices in Figure 1. While it is possible to multiply all adjacent matrices, the resulting matrices do not always results in value-added analysis of the system. It was determined through several examples that the following matrices add significant value to the system analysis. First, the R-

F and F-C matrices are multiplied together to form the requirements-to-components (R-C) matrix (Equation 1). The R-C matrix provides a means for understanding what requirements are related to functional components. Next, the R-F and F-C matrices are multiplied with the C-EC matrix determine how the requirements are related to the engineering characteristics (see Equation 2).

$$[\mathbf{R\text{-}C}]=[\mathbf{R}-\mathbf{F}]\times[\mathbf{F}-\mathbf{C}] \tag{1}$$

$$[\mathbf{R\text{-}EC}]=[\mathbf{R}-\mathbf{F}]\times[\mathbf{F}-\mathbf{C}]\times[\mathbf{C}-\mathbf{EC}] \tag{2}$$

In addition to multiplying matrices across information domains, individual matrices are multiplied by hierarchical representations of the system. For example, the requirement-to-assembly matrix is computed by multiplying the Equation 4 by the assembly-component hierarchy (see Equation 3).

$$[\mathbf{R\text{-}A}]=[\mathbf{R}-\mathbf{C}]\times[\mathbf{C}-\mathbf{A}] \tag{3}$$

Similarly, the logical grouping of components into functional groups is computed through Equation 4.

$$[\mathbf{F\text{-}A}]=[\mathbf{F}-\mathbf{C}]\times[\mathbf{C}-\mathbf{A}] \tag{4}$$

The analysis completed on the matrices uses simple mathematical functions including summation of rows and column and sorting. The methods provide insightful observations about system requirements, functionality, and components by focusing attention on important requirements, functions, and components.

4 EXAMPLE: MODELING AND ANALYSIS OF A HAIR DRYER

4.1 Overview

The matrix-based modelling scheme and analysis methods are applied to the engineering and analysis of a consumer hair dryer. The study illustrates the shortcomings and usage of the modelling method for existing consumer products. The design of a hair dryer is chosen because it is sufficiently complex to demonstrate the value of the modelling scheme. Moreover, the hair dryer design example is chosen because it has been used in previous research to demonstrate similar conceptual design tools and matrix-based modelling schemes. The hair dryer system model is developed through reverse engineering and from existing literature [6, 11, 14, 16-18].

4.2 Requirements Modelling

The first step for analyzing and re-engineering the hair dryer is the identification and modelling of the customer requirements. The hair dryer requirements list is generated from exiting literature. The requirements are listed in the Requirement column of Figure 4.

4.3 Function Modelling

The functionality of the hair dryer is determined through a combination of existing literature, product decomposition, and by developing a function structure [1]. The complete function structure diagram is not included for brevity; the resulting functional hierarchy is presented in Figure 2.

		F1.1	F1.2	F1.3	F1.4	F1.5	F1.6	F1.7	F2.1	F2.2	F2.3	F2.4	F3.1	F3.2	F3.3
		Provide electricity	Supply air	Convert electricity to rotational	Convert rotational to flow	Support flow generation	Convey flow	Control flow	Supply electricity	Convert electric to heat	Control temperature	Transfer heat to air	Provide handle	Provide controls	Protect user
F1	Provide airflow	1	1	1	1	1	1	1	0	0	0	0	0	0	0
F2	Heat air	0	0	0	0	0	0	0	1	1	1	1	0	0	0
F3	Provide user interface	0	0	0	0	0	0	0	0	0	0	0	1	1	1

Figure 2 - Matrix based function hierarchy

4.4 Component Modelling

The most straightforward modelling in this method is establishing the component-assembly model. The assembly-component decomposition is usually the easiest and most-straightforward because designers can physically examine the assembly and component structure. The assembly-component modelling is completed through reverse engineering and comparison to existing literature. Minor differences in the models resulted because of variations in the hair dryer chosen. However, these changes significantly affect the results and observations. The component-assembly model is represented in matrix form in Figure 3.

		C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16
		Switch	Fan Housing	Motor	Fan Blade	Heating element	Springs	Thermocouple	Temperature Switch	Heat Shield	Front Grid	Front case	Power cord	Switch Actuator	Rear housing	Screen	Ground wire
A1	Fan Assembly	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
A2	Heating Assembly	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
A3	Front Housing	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
A4	Rear Housing	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

Figure 3- Matrix based assembly-component hierarchy

4.5 Test Measurable Modelling

The final modelling step is to determine the performance metrics associated with the hair dryer. The engineering characteristics, or engineering attributes in the HoQ, provide a means for checking the performance of the system against the system requirements. The engineering characteristics for the hair dryer are extracted from [6]. The engineering characteristics and their associated units are included in the component-to-engineering characteristic matrix (see Figure 6).

4.6 Analysis of Requirements and Functions

The importance of customer requirements is determined by multiplying the weight of the requirement by mapping between requirements and functions. The weighted value is populated in each of the corresponding cells (see Figure 4). The weight of each the requirement is determined by the customer using a 9-3-1 cardinal scheme similar to that used in the HoQ. The cardinal ranking of each requirement is important because it decouples system requirements and enables customers to weight the requirement individually.

		F1.1	F1.2	F1.3	F1.4	F1.5	F1.6	F1.7	F2.1	F2.2	F2.3	F2.4	F3.1	F3.2	F3.3			
	Requirement	Provide electricity	Supply air	Convert electricity to rotational	Convert rotational to flow	Support flow generation	Convey flow	Control flow	Supply electricity	Convert electricity to heat	Control temperature	Transfer heat to air	Provide handle	Provide controls	Protect user	**Weight**	**Weighted Sum**	**Requirement Rank**
R1	Dries quickly	9	9	9	9	9	9	9	9	9	9	9	0	0	0	9	99	1
R2	Quiet	0	3	3	0	0	0	0	0	0	0	0	0	0	0	3	6	6
R3	Operates easily	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	4	7
R4	Operates safely	0	0	0	0	0	0	0	3	3	3	3	3	3	3	3	21	4
R5	Comfortable to hold	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	27	3
R6	Reliable	3	0	3	3	3	0	0	0	3	3	0	0	0	0	3	18	5
R7	Portable	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	2	8
R8	Energy efficient	9	0	9	9	9	0	0	9	9	0	9	0	0	0	9	63	2
	Weighted Sum	21	12	24	21	21	9	9	21	24	15	22	14	13	14			

Figure 4 - Hair dryer requirement-function matrix

The ranking of the system requirements is determined through the weighted column sum. For example, *Dries quickly* is ranked number one because of the high weight and relationship to eleven functions. Conversely, *Operates easily* has an importance of one and is related to four functions resulting in a rank ordering of seven. This type of analysis provides a first step in determining what functions are important to fulfil in the design of a product.

The importance of system functions is computed through a weighted column summation. The mapping between the requirements and the functions is multiplied by the weighting of each of requirement. The weighted mapping, shown in Figure 4, is then summed down each of the function columns. The weighted sum is then sorted in descending order to help designers determine the importance of system functions. For example, primary and secondary functions are determined using matrix mathematics and are tied directly to customer requirements. As illustrated in Figure 4, *Convert electricity to rotation* and *Convert electricity to heat* are primary functions and secondary functions include *Control Flow Convey Flow, and Provide Control.*

4.7 Analysis of Functions and Components

The assembly-component tree is a natural hierarchical representation of physical systems because it captures the component interaction based on assembly modules. However, it is not always the most beneficial grouping during design. It is equally important to determine the functionality of components in the system and subsequently analyze the system in terms of functional groups. The F-C matrix provides a means for capturing the functionality of individual components. Highly functional components are determined by summing the columns without taking into consideration the ranking of each function. As shown in Figure 5, the *Switch*, *Front Case*, *Switch Actuator*, *Rear Housing* are highly functional components and the *Heating Element* and *Motor* have a low functionality. The importance of each component is related to the functions is fulfils as well as the importance of each function. For example, the *Motor* and *Heating Element* are highly critical components and the *Screen* and *Temperature Switch* are low criticality.

		C1	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**	**C14**	**C15**	**C16**
	Function	Switch	Fan Housing	Motor	Fan Blade	Heating element	Springs	Thermocouple	Temperature Switch	Heat Shield	Front Grid	Front case	Power cord	Switch Actuator	Rear housing	Screen	Ground wire
F1.1	Provide Electricity	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
F1.2	Supply Air	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0
F1.3	Convert electricity to rotational	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
F1.4	Convert rotational to flow	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
F1.5	Support flow generation	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
F1.6	Convey Flow	1	1	0	1	0	0	0	0	0	1	1	0	0	1	1	0
F1.7	Control flow	1	1	0	0	0	0	0	0	0	1	1	0	1	1	1	0
F2.1	Supply Electricity	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1
F2.2	Convert electric to Heat	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
F2.3	Control Temperature	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0
F2.4	Transfer Heat to Air	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
F3.1	Provide handle	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
F3.2	Provide Controls	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
F3.3	Protect User	1	0	0	0	0	0	1	1	1	1	1	0	1	0	1	1
	Column Sum	6	4	2	3	2	1	3	2	2	3	5	2	5	5	4	2

Figure 5 – Hair dryer function-component matrix

4.8 Analysis of Components and Engineering characteristics

The performance, or behaviour, of a product must be determined in the context of the customer requirements established. However, customer requirements often cannot be measured and thus the design not evaluated. Engineering attributes are used in the HoQ as a means for mapping "soft" customer requirements to measurable quantities.

The engineering characteristics (EC) are correlated to the system components. As previously noted, the engineering characteristics are not directly related to customer requirements. First, the engineering characteristics associated with a system are dependent on the solution instantiation developed by the designers. For example, leakage amount is a very important measurable in the design of fluid based automotive cooling systems. However, this test measurable is non-existent for air-cooled automotive engines. Second, the mapping between customer requirements and engineering characteristic in the HoQ span several information domains.

The C-EC matrix, by itself, provides minimal insight into the design and analysis of the system. However, the C-EC matrix does enable the system requirements to be correlated against performance measures through matrix multiplication. For example, from the C-EC matrix presented in Figure 16 the weight and number of components of the hair dryer are important engineering characteristics because they are related to every component in the system. Naturally, the number of components and the weight of the hair dryer can be reduced by modifying any of the system components. However, *Air flow* and *Air temperature* are related to several components, seven and eight respectively. Thus, it can be inferred that these performance measures are achieved through a component-to-component coupling. Additionally, the *Air flow* and *Air temperature* have four common components: the *Motor*, *Fan Blade*, *Rear Housing*, and *Screen* (see Figure 6).

		EC1	**EC2**	**EC3**	**EC4**	**EC5**	**EC6**	**EC7**	**EC8**	**EC9**	
	Component	Air flow (m^3/sec)	Air temperature (degrees C)	Torque (N-m)	Weight (kg)	Volume (cm^3)	Number of parts (number)	Physical Lifetime (hours)	Energy consumption (Watt)	Noise / vibration (dB)	**Row Sum**
C1	Switch	0	0	0	1	0	1	1	0	0	3
C2	Fan housing	1	0	0	0	1	1	0	0	0	3
C3	Motor	1	1	1	1	0	1	1	1	1	8
C4	Fan blade	1	1	0	1	0	1	0	0	1	5
C5	Heating element	0	1	0	1	0	1	1	1	0	5
C6	Springs	0	0	0	1	0	1	0	0	1	3
C7	Thermocouple	0	1	0	1	0	1	0	0	0	3
C8	Temperature switch	0	1	0	1	0	1	1	0	0	4
C9	Heat shield	0	0	0	1	0	1	0	0	0	2
C10	Front grid	1	0	0	1	0	1	0	0	0	3
C11	Front case	1	0	0	1	1	1	0	0	0	4
C12	Power cord	0	0	0	1	0	1	1	0	0	3
C13	Switch actuator	0	1	0	1	0	1	1	0	0	4
C14	Rear housing	1	1	0	1	1	1	0	0	0	5
C15	Screen	1	1	0	1	0	1	0	0	0	4
C16	Ground wire	0	0	0	1	0	1	0	0	0	2
	Column Sum	7	8	1	15	3	16	6	2	3	61

Figure 6 – Hair dryer component-engineering characteristic matrix

4.9 Requirements to Component Analysis

As previously discussed, analyses can be completed at a higher level through matrix multiplication of the primary level matrices (see Figure 7 and Equation 1). The secondary and tertiary level matrices reveal implicit relationships with the model. The first computed matrix is derived from the R-F and the F-C matrices (see Equation 1). The R-C matrix reveals additional information about the system including how strongly or weakly a requirement is related to a component in the system through functionality.

		C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	
	Requirement	Switch	Fan housing	Motor	Fan blade	Heating element	Springs	Thermocouple	Temperature switch	Heat shield	Front grid	Front case	Power cord	Switch actuator	Rear housing	Screen	Ground wire	**Row sum**
R1	Dries quickly	36	36	18	27	18	9	18	9	9	18	18	18	36	27	27	9	333
R2	Quiet	0	3	3	0	0	0	0	0	0	0	0	0	0	3	3	0	12
R3	Operates easily	2	0	0	0	1	0	1	1	2	1	3	0	1	2	1	1	16
R4	Operates safely	9	0	0	0	6	3	9	6	6	3	9	3	9	6	3	6	78
R5	Comfortable to hold	18	0	0	0	0	0	9	9	9	9	27	0	9	18	9	9	126
R6	Reliable	3	3	6	6	3	3	3	3	0	0	0	3	6	0	0	0	39
R7	Portable	1	0	0	0	0	0	1	1	1	1	2	0	1	1	1	1	11
R8	Energy efficient	18	9	18	18	18	0	9	0	9	0	0	18	18	0	0	9	144
	Column Sum	87	51	45	51	46	15	50	29	36	32	59	42	80	57	44	35	

Figure 7 – Requirement to component mapping matrix

4.10 Requirements to Assembly Analysis

A secondary analysis completed is requirement distribution or allocation to assemblies (see Equation 5). The assembly-component matrix and the requirements-to-components matrix are multiplied, resulting in the requirement-to-assembly (R-A) matrix. The R-A describes the relationships between requirements and assemblies and enables designers to determine how strongly requirements are related to a physical collection of components (see Figure 8).

		A1	A2	A3	A4
	Requirement	Fan assembly	Heating assembly	Front housing	Rear housing
R1	Dries quickly	117	54	99	63
R2	Quiet	6	0	0	6
R3	Operates easily	2	3	7	4
R4	Operates safely	9	24	30	15
R5	Comfortable to hold	18	18	54	36
R6	Reliable	18	12	9	0
R7	Portable	1	2	5	3
R8	Energy efficient	63	27	45	9
	Column Sum	234	140	249	136

Figure 8 – Requirement to assembly analysis

The cell values in the R-A matrix do not have absolute meaning. However, the values capture the relative importance of requirements-to-assemblies. The matrix can be read across a row or down a column to provide different insight and system analysis. For example, the *Dries quickly* requirement is related to all of the assemblies in the product. However, the requirement is related to the *Front Housing* and *Rear Housing* stronger than to the *Heating Assembly*. This is not obvious based on the perceived functionality of the components, but can be attributed to the coupled functionality of both heating and generation of air flow. Thus, the *Fan Assembly* is strongly related to the requirement of *Dries quickly*.

4.11 Function to Assembly Analysis

The relationships between the functions and assemblies are computed by multiplying the assembly-component matrix and the function-to-component mapping matrix. The resulting matrix is the function-to-assembly (F-A) matrix (see Figure 9).

	Function	A1 Fan assembly	A2 Heating assembly	A3 Front housing	A4 Rear housing
F1	Provide electricity	1	0	2	0
F2	Supply air	1	0	0	2
F3	Convert electricity to rotational	1	0	0	0
F4	Convert rotational to flow	1	0	0	0
F5	Support flow generation	3	0	0	0
F6	Convey flow	3	0	2	2
F7	Control flow	2	0	3	2
F8	Supply electricity	1	1	2	1
F9	Convert electric to heat	0	1	0	0
F10	Control temperature	0	3	1	0
F11	Transfer heat to air	0	1	1	0
F12	Provide handle	0	0	1	1
F13	Provide controls	1	0	1	1
F14	Protect user	1	2	4	2
	Column Sum	**15**	**8**	**17**	**11**

Figure 9 – Function to assembly analysis

The F-A matrix enables coupled assemblies to be identified and the possibility of grouping assemblies into modules. Additionally, the function-assembly matrix indicates which assemblies provide functionality in the system. As shown in Figure 9, the *Front Housing* is a highly functional assembly. The *Front Housing* is associated with nine functions. Conversely, the *Heating Assembly* exhibits low functionality. The *Front Housing* assembly and *Heating Assembly* are comprised of four and five components respectively. The high level of functionality of the *Front Housing* is because the electrical power is supplied from the *Power Cord* and controlled through the *Switch Actuator*, two components in the *Front Housing* assembly. It is important to note, the analysis and subsequent conclusions from the F-A matrix are not based on functional or requirement criticality. Additional insight and conclusions can be gained by capturing the failure probability and criticality of a component and function and augmenting the analysis methods. Additional quantitative information is required from previous designed products to determine the vital components and assemblies in the hair dryer.

4.12 Requirement to Engineering Characteristic Analysis

The relationships between requirements and engineering characteristics are important in analyzing technical systems (see Equation 2 and Figure 10).

	Requirement	EC1 Air flow	EC2 Air temperature	EC3 Torque	EC4 Weight	EC5 Volume	EC6 Number of parts	EC7 Physical Lifetime	EC8 Energy consumption	EC9 Noise / vibration
R1	Dries quickly	171	180	18	297	81	333	135	36	54
R2	Quiet	12	9	3	9	6	12	3	3	3
R3	Operates easily	7	7	0	16	5	16	5	1	0
R4	Operates safely	21	39	0	78	15	78	33	6	3
R5	Comfortable to hold	63	54	0	126	45	126	36	0	0
R6	Reliable	15	27	6	36	3	39	24	9	15
R7	Portable	5	5	0	11	3	11	3	0	0
R8	Energy efficient	45	81	18	135	9	144	90	36	36

Figure 10 – Requirement to engineering characteristic matrix

The R-EC matrix enables the identification of key requirements and engineering characteristics. The *Dries quickly*, *Comfortable to* hold, and *Energy efficient* requirements are related to the engineering characteristics of the system. Additionally, the *Air flow*, *Air temperature*, and *Energy Consumption* are important measures that must be tested to ensure the hair dryer meets the needs of the customer. An important characteristic of the R-EC matrix is the similarity to the HoQ. The HoQ provides a concise means for directly mapping customer requirements to measurable parameters. However, the relationships between requirements and engineering characteristics in the R-EC are computed based on relationships between requirements, functions, and components. While the values in each of the cells may be different, the pattern in the matrices should be the same. For example, the HoQ developed for the hair dryer in [6] differs from the R-EC. The R-EC matrix differs not only in values, but also differs in their patterns. For example, the HoQ indicates there is no relationship between *Energy efficient* and *Energy Consumption*, whereas the R-EC matrix indicates a relationship exists. The R-EC matrix smears the *Weight*, *Volume*, and *Number of parts* test measures because each of the components in the system contributes to each of these measures.

5 CLOSURE

During the development of complex engineering systems, designers generate, analyze, and make decisions about the product based on conceptual design information including: requirements, functions, assemblies and components, and engineering characteristics. Several matrix-based modelling methods have been developed that enable designers to model and analyze this information. However, limitations of the existing matrix-based methods include: insufficient algorithms for analysis of the system, mapping of non-adjacent information domains, and not being closely tied to systematic design methods. Thus, the matrix-based modelling approach presented in this paper was developed to address the afore-mentioned shortcomings. The method enables designers to model requirements, functions, assemblies, components, and engineering characteristics as one would generate if following a systematic design process. Additionally, the matrices provide a concise means for visualizing complex product information. The modelling scheme consists of three primary-level matrices: the requirements-to-functions matrix, the functions-to-components matrix, and the components-to-engineering characteristics matrix. Three additional matrices are computed through matrix multiplication from the primary matrices. In addition to the matrices for modelling product information, several methods are developed as attention directing tools. These analyses enable designers to focus attention on key properties and target areas in the system. The following types of analysis are completed on the matrices: clustering of elements that are related based on summations of rows and columns, existence or lack of existence of mappings between information domains, and mapping between non-adjacent information domains through matrix multiplication.

REFERENCES

[1] Pahl G. and Beitz W. Engineering Design: A Systematic Approach 2nd edition, 1996 (Springer, New York).

[2] Johannson O. and Krus P. Configurable Design Matrices for Systems Engineering Applications. In 26th ASME Computers and Information in Engineering (CIE) Conference, Philadelphia, Pennsylvania USA, September 2006, Paper No. DETC2006-99481.

[3] Ghoniem M., Fekete J-D., and Castagliola P. A Comparison of Readability of Graphs Using Node-Link and Matrix-Based Representation. In IEEE Symposium on Information Visualization, Austin, Texas USA, October 2004, pp. 17-24

[4] Steward D. V. Systems Analysis and Management; Structure, Strategy and Design, 1981 (Petrocelli Books, Inc., New York)

[5] Hauser J.R. and Clausing D. The House of Quality. Harvard Business Review, 1988, 3, pp. 63-73.

[6] Olewnik A.T. and Lewis K.E. On Validating Design Decisions Methodologies. In 15th ASME Design Theory and Methodology (DEC) Conference, Chicago, Illinois USA, 2003, Paper No. DETC2003/DEC-48669.

[7] Otto K. and Wood K. Product Design, 2001 (Prentice-Hall, Upper Saddle River, NJ).

[8] Eppinger S. D. Model-based Approaches to Managing Concurrent Engineering. Journal of Engineering Design, 1991, 2, pp. 283-290.

[9] Ulrich K. and Eppinger S. Product Design and Development. 1999 (McGraw-Hill Inc., New York).
[10] Suh N.P. Axiomatic Design Theory for Systems. Research in Engineering Design, 1998, 10 (4), pp. 189-209.
[11] Arunajadai S.G., Stone R.B. and Tumer I.Y. Failure Mode Identification through Clustering Analysis. Quality and Reliability Engineering International Journal, 2004, 20, pp.511–526.
[12] Henning S. and Paasch R.K. Diagnostic Analysis for Mechanical Systems. In 12th ASME International Conference on Design Theory and Methodology (DTM) Conference. September 10 – 13, 2000. Baltimore Maryland USA, Paper No. DETC2000/DEC14580.
[13] Mocko G.M. and Paasch R.K. Incorporating Uncertainty in Diagnostic Analysis of Mechanical Systems. ASME Journal of Mechanical Design, 2005, 127(2), pp. 315-325.
[14] Leung P., Ishii K., Benson J. and Abell J. System Engineering Workshare Risk Analysis. In 11th ASME Design for Manufacturing and the Life Cycle (DFMLC) Conference, Philadelphia, Pennsylvania USA, 2006, Paper No. DETC2006-99252.
[15] Kwong C.K. and Bai H. Determining the Importance Weights for the Customer Requirements in QFD Using a Fuzzy AHP with an Extent Analysis Approach. IEEE Transactions, 2003, 35, pp. 619 – 626.
[16] Kmenta S. and Ishii K., Advanced FMEA Using Meta Behavior Modelling for Concurrent Design of Products and Controls, 18th ASME Computers and Information in Engineering Conference (CIE) Atlanta, Georgia USA, 1998. Paper No. DETC1998/CIE-5702.
[17] Kmenta S. and Ishii K. Scenario-Based FEMA: A Life Cycle Cost Perspective. In 14th Reliability, Stress Analysis, and Failure Prevention (RSAFP) Conference, Baltimore, Maryland USA. 2000, Paper No. DETC2000/RSAFP-14478.
[18] Masui K., Sakao T., Aizawa S. and Inaba A. Quality Function Deployment for Environment (QFDE) to Support Design for Environment (DFE). In 7th ASME Design for Manufacturing Conference (DFM), Montreal, Canada, 2002, Paper No. DETC2002/DFM-34199.

Dr. Gregory M. Mocko
Clemson University, Department of Mechanical Engineering, 243 Fluor Daniel Building
P.O. Box 340921, Clemson, South Carolina, USA
Telephone/Fax (864) 656-1812/4435, Email: gmocko@clemson.edu