The Function-Human Error Design Method (FHEDM)

Conference Paper · August 2018

DOI: 10.1115/DETC2018-85327

Nicolas Soria
Oregon State University

H. Onan Demirel
Oregon State University

Irem Y. Tumer
Oregon State University

Proceedings of the ASME 2018 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2018
August 26-29, 2018, Quebec, Canada

DETC2018-85327

THE FUNCTION-HUMAN ERROR DESIGN METHOD (FHEDM)

Nicolás F. Soria Zurita
School of Mechanical, Industrial
and Manufacturing Engineering
Oregon State University
Corvallis, Oregon, 97331
Email: soriazun@oregonstate.edu
Colegio de Ciencias e Ingeniería
Universidad San Francisco de Quito
Quito, Ecuador

Robert B. Stone
School of Mechanical, Industrial
and Manufacturing Engineering
Oregon State University
Corvallis, Oregon, 97331
Email: rob.stone@oregonstate.edu

H. Onan Demirel *
School of Mechanical, Industrial
and Manufacturing Engineering
Oregon State University
Corvallis, Oregon, 97331
Email: onan.demirel@oregonstate.edu

Irem Y. Tumer
School of Mechanical, Industrial
and Manufacturing Engineering
Oregon State University
Corvallis, Oregon, 97331
Email: irem.tumer@oregonstate.edu

ABSTRACT

During the design of products and systems, engineers must quickly and accurately satisfy customer needs while adequately developing the required system functions with the minimum number of failures. Identifying potential failure modes during early design stages is essential to create reliable designs. Different engineering methodologies such as Failure Modes and Effects Analysis (FMEA), allows engineers to identify how a set of components could fail. These methods are popular and commonly used in industry. However, such methodologies fail to recognize potential failure modes caused by human-product interaction. During the design of products, there is often a lack of sufficient attention to the human-product interaction. Even though human factors are considered during the design process, most of the design approaches fail to incorporate the human interaction correctly. In this research, we explore the implementation of a novel design methodology named Function-Human Error Design Method (FHEDM), which identifies possible generic human errors while completing a functional decomposition of the product. This method will provide engineers with useful information about potential failure modes caused by human-function interaction during early conceptual design.

1 INTRODUCTION

Human factors is one of the most important considerations during the design of a product or a system. Human factors engineers are responsible for the analysis, development and design of engineering system in which the primary objective is to incorporate capabilities and limitations of users within the system. [1].

---

*Address all correspondence to this author.

Conventional design practices treat human factors as a critical factor during the design process. However, the methodologies used in practice incorporate human factors engineering (HFE) principles partially during early design phases. Usually, HFE is treated in isolation from the system design process.

The human-product interactions assessment generally occurs towards the end of the design process and often requires full-scale physical prototypes and extensive human-subject data collection. System usability and safety are tested using human factors checklists and guidelines. Applying HFE after the system has been designed is not only costly and time-consuming but also have its limitations.

Design approaches that consider the user during early stages of design can significantly enhance the usability and capability of the final system. It would reduce the cost and time associated with the physical prototyping and human-subject data collection.

Functional modeling is a well-known early design technique that decomposes a product systematically into its essential functions, allowing engineers to build a functional representation of the system with no consideration for form or shape of the final product. However, functional modeling is incomplete and inadequate when it comes to analyzing the user-product interactions. Potential failure modes caused by user-product interactions cannot be directly identified or analyzed because the final user is not considered or implemented during the functional decomposition due to lack of form specification.

This paper introduces the Function-Human Error Design Method (FHEDM) which incorporates the user-product interactions to identify possible generic human errors while the user is interacting with different functions of the product. FHEDM offers a new approach for coupling user-product interactions with generic human errors during the conceptual design stage. FHEDM incorporates HFE within a functional model framework. Product functions are modeled using the Functional Basis [2], and the user-function interactions are mapped using Actionfunction diagrams presented by Sangelkar and McAdams [3, 4]. This paper explains the application of HFE to identify human errors in user-product interactions during conceptual design.

The remainder of this paper is divided into several sections, organized as follows. Section 2 first introduces the topic of adverse events & human error, and human factors engineering & human error; followed by a review of the functional modeling, and Actionfunction diagrams. Section 3 presents the Function-Human Error Design Method (FHEDM), providing a formalized methodology and general guidelines for using the method. Section 4 present in detail the application of FHDM to an electrical screwdriver; followed by the procedure of applying FHDM to eight consumer products selected from a design database. Finally, a discussion of the method followed by recommendations and future work is presented in Section 5.

2 BACKGROUND

In the following section, we first present adverse events and human error, followed by the perspectives used in human factors engineering to deal with human error. Finally, we provide a short introduction to functional modeling and the Functional Basis and present how the user interactions are incorporated into the Functional Basis framework using an Actionfunction diagrams.

2.1 Adverse Events and Human Error

Adverse events are deviations from normative procedures caused by errors. Commonly, errors are thought to be made exclusively by human operators or by machinery. However, current research on engineering and psychology view errors as deviations in standardized procedures that are the result of the combination of latent failures and active failures [5, 6].

Latent failures are those errors whose consequences lie dormant inside the system for long periods of time, that are activated when different factors are combined [5].

Active failures are those errors whose effects appear immediately in the system [5]. Active failures are actions (or the lack of actions) that trigger a chain of events resulting in an accident. Human operators are commonly blamed to cause active failures.

Human error is cited as the probable cause of 70 to 90 percent of accidents in systems such as aircraft, trains, ships, medical facilities and nuclear power plants [7–9]. Human error groups all those events in which a planned sequence of physical and mental activities fails to complete its predetermined objective, and when these failures cannot be attributed to the intervention of some random external factors [5, 10].

To diminish this cause of accidents, many researchers have focused their efforts on understanding and evaluating the concept of human error [5, 7–9]. Different models and frameworks are used to describe and classify human error. The primary error taxonomies in literature are Model of Internal Human Malfunction, Information Processing Model, and Model of Unsafe Acts.

- Model of Internal Human Malfunction [11] presents three basic levels of human performance. The skill-based model describes automated actions that follow an intention (sensory-motor behavior). The rule-based model describes the procedures or techniques that guide the action. Finally, the knowledge-based model represents actions that are developed to deal with situations that are not familiar with a standard operation (Skill-Rule-Knowledge model).
- Information Processing Model [12] presents a human error classification framework through a series of mental operations that begins with sensory stimulus and ends with the execution of physical responses.
- Model of Unsafe Acts [5] differentiates human errors in slips, lapses, mistakes, and violations. Slip and lapses occur when there is failure or omission in an execution. Mistakes are

the result of a judgment process while selecting or accomplishing a task. Violations are intentional deviations from standard operating procedures or practices.

However, even though there are different approaches to model human error, literature provides no guidance on how to classify an event into error categories [13].

2.2 Human Factors and Human Error

Human factors engineering has two different perspectives on human error and their contributions to adverse events.

The first perspective, also known as "the old view", points out human error as the cause of system failure. In this perspective, human error is responsible for the accidents in engineering systems [14, 15]. Engineering systems are designed and built to be safe. Safety is an inherent property of engineering systems. The threat to system safety comes from the inherent unreliability of humans [16]. Therefore, engineering systems need protection from users through training, standardization of procedures, and discipline.

The second perspective, also know as "the new view", points out human error as a symptom of a defective system [14, 15, 17]. In this perspective, engineering systems are not safe because systems are contradictions between multiple goals that are being pursued simultaneously. Safety is not an inherent property of the engineering systems. Engineers have to incorporate safety into the system. Human error is associated with characteristics and relationships between the user tasks, user tools, and the operating environment [16]. If designers want to achieve safety in engineering systems successfully, they need to understand such characteristics and relationships.

Instead of using the label of "human error" as the reason for failure, engineers need to investigate and understand system factors that triggered the failure such as design problems, procedural shortcomings, and organizational deficiencies.

2.3 Functional Modeling and Functional Basis

Functional modeling is a well-known design stage technique that decomposes a product systematically into its essential functions, allowing engineers to build a functional representation of the system with no consideration for form or shape of the final product. [18–20]. The Functional Basis is a standardized design vocabulary that uses a verb-object (function-flow pair) format to describe the different functions and flows working together within the functional model of a product [19, 21].

Together, the functional modeling technique and the Functional Basis standardize a design language that uses a verb-object (function-flow pair) format to describe how the functions interact with the flows moving through a system. This representation follows a flow diagram which illustrates the states and transitions of flows of energy, material, and information between the functions. Table 1 presents the different material, energy, and signal flows that can be used when creating a functional model, while Table 2 presents the eight function classes used in the Functional basis. An example of a functional modeling decomposition of a B&D electrical screwdriver is illustrated in Figure 1.

TABLE 1. CLASSES, FLOWS TYPES, AND COMPLEMENTS [21]

Class	Material	Signal	Energy	
Basic	Human	Status	Human	Hydraulic
	Gas	Signal	Acoustic	Magnetic
	Liquid		Biological	Mechanical
	Solid		Chemical	Pneumatic
	Plasma		Electrical	Radioactive
	Mixture		Electromagnetic	Thermal

TABLE 2. FLOW CLASSES, AND THEIR BASIC CATEGORIZATION [21]

Class	**Basic**	**Class**	**Basic**	**Class**	**Basic**
Branch	Separate	Control	Actuate	Signal	Sense
	Distribute		Regulate		Indicate
Channel	Import	Magnitude	Change		Process
	Export		Stop	Support	Stabilize
	Transfer	Convert	Convert		Secure
	Guide	Provision	Store		Position
Connect	Couple		Supply		
	Mix				

The functional model and Functional Basis for design are thoroughly accepted methods for representing product function [19–23]. However, these modeling techniques are incomplete and inadequate when it comes to analyzing the user-product interactions. Humans are only shown as material and energy flows. Representing humans as flows is an inadequate approach to describe the interaction of the user and functions of the product. The functions of a product in which the user is involved need to be identified if engineers want to design a product that is functional for humans. Furthermore, potential failure modes caused by user-product interactions are not identified or analyzed due to the final user not being considered during the functional decomposition.

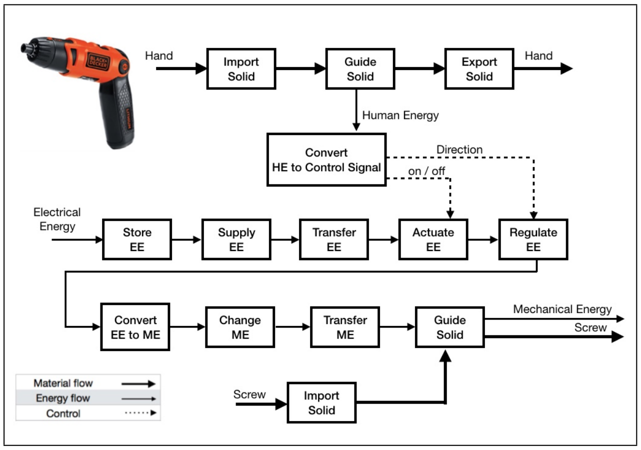

FIGURE 1. BLACK AND DECKER ELECTRICAL SCREWDRIVER FUNCTIONAL MODEL

2.4 Actionfunction Diagram

Actionfunction diagram is a product analysis framework used to enhance inclusive design by combining in one graphical representation activity diagrams and functional models to model the user-product interaction [3, 4, 24, 25]. Actionfunction diagrams help designers to analyze user-product interactions in early stages of design by coupling the interaction of user tasks and customer needs [26]. It relies upon the International Classification of Functioning, Disability, and Health (ICF), a standardized language to describe health and health-related states which was established by the World Health Organization [27]. Product functions are modeled using the Functional Basis, and user activities are represented using ICF Lexicon [27, 28]. An example of an Actionfunction diagram of an B&D electrical screwdriver is illustrated in Figure 2. The dashed blocks presented in the figure refer to actions performed by the user with the associated functions of the system. Systems will present functions that do not require the intervention of the user. Such functions would not be mapped in the Actionfunction diagram.

The Actionfunction diagram is a useful method for representing product user interaction. Results of the studies executed by Kostovich, Sangelkar and McAdams showed that the Actionfunction diagrams improve universal design research and practice by providing a clear coupling between the interaction of user [4, 24, 25]. Additionally, Actionfunction diagrams provide valuable information for developing to the needs of individuals with a disability [26].This paper builds on the Actionfunction diagram and Functional Basis methods to create a framework that allows designers to identify potential product failure modes caused by errors due to poorly considered user-product interactions.

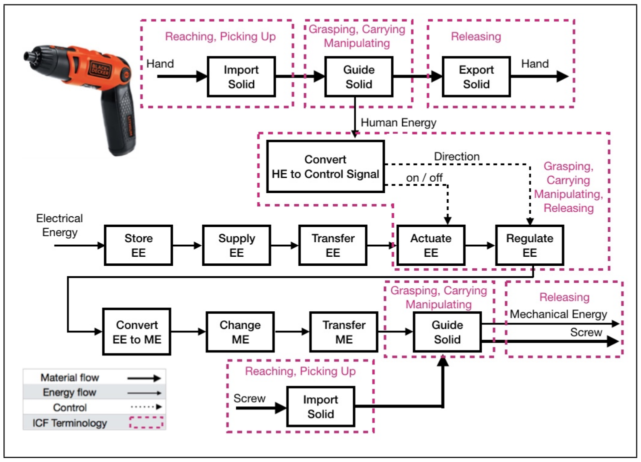

FIGURE 2. BLACK AND DECKER ELECTRICAL SCREWDRIVER FUNCTIONAL MODEL ACTION-FUNCTIONS USER DIAGRAM.

3 METHODOLOGY

In the following sections, we present the Function-Human Error Design Method (FHEDM). Specifically, we outline the steps required to form the Function-User Interaction Error matrix $[FUIE]$. The framework is presented in Figure 3.

Using the functional model and the Actionfunction diagram of the product, we obtain the Function Vector $[F]$ and User Interaction Vector $[UI]$. Combining these vectors, we proceed to build the Function-User Interaction matrix $[FUI]$. Then, using a generic human error database, we proceed to build the Human Error Vector $[HE]$. Combining the User Interaction Vector $[UI]$ and Human Error Vector $[HE]$, we generate the Human-Interaction Error Matrix $[HIE]$. The Function-User Interaction Error matrix $[FUIE]$ is formed from the matrix multiplication of the two matrices ($[FUIE] = [FUI] \times [HIE]$).

The Function-Human Error Design Method (FHEDM) breaks into three steps, which are described in detail in the following subsections.

3.1 The Function-User Interaction matrix [FUI]

We form the Function-User Interaction matrix $[FUI]$ with the function vector [F] and user interaction vector $[UI]$. The m-dimensional function vector $[F]$ captures the set of functions describing the system. The n-dimensional user interaction vector $[UI]$ identifies the set of activities or tasks that the user needs to complete to operate the product successfully. The m rows of the matrix are populated from the $[UI]$ tasks that correlate to the function $[F]$ that each row represents, forming a $m \times n$ matrix. We call this $m \times n$ matrix the $[FUI]$ matrix. For a given user task a number 1 is entered in the cells corresponding to the function it performs, and a number 0 is entered in the remaining cells. A typical Function-User Interaction matrix $[FUI]$ is presented Table 3.

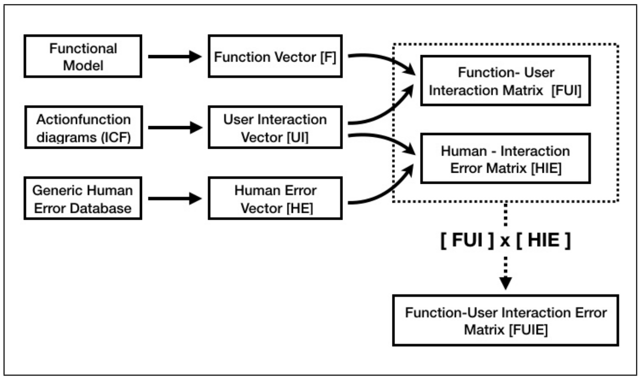

FIGURE 3. FUNCTION-HUMAN ERROR DESIGN METHOD (FHEDM) FLOW CHART

TABLE 3. A GENERIC FUNCTION-USER INTERACTION MATRIX [FUI]

	User :					
	Task-1:	Task-2:		...		Task-n
Function-1	1	0	0	0	0	0
Function-2	0	1	0	1	0	0
⋮	0	0	0	0	1	0
Function-m	0	1	1	0	1	0

3.2 Generic human error database

We next create a human task database to facilitate the human error mapping within a functional model based design process. The database is composed using human factors engineering literature [29–31]. First, the human task database uses a glossary of primary verb phrases that describe different basic human tasks with their interacting object. Then we add a compendium of interpretation of the primary verbs to the respective user activity (task) model. Next, we complete a list of generic human errors that can occur in achieving the individual activity, followed by a list of possible fallibilities between the human-product interaction. The generic human error database is presented in Appendix A in Table 9. At this time, the generic human error database is part of the preliminary work for this research. Future work will address validation and define rates for severity and frequency.

For each possible error, the two first columns contain a verb and the interacting object. Each verb phrase is associated with human functional ability and limitation definition using the ICF lexicon. The ICF lexicon allows us to map the interaction between the user (human) activities and the product functionality in the functional model [4]. The third column labeled generic human error describes an error that may occur when the user is completing the task. Finally, the interaction fallibilities column lists one or more product limitations that may contribute to the generic error.

3.3 The Human-Interaction Error matrix [HIE]

We next form The Human-Interaction Error matrix $[HIE]$ with the user interaction vector $[UI]$ and the human error vector $[HE]$. The p-dimensional human error vector $[HE]$ identifies the generic human errors that the user could commit while completing a specified task or activity. The n rows of the matrix are populated from the human error vector $[HE]$ that correlate to the $[UI]$ tasks that each row represents, forming a $n \times p$ matrix. We call this $n \times p$ matrix the $[HIE]$ matrix. As described in the previous matrix, a number 1 is placed to the cells corresponding to the generic human error it experienced, and a number 0 is set in the remaining cells. The Human-Interaction Error matrix [HIE] is presented in Table 4.

TABLE 4. A GENERIC HUMAN-INTERACTION ERROR MATRIX [HIE]

	Generic human error:					
	Error-1:	Error-2:		...		Error-p
User task-1	1	0	0	0	0	0
User task-2	0	1	0	1	0	0
⋮	0	0	0	0	1	0
User task-n	0	1	1	0	1	0

3.4 Function-User Interaction Error matrix [FUIE]

Finally, we assemble the Function-User Interaction Error matrix $[FUIE]$ by the matrix multiplication of the Function-User Interaction matrix $[FUI]$ (Table 3) and the Interaction Error matrix $[HIE]$ (Table 4) :

$$[FUIE] = [FUI] \times [HIE] \tag{1}$$

The resulting $m \times p$ matrix is called the $[FUIE]$ matrix. The cells of the matrix provide information on the number of occurrences about a particular human error for a given function of the system. A system function with a high number in the matrix presents a potential failure mode in the system caused by human-system interaction. The FUIE matrix will allow the designer to identify clusters for possible user errors. Designers can use the generic human error database (Table 9) to expand the description of the user error and to determine which interaction fallibilities can prevent the user to complete the desired task with the system.

4 RESULTS

In this section, the Function-Human Error Design Method (FHEDM) described in Figure 3 is applied to a representative set of products selected from the Design Repository[1]. The Design Repository is hosted by the Design Engineering Lab at Oregon State University and provides a database of design knowledge at various levels of abstraction, from form (components, subassemblies, and assemblies) to architecture description to function, during the product development process [32–34].

For this preliminary study, eight products were selected: a Black and Decker (B&D) electric screwdriver, a B&D electric jigsaw, a Delta circular saw, a Delta nail gun, a generic game controller, a lawnmower, a Razor scooter, and a Firestorm saber saw. These set of products were selected because human flows for energy and material were adequately identified in the functional models enabling the construction of the Actionfunction diagrams. The functional model for each product was extracted from the Design Repository database and graduate engineering students were used as subject matter experts (SME) to develop the Actionfunction diagram for each product to identify user-product interacting functions. The information gathered from the SME was used to build the Function-User Interaction Matrix [FUI] for each of the eight products. The B&D electric screwdriver is presented in the following subsections as an example to describe the three steps required to build the Function-User Interaction Error matrix [FUIE].

4.1 The Function-User Interaction matrix [FUI]

The Function-User Interaction matrix [FUI] is developed using the functional Model and the Actionfunction diagram of the system under study. The Functional model for the B&D electric screwdriver, illustrated in Figure 1, was extracted from the Design Repository database. The Actionfunction diagram for the B&D electric screwdriver, illustrated in Figure 2, was created using the ICF terminology and design knowledge provided by the subject matter experts.

Using the information gathered from the Actionfunction diagram, the Function-User Interaction matrix [FUI] was developed and implemented using an Excel workbook. The resulting FUI matrix for the B&D electric screwdriver is presented in Table 5. For a given user task a number one is entered in the cells corresponding to the function it performs. For this example, the function "Import Solid (Hand)" and the user tasks "Reaching" and "Picking up" have a direct relationship. The user needs to reach and pick up the electric screwdriver to start using the device. A number zero is recorded if there is no interaction between the user and the system function. For this particular example, the following system functions that do not interact with the user while the system is being operated Store Electrical Energy, Supply Electrical Energy, Transfer Electrical Energy, Convert Electrical Energy to Mechanical Energy, Change Mechanical Energy, and Transfer Mechanical Energy.

TABLE 5. BLACK AND DECKER ELECTRICAL SCREWDRIVER FUNCTION-USER INTERACTION MATRIX [FUI]

	User Task:					
System Functions:	Reaching	Picking up	Grasping	Carrying	Manipulating	Releasing
Import Solid (Hand)	1	1	0	0	0	0
Guide Solid (Hand)	0	0	1	1	1	0
Export Solid	0	0	0	0	0	1
Convert HE to CS	0	0	1	1	1	1
Store EE	0	0	0	0	0	0
Supply EE	0	0	0	0	0	0
Transfer EE	0	0	0	0	0	0
Actuate EE	0	0	1	1	1	1
Regulate EE	0	0	1	1	1	1
Convert EE to ME	0	0	0	0	0	0
Change ME	0	0	0	0	0	0
Transfer ME	0	0	0	0	0	0
Guide Solid (Screw)	0	0	1	1	1	0
Import Solid (Screw)	1	1	0	0	0	0

4.2 The Interaction Error matrix [HIE]

The Interaction Error matrix [HIE] is developed using the user tasks identified in the Actionfunction diagram of the system under study and the generic human error database presented in Appendix B Table 9. Each user task identified from the Actionfunction diagram is mapped to its possible generic human error using an Excel workbook. For a given user task a number one is entered in the cells corresponding to the generic human error that could be present while achieving such task.

---

[1] https://design.engr.oregonstate.edu/repo

The resulting HIE matrix for the B&D electric screwdriver is presented in Table 6. For this example, three generic human errors would cause the user to fail the task “Reaching”. The first cause, the user fails to reach the specific target. The second cause, the user fails to reach a small object. The third cause, the user fails to position its body to reach the objective.

TABLE 6. BLACK AND DECKER ELECTRICAL SCREWDRIVER HUMAN-INTERACTION ERROR MATRIX [HIE]

	Generic human error - Fail to:															
User Task:	Reach specific target	Reach small object	Pickup	Release	Grasp target object	Grasp sliding object	Re-Grasp object	Manipulate	Apply force	Apply pressure	Move to indefinite location	Transfer object to other hand	Move to exact location	Transfer grasped object	Position body	Turn to desired location
Reaching	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Picking up	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0
Grasping	0	0	0	0	1	1	1	0	0	0	1	1	1	1	0	0
Carrying	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
Manipulating	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	1
Releasing	0	0	0	1	0	0	0	1	0	0	1	1	1	1	0	0

4.3 Function-User Interaction Error matrix [FUIE]

The Function-User Interaction Error matrix is the result of the matrix multiplication (Equation 1) of the Function-User Interaction matrix [FUI] by the Interaction Error matrix [HIE]. The FUIE matrix was calculated using an Excel workbook.

The FUIE matrix for the B&D electric screwdriver is presented in Table 7. The cells of the FUIE matrix provide information on the number of occurrences about a particular generic human error for a given function of the system. For this example, we can see the generic human error clustered in two groups of functions. The first cluster groups the possible errors that the user could encounter while guiding the screw to the system, and while guiding the screwdriver towards the screw. The second cluster groups the possible errors that the user would encounter while transforming the force of the hand into a control signal that activates and control the electrical energy of the screwdriver.

Once the clusters for possible user errors are identified in the FUIE matrix, designers can use the generic human error database (Table 9) to expand the description of the user error and to determine which interaction fallibilities can prevent the user to achieve the desired task.

TABLE 7. BLACK AND DECKER ELECTRICAL SCREWDRIVER FUNCTION-USER INTERACTION ERROR MATRIX [FUIE]

	Generic human error - Fail to:															
Product Functions	Reach specific target	Reach small object	Pickup	Release	Grasp target object	Grasp sliding object	Re-Grasp object	Manipulate	Apply force	Apply pressure	Move to indefinite location	Transfer object to other hand	Move to exact location	Transfer grasped object	Position body	Turn to desired location
Import Solid (Hand)	1	1	1	0	0	0	0	0	0	0	0	0	0	0	2	0
Guide Solid (Hand)	0	0	0	0	2	2	2	1	1	1	3	3	3	3	0	1
Export Solid	0	0	0	1	0	0	0	1	0	0	1	1	1	1	0	0
Convert HE to CS	0	0	0	1	2	2	2	2	1	1	4	4	4	4	0	1
Store EE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Supply EE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Transfer EE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Actuate EE	0	0	0	1	2	2	2	2	1	1	4	4	4	4	0	1
Regulate EE	0	0	0	1	2	2	2	2	1	1	4	4	4	4	0	1
Convert EE to ME	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Change ME	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Transfer ME	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Import Solid (Screw)	1	1	1	0	0	0	0	0	0	0	0	0	0	0	2	0
Guide Solid (Screw)	0	0	0	0	2	2	2	1	1	1	3	3	3	3	0	1

4.4 Composite Function-User Interaction Error matrix [FUIE]

As the complexity of the design increases, the number of system functions and the number of user-system interactions will increase. The first example illustrated in the previous subsections only covers a small set of system functions and user interactions. To build a design framework that allows designers to identify or recognize potential failure modes caused by user-product interactions, we need to compile an extensive database of Function-User interactions errors that can be applied to any system. To assess the proposed methodology, we created a small Function-User Interactions Errors database using the eight products selected from the Design Repository.

For the products selected, the Actionfunction diagrams were completed using the subject matter experts and the functional models extracted from the Design Repository. The Function-User Interaction matrices [FUI] and the Interaction Error Matrix [HIE] were built for each system following the same procedure described in the previous subsections. The resulting [FUI] and [HIE] matrices are aggregated, and then multiplied to get the composite Function-User Interaction Error matrix [FUIE]. The composite [FUIE] matrix is a representation of all the distinct Function-User error interactions identified in the representative set of products selected from the Design Repository.

By aggregating the matrices into a system database, the resulting quantitative Function-User interaction data can be used as an archive of design knowledge. Consider the B&D elec-

TABLE 8. ABSTRACTION OF THE COMPOSITE FUNCTION-USER INTERACTION ERROR MATRIX [FUIE] FOR

	Generic human error - Fail to:															
Product Functions	Reach specific target	Reach small object	Pickup	Release	Grasp target object	Grasp sliding object	Re-Grasp object	Manipulate	Apply force	Apply pressure	Move to indefinite location	Transfer object to other hand	Move to exact location	Transfer grasped object	Position body	Turn to desired location
Actuate Electric Energy	1	0	0	1	1	2	2	3	2	1	2	3	3	3	0	1
Separate Solid	0	0	0	1	1	2	2	2	1	0	3	4	4	4	0	1

tric screwdriver from the previous example. From the functional model we can identify fourteen system functions, and from Actionfunction diagram we can identify six user tasks. If we want to create a reliable design database, we must aggregate all the distinct combination of system functions with user tasks to identify all the Function-User interactions and the possible human error associated with such interactions. More details about matrix aggregation are provided in Stone et al. [35].

The aggregated Function-User Interaction Error matrix is mathematically defined by:

$$[FUIE]_c = \sum[FUI] \times \sum[HIE] \tag{2}$$

The composite Function-User Interaction Error matrix [FUIE] for the eight selected systems is presented in Appendix B in Table 10. The composite [FUIE] matrix provides a list of sixteen possible user errors which could occur while interacting with twenty-nine distinct product functions. The composite FUIE matrix entries indicate the number of instances in the system where a given function is related to the corresponding failure mode caused by a generic human error. With the possible Function-User interaction error identified, a designer can perform further analysis and implement user considerations to achieve an improved design solution.

Designers, while redesigning an existing product or designing a new one, can apply the Function-Human Error Design Method (FHDM) by selecting failure modes corresponding to the derived Function-User interaction errors. For example, let's consider one scenario were designers are developing a product that needs two functions: Actuate Electric Energy, and Separate Solid. The FHDM method would allow the designer to identify the corresponding Function-User interaction errors for each function. Table 8 is an abstraction of the composite FUIE matrix values for the functions under study. The Function-User interaction errors for these functions are presented below:

- Actuate Electric Energy: Fail to reach specific target. Fail to release. Fail to grasp target object. Fail to grasp sliding object Fail to re-grasp object. Fail to manipulate. Fail to apply force. Fail to apply pressure. Fail to move to indefinite location. Fail to transfer object to other hand. Fail to move to exact location. Fail to transfer grasped object Fail to turn to desired location.

- Separate Solid: Fail to release. Fail to grasp target object. Fail to grasp sliding object. Fail to re-grasp object Fail to manipulate. Fail to apply force. Fail to move to indefinite location. Fail to transfer object to other hand. Fail to move to exact location. Fail to transfer grasped object Fail to turn to desired location.

After the Function-User interaction errors are identified, designers can develop design solutions that mitigate or eliminate such errors. In this particular example, solutions for actuate electric energy and separate solid functions must be analyzed for improving grasping and different manipulating tasks.

5 DISCUSSION & CONCLUSIONS

This works shows the preliminary evidence on how human factors engineering (HFE) can effectively be incorporated in a functional modeling basis for the design of inclusive products. The Function-Human Error Design Method (FHDM) is a first step towards implementing and developing a standard vocabulary for the description of failure modes caused by the user-product interaction.

The B&D electric screwdriver example illustrates that with the Function-User interaction errors identified, a designer can perform further analysis and implement user considerations to explore a design solution that take into account the user (human) error while interacting with the system. The Function-Human Error Design Method (FHDM) is meant to provide designers with an analytical tool to identify potential failure modes caused by user-product interactions and identify product functions that have a direct impact on the user during the conceptual design stage. Currently, FHDM is only applied to a small sample of systems, most of which are power tools. To validate the proposed method more products need to be analyzed and decomposed using the Function-User Interaction [FUI] and Human-Interaction Error [HIE] matrices.

One limitation of building the FUI and HIE matrices is the nature of the functional models. Systems have multiple different functional models depending on the designer perspective. This model variation would add or reduce the detail of the Function-User interactions. If the system-user interactions are not mapped, the failure mode caused by such interactions cannot be identified. The functional models used in this paper were selected from the Design Repository to avoid omissions on system-user interactions caused by different designers perspective.

The generic human error database presented in this work is a significant contribution to the failure mode literature. The database allows us to identify possible human errors while completing a set of different activities (tasks) towards an end goal. As ongoing and future work, we plan to expand the detail of such human (user) activities, including some human cognition tasks such as perception of the task and the environment. At this time the generic human error database do not include the severity or frequency of the human error. Adding a severity and frequency risk numbers would allow designers to allocate resources to correct failure modes caused by human-system interactions. Future work will include severity and frequency risk numbers in the generic human error database.

The Composite Function-User Interaction Error matrix [FUIE] presented in Table 10 shows twenty-nine distinct system functions and sixteen generic human errors. This preliminary database is a valuable contribution to the design community. The composite FUIE matrix entries indicate the number of instances in the system where a given function is related to the corresponding failure mode caused by a generic human error. The composite FUIE matrix forms the basis for the Function-Human interaction knowledge that we can use in determining the potential failure modes caused by human-system interactions.

6 FUTURE WORK

Future work will apply the Function-Human Error Design Method to a large number of systems to validate the results regardless of system scale and complexity. As part of our validation efforts, we want to use the FHEDM during the redesign of an existent product or the design of a new product.

This paper dealt with the binary form of the Function-Human error interaction matrix. From the $[FUIE]$ matrix we can extract the number of occurrences of human errors while interacting with a product function. Therefore, we can rank the occurrences and obtain a probability distribution.

At this time the FHEDM method is not considering system maintenance operations or safe operating procedures. Failure modes caused by human-system interactions will arise while completing maintenance operations or during unsafe operations. Including these scenarios will significantly improve this research.

We plan to generate an overall function failure computational tool that combines the failures modes of components and functions introduced by Tumer and Stone [19] while considering the failures modes caused by user-product interaction.

ACKNOWLEDGMENT

This research is supported by the National Science Foundation award number CMMI-1363411. Any opinions or findings of this work are the responsibility of the authors and do not necessarily reflect the views of the sponsors or collaborators. Special thanks go to Universidad San Francisco de Quito for supporting the primary authors graduate studies.

REFERENCES

[1] Phillips, C. A., Repperger, D. W., and Reynolds, D. B., 2000. *Human factors engineering*. Wiley Online Library.

[2] Stone, R. B., Tumer, I. Y., and Van Wie, M., 2005. "The function-failure design method". *Journal of Mechanical Design,* ***127***(3), pp. 397–407.

[3] Sangelkar, S., and Mcadams, D. A., 2012. "Creating actionfunction diagrams for user centric design". In American Society for Engineering Education, American Society for Engineering Education.

[4] Sangelkar, S., Cowen, N., and McAdams, D., 2012. "User activity–product function association based design rules for universal products". *Design Studies,* ***33***(1), pp. 85–110.

[5] Reason, J., 1990. *Human error*. Cambridge university press.

[6] Reason, J., 1995. "Understanding adverse events: human factors.". *Quality and Safety in Health Care,* ***4***(2), pp. 80–89.

[7] Di Pasquale, V., Miranda, S., Iannone, R., and Riemma, S., 2015. "A simulator for human error probability analysis (sherpa)". *Reliability Engineering & System Safety,* ***139***, pp. 17–32.

[8] Jung, W. D., Yoon, W. C., and Kim, J., 2001. "Structured information analysis for human reliability analysis of emer-

gency tasks in nuclear power plants". *Reliability Engineering & System Safety,* ***71***(1), pp. 21–32.

[9] Gawron, V. J., Drury, C. G., Fairbanks, R. J., and Berger, R. C., 2006. "Medical error and human factors engineering: where are we now?". *American Journal of Medical Quality,* ***21***(1), pp. 57–67.

[10] Di Pasquale, V., Franciosi, C., Lambiase, A., and Miranda, S., 2016. "Methodology for the analysis and quantification of human error probability in manufacturing systems". In Research and Development (SCOReD), 2016 IEEE Student Conference on, IEEE, pp. 1–5.

[11] Rasmussen, J., Pedersen, O., Mancini, G., Garnino, A., Griffon, M., and Gagnolet, P., 1981. Classification system for reporting events involving human malfunctions. Tech. rep.

[12] Wickens, C. D., and Carswell, C. M., 2012. "Information processing". *Handbook of Human Factors and Ergonomics*, p. 117.

[13] Saurin, T. A., de Macedo Guimarães, L. B., Costella, M. F., and Ballardin, L., 2008. "An algorithm for classifying error types of front-line workers based on the srk framework". *International journal of industrial ergonomics,* ***38***(11), pp. 1067–1077.

[14] Cook, R. I., Woods, D. D., and Miller, C., 1998. *A tale of two stories: contrasting views of patient safety*. The Foundation.

[15] Reason, J., 2000. "Grace under fire: compensating for adverse events in cardiothoracic surgery". In 5th conference on naturalistic decision making.

[16] Dekker, S. W., 2002. "Reconstructing human contributions to accidents: the new view on error and performance". *Journal of Safety Research,* ***33***(3), pp. 371–385.

[17] Hoffman, R. R., and Woods, D. D., 2000. "Studying cognitive systems in context: Preface to the special section". *Human Factors,* ***42***(1), pp. 1–7.

[18] Otto, K., and Wood, K., 2001. "Product design: techniques in reverse engineering and new product design". *Prentice-Hall*.

[19] Stone, R. B., and Wood, K. L., 2000. "Development of a functional basis for design". *Journal of Mechanical design,* ***122***(4), pp. 359–370.

[20] Kurfman, M. A., Stock, M. E., Stone, R. B., Rajan, J., and Wood, K. L., 2003. "Experimental studies assessing the repeatability of a functional modeling derivation method". *Journal of Mechanical Design,* ***125***(4), pp. 682–693.

[21] Wood, K. L., Stone, R. B., Mcadams, D., Hirtz, J., and Szykman, S., 2002. "A functional basis for engineering design: Reconciling and evolving previous efforts". *None*.

[22] Ahmed, S., and Wallace, K., 2003. "Evaluating a functional basis". In Proc. Design Theory and Methodology, International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, ASME, Chigaco, Illinois DTM-48685.

[23] Caldwell, B. W., Sen, C., Mocko, G. M., Summers, J. D., and Fadel, G. M., 2008. "Empirical examination of the functional basis and design repository". *Design Computing and Cognition'08*, pp. 261–280.

[24] Kostovich, V., McAdams, D. A., and Moon, S. K., 2009. "Representing user activity and product function for universal design". In Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, San Diego, CA.

[25] Sangelkar, S., and McAdams, D. A., 2012. "Adapting ada architectural design knowledge for universal product design using association rule mining: A function based approach". *Journal of Mechanical Design,* ***134***(7), p. 071003.

[26] Sangelkar, S., and McAdams, D. A., 2011. "Formalizing user activity-product function association based design rules for universal products". In Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Washington, DC.

[27] Organization, W. H., 2001. *International Classification of Functioning, Disability and Health: ICF.* World Health Organization.

[28] Sangelkar, S., and McAdams, D. A., 2010. "Adapting ada architectural design knowledge to product design: Groundwork for a function based approach". In ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, pp. 185–200.

[29] Helmreich, R. L., and Merritt, A. C., 1998. "Culture at work in aviation and medicine". *National, organizational and professional influences. Hampshire, UK: Ashgate*.

[30] Konz, S., and Johnson, S., 2004. *Work design: occupational ergonomics*. Holcomb Hathaway, Publishers.

[31] Wickens, C. D., Gordon, S. E., Liu, Y., and Lee, J., 1998. "An introduction to human factors engineering".

[32] Bohm, M. R., Haapala, K. R., Poppa, K., Stone, R. B., and Tumer, I. Y., 2010. "Integrating life cycle assessment into the conceptual phase of design using a design repository". *Journal of Mechanical Design,* ***132***(9), p. 091005.

[33] Bohm, M. R., Stone, R. B., and Szykman, S., 2005. "Enhancing virtual product representations for advanced design repository systems". *Journal of Computing and Information Science in Engineering,* ***5***(4), pp. 360–372.

[34] Bohm, M. R., Vucovich, J. P., and Stone, R. B., 2008. "Using a design repository to drive concept generation". *Journal of Computing and Information Science in Engineering,* ***8***(1), p. 014502.

[35] Stone, R. B., Wood, K. L., and Crawford, R. H., 2000. "Using quantitative functional models to develop product architectures". *Design Studies,* ***21***(3), pp. 239–260.

Appendix A: Generic Human Error Database

TABLE 9. HUMAN GENERIC ERRORS ASSOCIATED WITH THE HUMAN TASKS (ICF) AND SOME INTERACTION FALLIBILITIES THAT CAN CONTRIBUTE TO THEM

Verb	Object	Interpretation for activity (task) modeling	Generic human error	Interaction Fallibilities
Reaching	Location	Reach out location	Fail to reach from fixed location Fail to reach to general location Fail to position body	Limited accessibility
	Object	Reach out or extend outwards to position an object using hands	Fail to reach specific target Fail to reach small object	Obstruction, limited accessibility
Picking up	Object	Pick up hand held products	Fail to pick up object	Object size, weight, surface conditions, speed-accuracy trade-off
Releasing	Object	Release hand held objects	Fail to release object	Object size, shape, surface conditions
Grasping	Object	Hold an object firmly in hand for required operation	Fail to grasp target object Failure to grasp sliding object Fail to re-grasp object	Object size, shape, surface conditions
Manipulating	Object	Complex hand activities that requires manipulation with fingers	Fail to manipulate	Object size, weight, shape, surface conditions
Pushing	Object	Pushing with finger, arm, hand	Fail to apply force Fail to apply pressure	Object shape, weight, surface conditions force required
Pulling	Object	Pulling with finger, arm, hand	Fail to apply force Fail to apply pressure	Object shape, weight, surface conditions force required
Turning	Object	Rotate something with hand	Fail to turn to desired location	Object shape, weight, surface conditions force required
Carrying (Moving)	Object	For importing and positioning an object	Fail to move to object indefinite location Fail to transfer grasped object to other hand Fail to move object to exact location Fail to transfer grasped object	Limited accessibility, object size, weight, shape, surface conditions

Appendix B: Composite Function - User Error Interaction matrix [FUIE] Table

TABLE 10: COMPOSITE FUNCTION-USER ERROR INTERACTION MATRIX [FUIE]

	Generic human error - Fail to:															
Product Functions	Reach specific target	Reach small object	Pickup	Release	Grasp target object	Grasp sliding object	Re-Grasp object	Manipulate	Apply force	Apply pressure	Move to indefinite location	Transfer object to other hand	Move to exact location	Transfer grasped object	Position body	Turn to desired location
Actuate Electric Energy	1	0	0	1	1	2	2	3	2	1	2	3	3	3	0	1
Convert Human Energy to Mechanical Energy	1	0	0	1	1	2	2	5	4	2	3	4	4	4	1	2
Convert Human Energy to Control Signal	1	0	0	1	1	2	2	5	4	2	3	4	4	4	1	2
Couple Solid	0	0	0	0	1	2	2	1	1	0	2	3	3	3	0	1
Export Human	0	0	0	1	0	0	0	1	0	0	1	1	1	1	1	0
Export Human Energy	0	0	0	1	0	0	0	1	0	0	1	1	1	1	1	0
Export Solid: Hand	0	0	0	1	0	1	1	2	1	0	3	3	3	3	0	1
Guide Human Material	0	0	0	0	1	2	2	1	1	0	2	3	3	3	0	1
Guide Solid: Hand	0	0	0	0	1	2	2	1	1	0	2	3	3	3	0	1
Guide Solid: Screws	0	0	0	1	1	2	2	2	1	0	3	4	4	4	0	1
Guide Weight	1	0	0	1	1	2	2	3	2	1	3	4	4	4	0	1
Import Control Signal	2	1	0	1	1	2	2	5	4	2	2	3	3	3	2	2
Import Human Material	1	1	1	0	0	0	0	0	0	0	0	0	0	0	3	0
Import Human Energy	1	1	1	0	0	0	0	0	0	0	0	0	0	0	3	0
Import Human Force	1	1	1	0	0	0	0	0	0	0	0	0	0	0	2	0
Import Solid	1	1	1	0	0	0	0	0	0	0	0	0	0	0	2	0
Import Solid: Hand	1	1	1	0	0	0	0	0	0	0	0	0	0	0	2	0
Import On/Off	1	0	0	1	1	2	2	3	2	1	2	3	3	3	0	1
Position Human	0	0	0	0	1	2	2	1	1	0	1	2	2	2	1	1
Position Human Material	0	0	0	0	1	2	2	1	1	0	2	3	3	3	0	1
Regulate Electric Energy	1	0	0	1	1	2	2	3	2	1	2	3	3	3	0	1
Secure Solid	0	0	0	0	1	2	2	1	1	0	2	3	3	3	0	1
Separate Solid	0	0	0	1	1	2	2	2	1	0	3	4	4	4	0	1
Stabilize Human	0	0	0	0	1	2	2	1	1	0	1	2	2	2	1	1
Stop Mechanical Energy	2	1	0	1	0	1	1	5	4	2	2	2	2	2	2	2
Support Human Material	0	0	0	0	1	2	2	1	1	0	2	3	3	3	0	1
Transfer Human Material	0	0	0	1	0	0	0	1	0	0	1	1	1	1	1	0
Transfer Human Energy	0	0	0	1	1	2	2	2	1	0	2	3	3	3	0	1
Transmit Human Force	1	0	0	1	1	2	2	3	2	1	3	4	4	4	0	1