CAN DESIGN EVALUATION TOOLS PREDICT/PREVENT CHANGE PROPAGATION?

Prabhu Shankar  
Department of Mechanical Engineering  
Clemson University  
pshanka@clemson.edu

James Mathieson  
Raveesj Ramachandran  
Joshua D. Summers  
Gregory M. Mocko  
Department of Mechanical Engineering  
Clemson University, USA  
jmathie@clemson.edu, raveesr@clemson.edu, jsummer@clemson.edu, gmocko@clemson.edu

Department of Mechanical Engineering  
Clemson University
ABSTRACT

This paper presents requirements for a change propagation tool and reviews existing design evaluation tools against them for their suitability and limitations. The expensive and time consuming design changes occurring at the later stages of the design due to change propagation motivated this research with an overarching goal of addressing the identified limitations. Fifteen design tools are identified and evaluated in how well they meet the defined requirements through a coding scheme focused on the input and output information. Function failure identification and propagation (FFIP), advanced failure mode and effect analysis (AFMEA), and change prediction method (CPM), meet the requirements partially, yet presenting promise for further enhancements.

KEYWORDS

Change propagation, failure analysis tools, change propagation tools, design change, engineering change

1. DESIGN CHANGE

Products are changed continuously to improve their performance, address field problems, or to incorporate new technological aspects into the system. The effect of these changes can propagate to other sections of the design in a complex system and cause an unforeseen violation of the acceptable performance limits. This phenomenon leads to time consuming and expensive changes after the product has entered the manufacturing phases of the product lifecycle [1]. In some instances, this can lead to product recalls. Thus, it is necessary to reduce this effect by developing and applying design tools for predicting change propagation, which is the overarching goal of this research. This paper aims to understand the limitations of the existing design tools both used for predicting change propagation and evaluating designs through literature survey, which will form the basis for the development of future design tools. This paper presents the requirements for a change propagation tool, a protocol for reviewing the design tools, and review of the design tools in this order.

1.1. PROBLEM OF CHANGE PROPAGATION

Higher-order functions, beyond those of individual elements, are achieved in a complex system through interactions between system elements. The interconnections within the system establish pathways between different sections of the design. These pathways form the backbone for propagating the effects of an engineering change (EC).

Performing an EC in a complex system can instigate a chain of propagations, thereby affecting the elements. When this propagation has a degrading effect in one or more of the system requirements, product failures can result thereby requiring further change. This phenomenon of one change leading to another is termed “change propagation”. Thus, the challenge in this problem is to identify and evaluate the different propagation pathways initiated during an EC. Such initiation is dependent on change modes, which are discussed in the next section.

1.2. propagation and its pathway: dependency on change mode

The phenomenon of propagation and its associated pathways are dependent on change modes; that is, what type of change is implemented in the product. A change mode could be an alteration in a physical quantity, a change in a part property, or the addition and/or removal of a part to the product. These change modes can be classified into two types: (i) An active change mode, one that initiates propagation; and (ii) An inactive change mode, one that does not. A change in the color of the hand drill’s battery, for example, from black to blue may not initiate propagation - an inactive change mode - although a change in the battery’s voltage will initiate - an active change mode. Part addition/deletion; feature addition, deletion, or modification; part orientation/translation; material change; and document changes are all examples of types of change modes.

In the event of an active change mode, the effect can travel through several different pathways. This can be visualized as arcs connecting nodes in a graph (Figure 1). A series of these arcs forms the propagation chain. Recent industrial case studies have suggested that propagation pathways are not restricted within the interconnected system elements but extends between two different interconnected departments of an organization [2,3]. This concept is illustrated using Figure 1 in which the nodes A, BI, and B2 represent system elements, BI and B2 being product variants, while node C represents a department in the organization. A change in the element A can propagate to node C through α, β, and γ pathways for one system configuration whereas in another instance node C can be affected via α1, β1, and δ pathways. Thus, the propagation chain in one configuration could be A- α- BI - γ - C, A- β- BI - γ - C or A- α and β- BI - γ - C. However, there can be other pathways between two nodes unlike the two shown in the illustration: (i) ambient pathway such as liquid and air; (ii) behavior pathway such as energy, material, information, spatial; (iii) configuration pathway; (iv) geometric pathway such as size relationship; and (v) organizational
pathway such as design affecting manufacturing and design affecting logistics.

![Schematics of propagation pathway](image)

**Figure 1:** Schematics of propagation pathway

The selection of these pathways, in turn, also depends on the change mode. An alteration of the electrical energy in a hand drill's battery, for example, can change the quantity of the input current to the motor via interconnected system elements. This differential physical quantity, due to change, may necessitate a change in the motor. Thus, this change mode has initiated propagation through a behavioral pathway. In another instance, the exterior color of the hand drill’s box may be changed from black to green for ease of identification in the assembly line, which may cause corrosion due to environment interaction; in this case, change initiates an ambient pathway.

In addition, an active change mode can invoke single or multiple pathways at the onset of propagation such as A- α -B1 or A- β -B1. Hence, design tools used for predicting change propagation must identify the change modes, must identify their associated propagation pathways, must identify the propagation chain, and must evaluate propagation effects. The five specific requirements for tools to manage change propagation are found in Table 1.

**Table 1** Change propagation tools’ requirements

<table>
<thead>
<tr>
<th>Requirements index</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1.1</td>
<td>Identify change modes</td>
</tr>
<tr>
<td>R1.2</td>
<td>Identify different type of pathways (ambient, behavior, geometric, organizational, and variant)</td>
</tr>
<tr>
<td>R1.3</td>
<td>Identify propagation chain</td>
</tr>
<tr>
<td>R1.4</td>
<td>Evaluate effects of propagation in the chain</td>
</tr>
<tr>
<td>R1.5</td>
<td>Suitable for use in the production phase</td>
</tr>
</tbody>
</table>

1. **design tools’ review protocol**

Design tools capable of predicting change propagation are the primary focus of this review. However, failure analysis tools are also included because a failure mode can result from a change and can lead to subsequent ECs. The essential information considered in the design tools is: (i) whether or not propagation chain is predicted/identified; (ii) If it is a failure analysis tool, whether or not interaction-based failure modes are predicted/identified. The rationale is such failure type can implicitly address propagation effects. The coding scheme is divided into three steps, found in Table 2, Table 3, and Table 4.

**Table 2:** Identification of the input information

<table>
<thead>
<tr>
<th>Tools</th>
<th>Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMEA&lt;sup&gt;1&lt;/sup&gt;</td>
<td>✓</td>
</tr>
<tr>
<td>CPM&lt;sup&gt;2&lt;/sup&gt;</td>
<td>✓</td>
</tr>
</tbody>
</table>

1. FMEA – Failure mode and effect analysis
2. CPM – Change prediction method

First, the initial information to the design tools, as shown in Table 2, are reviewed to verify the first requirement, that is, whether or not change modes are analyzed. Second, the type of data and reasoning scheme that is used in conjunction with the initial information to obtain the output information is captured. As shown in Table 3, column (a) shows what is identified with the initial information; Column (c) shows two different means, that is, historical and model data that are used to identify items in column (a). The sources for historical data includes documents, human memory, and codified database while model-based include contains two parts: first, what information is contained within the structure of the model, and second, in which environment the behavior is studied. Interactions being considered as a main contributor to induce propagation effects in the system, modeling them is essential to identify propagation paths or to identify failure modes [4,5]. Thus, for those tools where interactions are modeled column (d) identifies what is modeled - interaction-interrelationship or interaction effect? Column (b) identifies what reasoning scheme is used: abductive or deductive? It is included to identify the possibility of automating the tool. For instance, most computer-aided design tools uses deductive reasoning scheme [6]. Therefore, if any of the tools use such reasoning scheme then it provides a window of opportunity for automation. Third, the output information is listed (see in Table 4), which could be the same as items in column (a) in certain instances.
Table 3: Input-output information of the design tools

<table>
<thead>
<tr>
<th>Tools</th>
<th>Failure modes</th>
<th>Propagation pathway</th>
<th>Abruptive</th>
<th>Deliberate</th>
<th>Document</th>
<th>Historic</th>
<th>Model</th>
<th>Interaction effect</th>
<th>Interaction interrelationship</th>
<th>Abstract (Ab) / Geometrical (G) connection</th>
<th>Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMEA</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>Ab</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPM</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>G</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Tools that use interaction based model data are further analyzed to determine their limitations in terms of their model and the predicted pathways. The entries in Table 2, Table 3, and Table 4 are for illustration purpose. Readers can refer to the completed list of tables from Appendix I through Appendix III.

Table 4: Identification of output information from the design tools

<table>
<thead>
<tr>
<th>Tools</th>
<th>Design controls</th>
<th>Failure risk</th>
<th>Change risk</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMEA</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>CPM</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

2. REVIEW OF THE DESIGN TOOLS

This section reviews the design tools, as listed in Appendix-I. In this section, significant portion of time is spent on discussing FMEA and their limitations because it is a widely used and accepted tool for preventing failures and its fundamental concepts are used in different other tools.

2.1. Family of FMEA

Failure mode and effect analysis (FMEA) is a design tool used extensively in industries to predict the failure modes, to develop preventive measures in the design, and to prioritize the failures based on its risk [7,8].

As per SAE J 1739 standard [9], FMEA is initially conducted at the system level and continued for other lower levels in the system hierarchy. At each level, the functions are identified by a cross functional team of engineers. A component block diagram (CBD) of the system with appropriate resolution of the system is used (depending on the level of FMEA) to facilitate function identification. The manner in which these functions fail, termed as failure modes, are identified based on the team’s prior experience. The possible cause and effect of these failures are analyzed among the team. Finally, the failures are rated by a metric called Risk Priority Number (RPN) that accounts for the likelihood of each failure mode, their impact, and the ability to detect it. The prediction of failure mode, however, is not based on any systematic process that considers element interactions rather it relies on the team’s experience [10-12]. This factor combined with the fact that the change mode is not analyzed renders the tool not suitable for predicting change propagation.

How good are the inputs?

As per SAE’s recommended practice for FMEA, a component block diagram is used, which is analogous to function structure, for analyzing the components’ functions within the system boundary. Blocks represent the components while the interconnections between them are system parameters such as energy, material, and information. Both the components and their associated functions are listed by the designer in the recommended documentation template. Subsequently, the failure modes are predicted for these functions. The question posed here is: How effective will be the system description?

Technical systems are represented at the highest level of abstraction, that is functions, for distancing designers from the physical embodiment to escape fixation [13]. This approach is beneficial for synthesizing design concepts, but it may not be suitable for analyzing existing technical systems [14] for the following reasons:

(i) Design cognitive standpoint

Designer finds it difficult to consciously disassociate themselves from the physical embodiment and think what the design should do in abstract terms, such as function, because the functional properties of the system are stored in the designer’s memory with close association to the physical embodiments [15]. After switching to abstract terms, the inter relationship between the abstracted details of the components are dispersed in the designer’s mind, which reduces the memory’s retrieval capacity as the abstract-concrete relationship strength is weakened [16]. As a result, a complete list of functions may not be retrieved even from an experienced designer’s mind [14].
(ii) Perspective change

Designers may end up listing fewer functions than what actually exists because of switching from concrete to abstract representation. In abstract terms, an input shaft’s functionality in a motor is ‘to transmit power’ whereas the other features in it that supports manufacturing, such as centering feature, may not necessarily be identified in the analysis.

(iii) Notion of function

Designers with similar background have different notions for functions at different levels of abstraction in their mind, which can lead to different degrees of interpretation of the existing system [14]. These differences can also lead to fewer functions identified in the system than what actually exists.

These reasons can lead to incomplete description of the system’s functionalities, which is highlighted by other researchers [17, 18].

Notion of functional dependency on single components

Functionality of the system is associated with the product and each component in it has associated functions [19]. However, functions in technical systems are achieved by interaction between components [20], which is not considered in the existing FMEA method. It is assumed that each component will realize its associated function independently. This notion, not limited to, prevent designers from identifying failures that may arise from interactions. Such inability to identify interaction-based failures has been pointed out by several researchers [17, 21].

Functional propagation idea

A top-down approach is followed during FMEA, that is, starting from system, sub-system to component level. The failure mechanism predicted at the system level, for each identified function, is carried forward as a potential failure mode at the sub-system level, and the process is continued until the last level. This hierarchical transformation can help identify functional-failure propagation (See Appendix IV for an example on a brake system). Therefore, interaction based failures can be prevented with the use of FMEA, as suggested by [22], only if the failure mechanism is identified based on the interactions at the system level. Two limitations reduce the value of this hierarchical transformation idea built in the method:

(i) The association between functions at different levels of hierarchy is not traceable, as the existing documentation scheme does not have explicit provisions referring to affected upstream and downstream functions. As a result, designers may

 loose track of the association between them when performing FMEA for large technical system.

(ii) The identification process of failure mechanisms is based on the cross functional team’s experience rather than a systematic process. Hence, the prediction is dependent on the team’s best engineering judgment based on prior experience, which may not necessarily include interaction-based failures. The issue in relying on designer’s memory is discussed in the following section.

Reliance on expertise for failure mode identification

The prediction of failure modes for each identified function is based on the team’s knowledge of their prior experience, which limits the prediction within their domain knowledge [17]. There can be failure modes for similar functions in other domains that may be eliminated from the analysis. In addition, the use of expert domain knowledge leads to the following questions:

(iii) Can the degree of engineers’ expertise hinder the recollection ability of the failure modes?

(iv) How suitable will be their recollection to the problem under investigation?

(v) How many years of memory can be traced back to recollect the failure instances and their associated failure modes?

(vi) How reliable will be their recollection of thoughts?

The first two questions can be answered to certain degree from cognitive studies. Experts do posses a greater stock of relevant designs in their mind, but they will have difficulty in escaping the similar current situations because of the stronger situation-action associations [23]. This is a positive sign because designers may get cues about the failure modes from the past that they may have experienced. How suitable will be their recollection to the situation depends upon what type of expert are they. Two types of domain experts exist: Inflexible and flexible; Inflexible experts exhibit mental set and fixation while flexible experts do not exhibit such psychological effects in thinking [24]. Even though both experts possess rich association between their factual and proceduralized domain knowledge, the inflexible experts will have trouble escaping the embodied tacit constraints experienced previously [25, 26]. This means that inflexible experts may predict failure modes that may no longer be valid for the present situation. Although inferences from the past design-cognitive studies can be used to answer the above questions to a certain extent, further research is required on these aspects. For the third and fourth
questions, it presents a window of opportunity for future research.

Change modes are not documented in the existing format of FMEA. Therefore, designers while predicting the failure mode may overlook the possible failures due to the change mode during their recollection process. Currently, they may implicitly think about the failure modes based on change mode, for which no documented evidence is yet available.

Why cannot FMEA identify interaction-based failures?

As component block diagram (CBD) models interaction inter-relationship, it has been used in industry to conduct cause-effect analysis for identifying failure modes. The limitation in the representation of the CBD prevents causal reasoning of the interaction effects, which is illustrated using a portion of the CBD, as shown in Figure 4, developed for a hydraulic brake system in passenger cars. Brake system circuit and its components functionality are identified from [27].

During braking application by the vehicle driver, the brake pedal actuates the servo (otherwise called as brake booster) that magnifies the input pedal force to the required level. This servo performs the force magnification action with vacuum assistance from a vacuum pump. For the purpose of illustration, it is assumed that FMEA is conducted on the component ‘servo’. As per the standards, the component and its item function (in this case ‘magnify force’) is written in the document. This component receives an energy input from the pedal (ME1), and energy (ME4) and a material input (V1) from the vacuum pump, magnifies the input, and outputs a magnified energy (ME5). The next step is to identify the failure modes. The designer examines the input lines and starts the analysis by removing the input energy line from the vacuum pump, as shown in Figure 2 with red crossed symbol.

Figure 2 Servo and vacuum pump portion of CBD

From this representation, what can be inferred is servo produces a magnified energy (ME5) with one material V1 and one energy input ME2. In reality, this doesn’t make any sense because if there is a loss of energy (ME4) from the vacuum pump then there is an associated loss of material transfer, that is, vacuum (V1). Having recognized this limitation both the material and energy output from the vacuum pump to the servo is removed. Does it make sense now? No, because the diagram shows that ME5 is produced with ME2 without any vacuum assistance, which is not possible. ME5 from the servo is possible if and only if both ME4 and V1 are present in conjunction with ME2. The loss of function ‘magnify force’ due to the loss of vacuum is not reflected due to the lack of representation of dependencies and the logical relationships between the function flow lines. Therefore, this representation does not support causal reasoning of the interaction effects even at a qualitative level.

Change in state of the system can cause a change in the routing of the flow lines, which is not captured in CBD. The portion of the CBD with the component ‘valve’ is specifically examined to illustrate this limitation. As shown in Figure 3, the pressurized brake fluid is distributed to the rear brakes via a pressure-reducing valve. This is essentially included in the circuit to avoid rear wheel locking and vehicle instability only in the vehicle-unladen state (driver only condition). When the vehicle changes from unladen to laden state, the input flow to the function ‘reduce pressure’ may be bypassed and directly distributed to the rear wheels through the distributor. In certain cases, the function ‘reduce pressure’ is required on both laden and unladen system states, but the point of initiation of this function is directly dependent on the braking dynamics, specifically, the deceleration point. Therefore, representing a function that is dependent on the dynamic state of the system as a quasi-static state independent function limits the correctness of the interaction analysis.

To further illustrate the limitation in CBD representation, a parking brake circuit is also added, as shown in Figure 5. The components, rear brake right and left, will now receive an additional energy flow. The question here is: Does the rear brakes receive energy from the parking lever at all times during the vehicle operation? No, it receives energy from the parking brake only while the vehicle is parked and, occasionally, from both the parking brake circuit and the main circuit during an emergency. Hence, the function’s input dependency on the operation mode is not represented.

Figure 3 Valve portion of the CBD

Figure 4 Servo and vacuum pump portion of CBD
Figure 4: CBD for a hydraulic brake system

As identified above, CBD is independent of the operation modes of the system. In Figure 4, the flows during the ‘apply brake’ mode are represented. What will be the flow interaction during the ‘release brake’ mode? In rear brakes, the function ‘apply force’ and ‘absorb kinetic energy’ obviously vanishes and the function ‘release force’ is not even modeled. The inclusion/deletion of functions under different operation modes can have different interaction flows, which cannot be identified with this representation. If modeled then it will lead to the issues discussed in the above paragraph.

Figure 5 CBD with parking brake

The three major limitations in CBD representation: (i) exclusion of flow line dependency and their logical inter-relationships in the model; (ii) representing functions as a quasi-static state independent functions; (iii) modeling flows and functions as if they are independent of the operation mode, prevents designers from identifying the interaction effects and their associated failures.

CBD models only the interaction-relationships between the components, not the effects due to their behavior. As interaction-based failures result due to their interaction effects, it can be stated that these failures cannot be identified in FMEA using CBDs.

Lack of clarity in the definition of terms

The terms used in FMEA are not well defined [28,29]. The term failure is not defined in the SAE J 1739 standard. A function can either loose its functionality or degrade. Which one should be considered for analysis – loss of functionality or degradation? The second definition that is vague is ‘detection’. As per [9] standard, the definition is:

‘Detection is an assessment of the ability to prevent the cause/mechanism or failure mode/effect from occurring, or reduce their rate of occurrence. Detect the cause/mechanism and lead to corrective actions, and detect the failure mode.’
This definition spans both department in a manufacturing firm – design and manufacturing. One of the definition rates the quality process while the others rate the detection ability in product development process. It is also not clear if this ability to detect is during the product development process or during the operation of the product. This has led to confusion in the designer’s mind [22].

Apart from the ones mentioned above users feel it is tedious and time consuming [7,11,18,29]; it is often used to satisfy quality audit purpose; it has vague representation of functions, failure modes, failure effects, failure causes, and detection controls; it has inconsistent formats across the supply chain, and lack of integrity of documents across the supply chain [8,22].

2.2. Failure mode and effect criticality analysis (FMECA)

FMECA is an improved FMEA method that specifically addresses the difficulties in ranking failure modes. In FMEA, failure modes are prioritized using RPN, which can be misleading [30]. For example, let two failure modes have same RPN of 120. Does it mean that both these failure modes carry the same rank? How to pick which one is more important? To address this ambiguity, criticality of the effect is determined as a mathematical function of both occurrence probability and the severity, which can be subsequently used to prioritize the failure modes. Apart from this difference, FMECA follows FMEA and its limitations.

2.3. FMEA for modification

Neglecting change mode in FMEA prevents designer’s from focusing on the possible failure modes due to a design change [12], which is one of the limitation of traditional FMEA. Therefore, the documentation template has been modified to account for the change mode, which enables explicit thinking of the failure modes specific to changes but not due to interactions.

2.4. Advanced Failure Mode and Effect Analysis

Advanced failure mode and effect analysis (AFMEA) is a semi-automated FMEA used to enhance product reliability at the conceptual design stage [31,32]. The process begins with identifying key characteristic requirements from quality function deployment (QFD) that the system should meet. A function-behavior-structure (F-B-S) model is built to develop the knowledge of the product – an adopted approach from [33]. In this modeling philosophy, functions are what the product is intended to do while behaviors are how these functions are achieved, which indirectly refers to the individual components and their interactions. Behaviors are represented as a sequence of transitions of partial states. Readers may consult [32-34] for further details on behavior modeling.

The key function’s behavior is decomposed into a sequence of partial transitional behavioral states at various levels of hierarchy. This behavior tree is mapped to the parts of the component in an assembly, thereby achieving a function-behavior-structure mapping. At each level of the behavior hierarchical tree, every behavior is qualitatively assigned the required initial and final states. The developed behavior model is used for simulation to identify failure modes, which is defined as the inability to transit from initial or intermediate state to the final state or to an undesired state. Three different types of failures are identified: (i) nonbehavior (loss of behavior); (ii) misbehavior (unintended behavior); and (iii) undesired behavior (degraded behavior).

The failure modes are predicted by selecting a behavior candidate and assuming it to fail in the computer simulation. The resulting behaviors are compared against its desired state to identify any possible failures. Subsequently, these failure modes are entered manually in the standard FMEA template for further evaluation.

Since the failure modes are predicted based on behavior simulation, interaction-based failures can be predicted using this method. However, simulation is not based on the specific change mode rather it is a random choice by the user and it does not intent to identify ambient, variant, geometrical, and organizational pathways. Apart from this limitation, this tool is a potential candidate for predicting change propagation at the conceptual design phase.

2.5. Scenario based Failure Mode and Effect Analysis (SFMEA)

Scenario based FMEA is focused on: (i) risk evaluation of each possible scenario of an identified failure; and (ii) prioritizing the failure scenarios based on their likelihood of occurrence and their cost impact [35].
As per SAE J 1739 standard, no explicit reference is made to consider multiple scenarios for each functional failure. In SFMEA, this limitation is addressed by explicitly analyzing multiple scenarios of failure. A scenario is a path, like a-b-c, in the function-failure mode-cause-effect tree, as shown in Figure 6.

In the context of this paper, this method carries forward the limitation of the traditional FMEA. Therefore, this FMEA technique cannot analyze change modes and their associated interaction-based failures.

![Figure 6: Scenario trees](image)

### 2.6. Conceptual Failure Mode Analysis (CFMA)

Conceptual failure mode analysis (CFMA) is another failure analysis tool with focus on concept design evaluation [36]. The process begins with the functional analysis using a function analysis system technique (FAST) diagram [37]. The failure modes for these functions are identified by the product development team based on prior experience. Later, the causes and effects of these failure modes are identified by the team. The product of O, S, and D is used as a metric to prioritize the failures, which is similar to RPN in traditional FMEA. The occurrence (O), severity (S), and the detection (D) of these failures are measured in a non-linear scale of 1, 2, 4, and 10 unlike the linear scale used in traditional FMEA. Apart from the difference in functional analysis procedure and the rating scale of the failures, this tool is fundamentally not different from FMEA. Therefore, their limitations are equally applicable in this tool too.

### 2.7. WIFA

WIFA is a German acronym for knowledge-based FMEA [18]. Three items that stored in the database are: (i) historic FMEAs for easy access to the designers when they conduct new FMEAs; (ii) product descriptions such as system and function taxonomy, function failures, and failure modes; (iii) it stores the sequence of steps required for conducting FMEA. With this information, this technique aims to improve the FMEA quality. However, this carries forward FMEA’s limitations and hence this tool is rejected from further analysis.

### 2.8. FFDM

Function failure design method (FFDM) is an approach that couples failure analysis at the conceptual stage of the design with a goal of reducing failures at later stages of the design [11,17]. This method uses a matrix-based approach to populate the function-failure knowledge base in which designers identify the relationship between function-components (EC) and components-failure (CF). Using simple matrix multiplication, the relationship between function-failure (EF) is identified. The rows in this matrix represent functions and the entries in the columns corresponding to that row represent the number of components satisfying the function failed by the failure mode. Using this database, designers can analyze the possible failure modes for associated functions early in the design, and thereby, reduce the design changes at a later stage.

The primary advantage of using this method is moving away from the reliance on designer’s memory on the past failure modes, which is a major drawback in FMEA. Nevertheless, this tool doesn’t meet the requirements for the change propagation tool because:

- (i) Change modes are not analyzed;
- (ii) Interaction models are not developed to study the effect of change in functionality and how that impacts others.

### 2.9. Fault Tree Analysis (FTA)

Fault tree analysis (FTA) is one of the most widely used techniques for studying system reliability and safety. In this technique, a foreseeable undesired scenario of high risk is considered for analysis, such as unexpected loss of brake fluid in a hydraulic brake system of a passenger car, and then analysis is conducted on the system to identify the basic events, the causes, that will result in the occurrence of this event [38,39]. The basic events could be component hardware failure, human errors, environmental conditions, or any other reason. The sequence of these events leading to the described high-risk failure
is represented in a graphical representation using logic gates (see [40] for more information on gates). Such logical interconnections are built through system analyses using functional block diagram [41]. Probability of failure is assigned to each of the failure events based on historical data, and the failure probability of the top-level event is determined. In addition, sensitivity analysis is also performed to identify the sub-events that act as significant contributors.

FTA uses abductive reasoning to identify different events that may lead to a failure state. It is continued to a point where failures cannot be further decomposed. Each branch in the tree represents a failure path. Since FBDs do not support causal reasoning, failure paths identified may be incorrect.

FTA does not meet the criteria for a change propagation tool as change modes and their associated propagation effects, specifically performance degradation, are not analyzed [42]. Implementation wise, it is time consuming and expensive to construct [43].

### 2.10. Event Tree Analysis (ETA)

Event tree analysis (ETA) is used to identify the consequences of the occurrence of a potentially hazardous event [44]. This is in contrast with FTA where causes are identified for a hazardous event [45]. ETA begins with identification of an initiating event, such as fluid leakage in the brake’s master cylinder, and then identifying the consequences of this event in different system elements using function block diagrams. Different outcomes are identified by postulating a success or a failure of all the connected system elements. At each branch point, the failure probability of success (S) and failure (F) is determined using FTA. A simple event tree analysis of a brake fluid leak from a master cylinder is shown in Figure 7. Each branch in this diagram is a functional failure propagation path.

![Figure 7 ETA example on a brake system](image)

Two major limitations of this tool are: (i) consequential effects are identified using FBDs that have limited capabilities to support causal reasoning; hence, the path identified may be incorrect; (ii) using fault trees in this tool carries forward its limitation too, specifically, lack of change mode and performance degradation analyses.

### 2.11. Probabilistic Risk Assessment (PRA)

Probabilistic risk assessment (PRA) is used to predict the risk of a system failure based on the impact of the resulting hazardous event. It extends the ETA and FTA model, as shown in Figure 8, to predict the risk of failure scenarios by multiplying the failure probability, hazardous event’s occurrence rate, and its consequence. The major outcome is prioritizing the system elements based on their degree of contribution to the failure risk and measuring the uncertainties of the contribution values. An exhaustive and systematic procedure is followed for conducting PRA [46]. The important output to this tool is identifying elements that contribute most to the failure risk using sensitivity analysis and ranking them using importance measures (See [47,48] for more details on importance measures). Limitations in this tool are same as what is identified in ETA and FTA, hence, not discussed in detail.

![Figure 8 PRA framework](image)

### 2.12. FFIP

Function failure identification and propagation (FFIP) is a framework to identify the functions that may fail due to propagation effect [21]. This framework, as shown in Figure 9, uses a low fidelity high level functional model to analyze the possible functional failures of the system using behavior models and an associated reasoning model. This framework is used as a pro-active tool to evaluate the propagation effects at the conceptual design stage.
when the system components and their design parameters are not yet known.

FFIP is built on a function-configuration-behavior architecture of a system at a high level of abstraction with a function failure logic (FFL) reasoner to support fault analysis and their consequences without relying on the expert opinion. The process of developing this framework begins with development of a function model using standardized taxonomy of functions. An abstract description of components for these functions are identified based on [49] and represented as a graph termed configuration flow graph (CFG). The functions in the function model are mapped to the components in the CFG. The third layer is the behavior model used in conjunction with CFG to conduct behavioral simulation. Each component's behavior is defined by their input and output relations, and their underlying physical principles. The behavior of each component can be viewed as an event-mode-behavior architecture where a user specified event triggers an operational mode that results in a behavior. The transition from one mode to the other is triggered by an event that can be due to environment interaction, due to human interaction, or due to component malfunction. During the modeling stage, the user prescribes the different conditions in which the intended functionality cannot be achieved. This logic of condition prescription is termed as function failure logic reasoner in the framework.

The computer-based simulation is initiated by initializing the state variables to their nominal modes and executed in a continuous time frame. The physical state of the system is fed to the reasoner where it evaluates the function failure logic for the received state at the end of each intermittent state of the system. After evaluation, it returns the information to the simulator for further simulation. User can initiate different failure events to determine what functions can fail due to propagation effects. The simulation will continue until a prescribed end state or for specific number of time steps.

This tool meets the change propagation tool requirement except that it is not specific to change mode and do not consider ambient, organization, variant and geometric pathway. Though it is limited to the conceptual stage of the design, this framework can be extended to later design stages by including the design parameters.

2.13. Change prediction method (CPM)

Change prediction method is a tool to predict the elements that are affected due to change propagation using probability-based risk analysis [50]. In this method, two matrices are developed based on the designer’s experience. In the first matrix, the likelihood of change in one component affecting the other is captured in a structural connectivity design structure matrix (DSM). In the second matrix, the degree of impact a change can have on the other component is captured. Since the values in the likelihood and impact matrix are valid only for the direct connections, a predictive model is developed to take into account the effects of a change on a component due to indirect relations. An average likelihood of change resulting from all different connections is computed. Subsequently, using probabilistic risk theory the risk and the impact of a change is predicted from the predicted average likelihood matrix. This risk matrix is further used to identify components affected in the chain of propagation through structural pathway.

There are three assumptions in this method. First, it is assumed that changes to an element can cause propagation irrespective of the type of change mode and whether or not it is active or inactive. The effect of not considering it in this tool encompasses a nature to predict elements that are not affected.

Second, this tool assumes the applicability of probability theory on the likelihood information elicited from the design engineers, which is captured in the likelihood matrix. Probability theory can be applied when the measured quantity’s uncertainty is non-deterministic and objective [51]. Thus, the fundamental question here: is the information captured in this tool deterministic or non-deterministic? In an electro-mechanical system,
which is governed by a set of physical principles, the effect of a change in an element on the other, in a specific change mode, is deterministic, but it may not be known to the designer due to the lack of knowledge for various reasons such as limited experience or limited modeling capabilities. A second question that follows the first is: On what basis does the engineers’ provide the likelihood numbers? Were they subjective or objective? It is understood from the literature that the numbers provided in the likelihood matrix are not based on the variability obtained from a repeated set of experiments under controlled conditions rather based on their best guess, which is subjective. Thus, the two requirements for the applicability of probability theory are not met; therefore, the risk prediction needs a different approach to model the uncertainty than what exists.

Third, the likelihood numbers provided by the designers assume that every change in a given component followed the same propagation pathway in the past, which is not necessarily true. Indeed, some change mode can initiate propagation through an entirely different pathway even when the component where the change initiation is the same. Therefore, the construction of the likelihood matrix lacks rigor, and hence, the risk predicted using this approach might not produce reliable results.

CPM does not consider ambient, organization, variant, and behavioral pathways. However, within its proposed context of identifying change risk, it can be enhanced further with suitable modifications that will address the limitations identified here.

2.14. Integrated CPM with channel and contact model (C&CM)

The motivation to integrate CPM with channel and contact model (C&CM) is to provide designer’s with two different perspectives of the product: (i) abstract level using functional representation; and (ii) concrete level [52]. This perspective switching between abstract to concrete representation of the product and vice-versa enables designers to solve problems more efficiently [13]. With two different representations, designers’ ability to understand the system is enhanced, and, with added product information, a better decision-making process to assess the change risk is possible.

A strategy to integrate CPM and C&CM, as shown in Figure 10, have been proposed in this modeling approach in order to estimate the change risk. In this strategy, a CPM model is developed using a high-level component design structure matrix (DSM). After identifying the high-risk connection path from this model, C&C M is applied to focus on the components in this path to have a more deterministic view of the change effects. The use of CPM implies that the limitations discussed in section 2.13 is applicable in this tool also. However, the change effects can be identified in the deterministic sense using C&CM method, which will be beneficial to the designer, provided the limitations in the CPM tool are addressed.

2.15. Requirement based change propagation

The effects of a change in a component on the system level requirements are studied using a matrix-based requirement-modeling scheme (RMS). Three different modeling schemes exist in the literature. First, a model mapping three information domains in the design: requirement to function, function to component, and component to engineering characteristics [53]. Second, this modeling scheme is extended to capture more design information by mapping seven different domains: requirements, functions, working principle, components, design parameters, tests, and test parameters [54]. Third, an eight-domain modeling scheme that includes non-functional requirements is proposed to address the limitation in the earlier modeling scheme [19]. Using any of these models, the components that may get affected due to a requirement change or vice-versa can be determined using a series of matrix multiplication.

The fundamental limitation in this model is it cannot identify propagation pathways, as the geometric interdependency between the components is not
modeled. It can only provide the set of components that participate in a function, not the propagation pathway. Hence, this tool is eliminated from further review.

3. CONCLUSION

This paper presented a detailed review of the different design evaluation tools that can reduce change propagation. A bi-level review of the design tools is conducted where, in the first level, those tools that cannot predict propagation pathways are eliminated while the rest are analyzed in detail at the second level. The filtered tools and the extent to which they meet the change propagation tool’s requirements are presented in Table 5. Evidently, CPM and its family is the only tool that has the potential to address propagation issues due to change. However, the use of probabilistic modeling approach to model subjective information is strongly opposed by subjectivist researchers in the uncertainty field. This presents an opportunity to research this tool with a new approach. Also, the tool is focused to predict geometric pathway of propagation. This limitation presents opportunity to extend these tools for identifying other types of propagation pathways. For instance, FMEA, if conducted systematically, has the ability to reduce the function failure propagation effects. The limitations in the interaction model representation, use of functional representations, notion of functional dependency on individual components, reliance on designer’s memory, lack of clarity in the definition of terms presents a myriad of problems to this tool. Despite these limitations, industry, especially automotive, widely uses this tool, thus, presenting a need to advance this tool. The framework of FFIP and AFMEA design tool, though tailored for conceptual design, can be extended to support propagation prediction during later stages of the design.

None of these tools, either individually or in combination, can identify the different propagation pathways for a specific change mode. Thus, the current research efforts are directed towards developing a comprehensive design tool from a verification and validation standpoint.

Table 5 Existing design tools ability to predict pathways

<table>
<thead>
<tr>
<th>Tools</th>
<th>Change modes</th>
<th>Production stage</th>
<th>Behavioral dependency pathway</th>
<th>Ambient pathway</th>
<th>Variant pathway</th>
<th>Organizational pathway</th>
<th>Propagation chain</th>
<th>Recapitulation of effects of propagation chain</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPM</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPM, C&amp;CM</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AFMEA</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FFIP</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4. REFERENCES


Conferences and Computers in Engineering Conference, ASME, Irvine, California.


