EVALUATING THE RISK OF CHANGE PROPAGATION

Arman Oduncuoglu[1] and Thomson, Vince[2]
(1) McGill University (2) McGill University

ABSTRACT

The ever changing trends in current markets along with customers' rising demands for quality require many companies to continuously develop new products. Many companies use iterative design to add new features to old products. The changes resulting from the iterative approach along with the usual changes demanded by customers have created a difficult environment to manage. In this environment, many changes drive other design changes. This paper develops a technique to evaluate the risk of change propagation by using functional analysis, domain mapping matrix (DMM), and component design structure matrix (C-DSM) methods. This technique obtains the change propagation risk for a conceptual design solution at a functional level and provides insight for future resource requirements (i.e., development effort, product cost, etc.). The objective of the technique is to increase product knowledge in the early stages of design, to provide insight on the effects of engineering change, and to support design engineers in decision making.

Keywords: Engineering Change Management, Change Propagation Risk, Quantitative Risk Assessment, FAST Diagram, Value Engineering, Product Development

1 INTRODUCTION

Ever increasing competitiveness along with fast changing environments in current markets force companies to seek ways to produce higher quality products at the lowest cost in the shortest amount of time. The ability to make changes to existing designs, in other words agility in product development and production, has become a crucial issue for many companies that want to be at the leading edge or just want to survive in this dynamic market. One of the most prominent ways for companies to remain competitive is by making changes to existing designs while avoiding time and cost overruns as well as maintaining quality.

In order to handle the risk generated by an uncertain context in large complex projects, companies have adopted 'incremental innovation', which gradually introduces new functions, performance characteristics and technologies that rely on existing product designs.

As a result, adaptive design, where existing products are modified to produce new solutions that satisfy changing needs, new requirements or the desire for improvement, constitutes 75 to 85% of new product development projects [1]. In order to efficiently and effectively deal with the introduction of a new product solution, it is paramount that the impact of engineering change (i.e., effort, span time, technical difficulty, quality, fulfillment of customer requirements, and cost) be identified and assessed as early as possible within the product life cycle.

On the other hand, evidence from empirical investigations [2] and from the literature [3] show that 70-80% of total product cost is decided during early design stages where 56% of changes occur after the initial phase, of which 39% are avoidable.

This paper outlines a method to evaluate the effect of engineering change in terms of propagation type by using functional analysis and the matrix based methods, domain mapping matrix (DMM) and component design structure matrix (C-DSM). The proposed technique provides insight into risk of engineering change propagation if the current design needs to be modified at some point in the future. With this analysis, engineers can better visualize how to modify their design to increase modularity and to avoid changes with high costs.

The remainder of this paper is organized as follows. Section 1 has defined the objectives and approach. Section 2 briefly provides the background information required to understand the proposed tool. Section 3 introduces the tool by means of a thermoflask example, which is followed by conclusions in Section 4.

2 BACKGROUND INFORMATION

In this section, a brief theoretical background is provided on the building blocks of the proposed tool. First, a brief description of Functional Performance Specification (FPS) is provided. Then, two concepts, risk and change propagation are introduced.

2.1 Functional Performance Specification

SAVE International (Society of American Value Engineers) defines value engineering as [4]:

- identifying the functions of a product or service
- establishing a monetary value for the functions, and
- providing the required functions at the lowest overall cost.

Functional Performance Specification (FPS) is a methodology used to detect, formulate and justify the functional requirements of a product or a service without specifying any solutions. It concentrates on the functions of a product rather than product parameters, and aims to satisfy these functional requirements with the lowest cost, effort, time, etc. The Functional Analysis System Technique (FAST) diagram (Figure 1) is one of the key functional performance specification methods. The FAST diagram allows a functional hierarchy to be developed from left to right where functions are broken down to simpler ones. At this point, specific devices that can perform the functions are given along with associated costs at the right of the diagram (Figure 1).

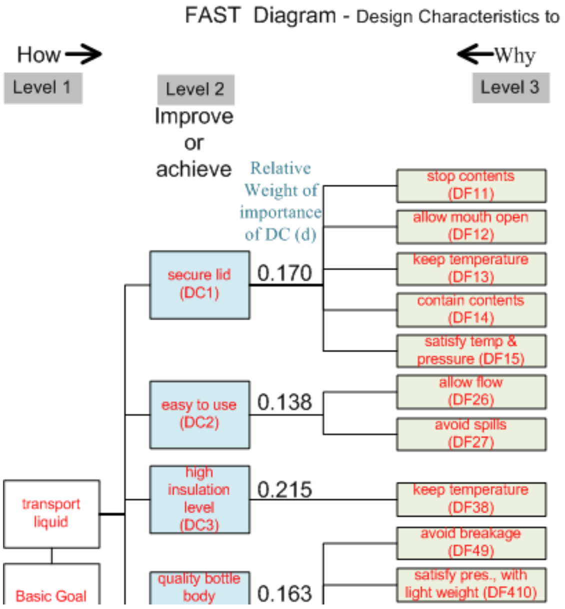


Figure 1. A section of a FAST diagram for a thermoflask

2.2 Risk

Risk can be defined as the “variation in outcomes around an expectation”; in other words, risk refers to how life differs from what is expected [5]. In addition, according to Wang and Roush [6], the central features of any engineering project are to produce a result that leads to customer satisfaction within the scheduled span time and budgeted cost. During an engineering design activity, risk includes late design changes, product defects, manufacturing variability, structural and technical failures (materials, tools, process procedures, installation specification), wear failures, special severe environments, human factors, etc., that can lead to customer dissatisfaction and to time and budget overruns. At the start of any project, there are uncertainties for each of these factors for which the associated risks should be managed. This kind of management can help to mitigate the potential of negative consequences arising from uncertainties, and maximize the possibilities that results will be better than target values.

2.3 Engineering Change Propagation

Most engineering change methods track change from design through manufacturing to minimize detrimental effects on products [7]. Reidelbach [8] states that controlling the effects of engineering change becomes a more crucial issue especially for products with long lead times such as airplanes or automobiles. In the field of software engineering, change models break a program down into pieces that are then linked into a propagation graph, and subsequently, immediate changes that might be necessary for software redesign are highlighted. Furthermore, the models rely on programming variables to indicate links, and predict only one step of change at a time [9-11].

The change propagation method (ChPM) [12] was chosen to model and analyze design change patterns. The ChPM was developed to determine the consequences of a change resulting from part faults or new requirements for complex mechanical designs. Using a model of the dependencies within and between functions, subsystems, and components, the ChPM yields a prediction and visualization of the components potentially affected by a change. The ChPM approach represents an advanced technique for change propagation. Nevertheless, it does not consider new patterns of change. Previously unanticipated changes cannot be systematically identified and analyzed. Below, two of the connectivity models, core elements of the ChPM, are introduced and explained in detail.

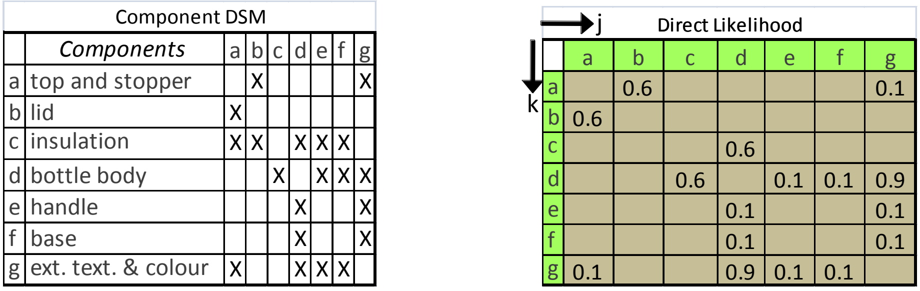
2.3.1 Connectivity Model

Domain Mapping Matrix (DMM) as Connectivity Model

The DMM is a rectangular matrix in which design intents from different domains such as organization, customer requirements, and processes can be linked with each other. The matrices used in the QFD including the HOQ matrix are among the most famous examples of DMMs [13]. In a DMM each row represents a design intent from one domain and each column represents a design intent from another domain. The cells of the matrix contain dependency values between each row and column.

The DMM method [14] can be used to represent inter-domain dependencies, or in other words, the mapping between design functions and subsystems, and the mapping of subsystems to critical component types. This is accomplished by taking selected design functions from the lowest level of the functional diagram and first bringing them into subsystem domain as "WHATs" of the DMM, whereas, the "Hows" of this DMM are the subsystems. Then, the subsystem domain can further be broken down into the component domain by repeating the same procedure.

Design Structure Matrix (DSM) as Connectivity Model



Component DSM

	Components	a	b	c	d	e	f	g
a	top and stopper		X					X
b	lid	X						
c	insulation	X	X		X	X	X	
d	bottle body			X		X	X	X
e	handle				X			X
f	base				X			X
g	ext. text. & colour	X			X	X	X	

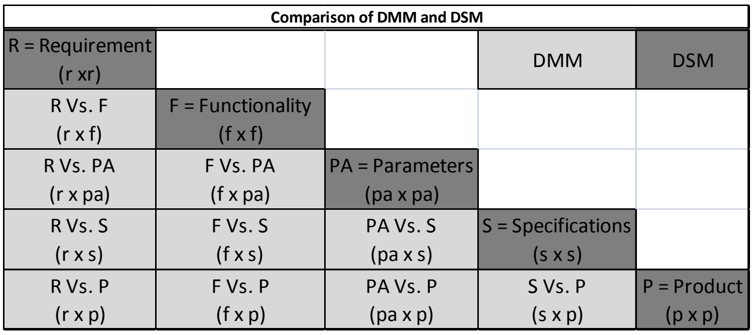
Direct Likelihood

	a	b	c	d	e	f	g
a		0.6					0.1
b	0.6						
c				0.6			
d			0.6		0.1	0.1	0.9
e				0.1			0.1
f				0.1			0.1
g	0.1			0.9	0.1	0.1	

Figure 2. Sample DSMs, where the matrix on the left shows dependencies between components in binary form. The matrix on the right shows the probability of change for a component, given that the corresponding component in the row changes

DSM is a matrix based approach which was originally introduced by Steward [15] to model dependencies or the information about dependencies between different design tasks. Since then, the use of the DSM has been extended to represent dependency links within other domains, such as resources, product components, etc.

Component Design Structure Matrices (C-DSMs) [16] are matrices in which rows and columns represent design components, and the cells in the matrix record connections between the corresponding elements. C-DSMs have been used to model component connectivity either by itself or in combination with a product's functional structure to illustrate the relationship between the two. Methods have been proposed to analyze a C-DSM to gain a better understanding of design [17] in order to estimate specific properties such as change propagation risk and behaviour [12], or the modularity of components [18] or subsystems [19].



Comparison of DMM and DSM				
R = Requirement (r xr)			DMM	DSM
R Vs. F (r x f)	F = Functionality (f x f)			
R Vs. PA (r x pa)	F Vs. PA (f x pa)	PA = Parameters (pa x pa)		
R Vs. S (r x s)	F Vs. S (f x s)	PA Vs. S (pa x s)	S = Specifications (s x s)	
R Vs. P (r x p)	F Vs. P (f x p)	PA Vs. P (pa x p)	S Vs. P (s x p)	P = Product (p x p)

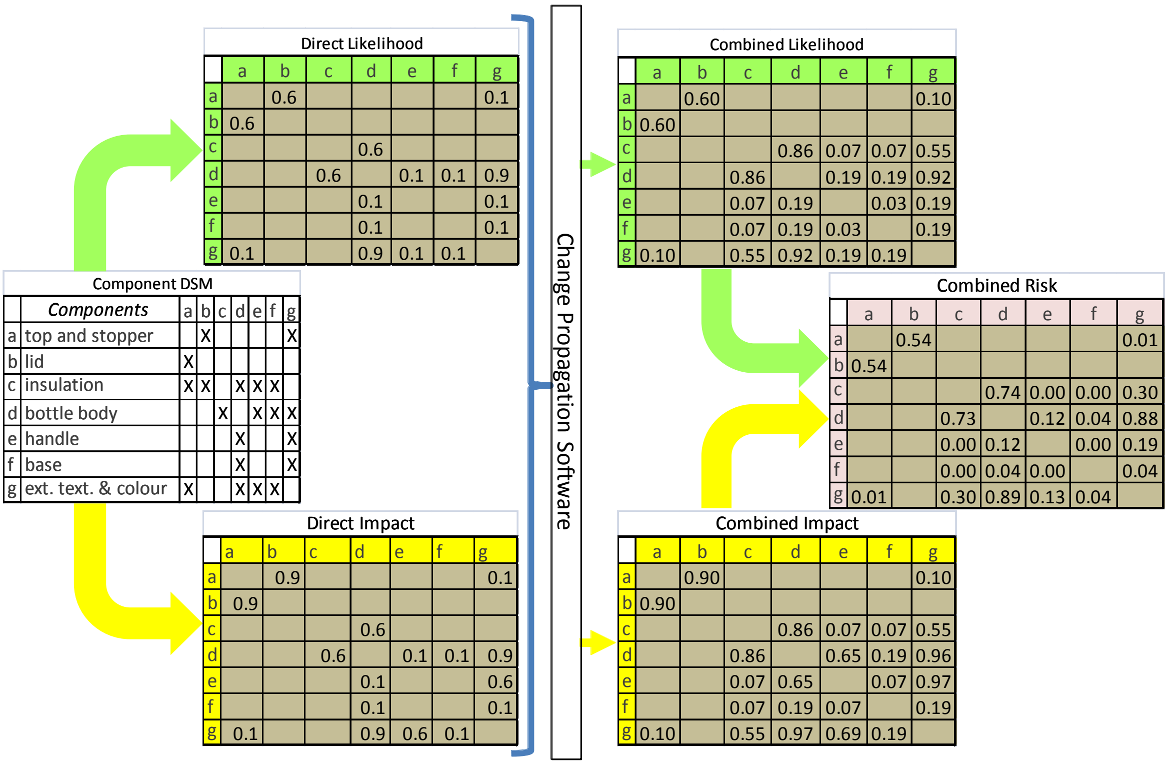
Figure 3. DMM versus DSM, adapted from Lindemann et al. [13], where DSM and DMM are highlighted in dark and light gray colours respectively. The lower case letters in parentheses represent the size of the corresponding matrices (rows x columns).

In the ChPM, C-DSM matrices are used to model change propagation risk such as dependencies between product components. Each cell of the matrix represents the change risk associated with a product component given that another component has changed as shown in Figure 2, where change propagation dependencies are defined subjectively by experts.

A DSM is very similar to a DMM. However, since columns and rows of the matrix represent the same domain, a DSM has to be symmetric along the diagonal elements and is always square, whereas a DMM is non-symmetric along the diagonal and is always a rectangular matrix [13]. Figure 3 presents a comparison of the uses of DMM and DSM in the product development process. The letters in parenthesis represent the sizes of the corresponding matrices.

Change Propagation Method (ChPM)

The specific steps followed in estimating the change propagation risk are shown in Figure 4 adapted from Clarkson et al. [12]. In the change propagation method, C-DSM matrices are defined to represent two major parts which are Direct Likelihood and Direct Impact matrices. The C-DSM shown on the left in Figure 4 is used to identify the dependencies between components in binary form. Then, the direct likelihood and impact C-DSMs are constructed based on these dependencies.



Direct Likelihood

	a	b	c	d	e	f	g
a		0.6					0.1
b	0.6						
c				0.6			
d			0.6		0.1	0.1	0.9
e				0.1			0.1
f				0.1			0.1
g	0.1			0.9	0.1	0.1	

Component DSM

	Components	a	b	c	d	e	f	g
a	top and stopper		X					X
b	lid	X						
c	insulation	X	X		X	X	X	
d	bottle body			X		X	X	X
e	handle				X			X
f	base				X			X
g	ext. text. & colour	X			X	X	X	

Direct Impact

	a	b	c	d	e	f	g
a		0.9					0.1
b	0.9						
c				0.6			
d			0.6		0.1	0.1	0.9
e				0.1			0.6
f				0.1			0.1
g	0.1			0.9	0.6	0.1	

Combined Likelihood

	a	b	c	d	e	f	g
a		0.60					0.10
b	0.60						
c				0.86	0.07	0.07	0.55
d			0.86		0.19	0.19	0.92
e			0.07	0.19		0.03	0.19
f			0.07	0.19	0.03		0.19
g	0.10		0.55	0.92	0.19	0.19	

Combined Risk

	a	b	c	d	e	f	g
a		0.54					0.01
b	0.54						
c				0.74	0.00	0.00	0.30
d			0.73		0.12	0.04	0.88
e			0.00	0.12		0.00	0.19
f			0.00	0.04	0.00		0.04
g	0.01		0.30	0.89	0.13	0.04	

Combined Impact

	a	b	c	d	e	f	g
a		0.90					0.10
b	0.90						
c				0.86	0.07	0.07	0.55
d			0.86		0.65	0.19	0.96
e			0.07	0.65		0.07	0.97
f			0.07	0.19	0.07		0.19
g	0.10		0.55	0.97	0.69	0.19	

Figure 4. C-DSM and change propagation method (ChPM) for the thermoflask [12]

The Direct Likelihood C-DSM provides the probabilities of change for a component in a column given that the component in the corresponding row has changed. Similarly, a Direct Impact C-DSM contains the information on the impact of change in a column element given that a row element has changed. These relationships indicate subjective probabilities that can be obtained through the usual estimation techniques.

These matrices can be fed into a change propagation software, shown after the blue bracket in Figure 4. The software traces each known change route to produce Combined Likelihood and Impact C-DSMs which are pairwise multiplied to produce a Combined Risk C-DSM. A Combined Risk C-DSM can be read as the risk of change propagation for components in a column given that the dependent component in the row changes. These combined risk values include all the direct and indirect risks associated with the change of another component. Therefore, they provide a single value of risk for each pair of components.

3 EVALUATION OF CHANGE PROPAGATION RISK

The method to evaluate risk of change propagation starts with a FAST diagram by following the "Why" and "How" logic from left to right which is explained in section 3.1. The functions at the lowest level of the FAST diagram are mapped onto components where the change propagation analysis is performed. In this section, the link between these building blocks is presented.

3.1 FAST Diagram

In the first level of the FAST diagram in Figure 1, the basic goal which is *transport liquid* is defined. The design characteristics are defined in the second level of the FAST diagram by asking the question "How". For instance, by asking the question "How to transport liquid?", the answer "transport liquid through stop contents." is obtained. The subsequent levels of the FAST diagram are constructed by using the same logic and asking the "How?" question (Figure 1). The FAST diagram in Figure 1 has three levels; however, for more complex designs more levels would be defined. At the lowest level of the FAST diagram, functions are matched with subsystems or components.

3.2 Change Propagation Risk

Each new design problem is represented by design characteristics (DC), such as weight, finish, dimensions, that are deployed through design solutions. Hence, it is assumed that a new DC represents changes to previous designs, and these changes can propagate to new and existing design functions and thus impact product cost and design effort. The raison d'être for developing the new technique is to estimate the risk of change propagation and the extent of design effort needed to handle all the changes. Change probabilities are derived from the C-DSM, explained in section 2.3.1. The different traceability matrices (including the DMM and the C-DSM matrices) are read to map the design functions onto the change propagation risks.

3.2.1 Connectivity Model

Domain Mapping Matrix: Design Functions to Subsystems/Components

As illustrated in Figure 1, design characteristics are mapped onto the elementary design functions. Taking this a step further, many-to-many mapping associates design functions with subsystems and/or product component types in Figure 5, since each design function may be associated with several subsystems and/or component types. For instance, to achieve a design characteristic, many functions and subsystems may be required, or many design characteristics may depend on a single component. Similarly, a change in design characteristic does not necessarily lead to a change in a subsystem or vice versa. A Domain Mapping Matrix (DMM) makes it possible to link design intents from different domains. This is why the use of DMM is particularly suitable in this case.

The functions of the thermoflask are many-to-many mapped onto its components through a domain mapping matrix (DMM) and is presented in Figure 5. First, the components which satisfy all the functions are determined by design experts. The links between the functions and the components are established by using dependency values highlighted in light brown. The dependency values are simply percentages which indicate what percentage of change in a component is reflected by its corresponding function. For instance, there is a very high dependency of 80% between the function *stop contents* and the component *top and stopper*. This means that if there is 50% change in the component *top and stopper,* the function *stop contents* has a probability of 40% that it will change. On the other hand, there is no dependency between *stop contents* and the *handle*; therefore, the corresponding cell is left blank indicating that even if the handle is completely changed, the function *stop contents* will not experience any change. For simplicity, it is assumed that the dependency values are determined by the design and manufacturing functions in a semi-quantitative manner.

In complex products, it may be more relevant to map design functions to subsystems rather than directly to the components which may not be feasible. The subsystems can also be mapped to components to achieve the desired granularity by using the same procedure. Since the thermoflask is a relatively simple product, the functions can be directly mapped to the components. However, in complex products the steps explained in this section have to be repeated twice, once for functions to subsystems and then again for subsystems to components.

In this analysis, the level of accuracy is crucial; so, a suitable choice of scale is required to represent the actual level of dependencies. For the thermoflask example, the dependencies are categorized into the following four groups:

- None: There is an insignificant or no physical relation between the component and the corresponding design function.
- Low: There is a very low relationship between the component and the corresponding design function. Range: $0.1 < de_{jkh} < 0.4$.
- Medium: There is significant relationship between the component and the corresponding design function. Range: $0.1 < de_{jkh} < 0.7$.
- High: There is very substantial relationship between the component and the corresponding design function. Range: $0.7 < de_{jkh} < 1$, where de_{jkh} is the dependency value between the jkth function and the hth component.

The dependency values are primarily used in tracing back the change propagation risk from the component level to the functions.

Component Design Structure Matrix (C-DSM): Component to Component

After the functions are linked to the components directly or indirectly through subsystems, the risk of change for each component is calculated. Component types that are sources of change and the impacted features are identified using the change propagation method as explained in section 2.3.1.

In this step, first the C-DSM which shows the dependencies between the components in binary form is constructed. Then, the likelihood of change (L_{hh1}) and the impact of change (I_{hh1}) for each component are assessed by design experts, and the corresponding matrices are constructed as shown in Figure 4. Once again, the choice of scale is important in estimating the likelihood and impact of change between components. In some cases, even though a risk can be estimated accurately, the time and computation power requirements may present challenges. Therefore, a discrete scale with a range of 0 to 1 is chosen and provided below to quantify the change propagation relation between components: None: 0, Low: L_{hh} or I_{hh} = 0.1, Medium: L_{hh} or I_{hh} = 0.6, High: L_{hh} or I_{hh} = 0.9, where:

L_{hh}: likelihood of change in a component (in the column) given that another component (in the row) is changed,

I_{hh}: impact of change on a component (in the column) if another component (in the row) changes.

Change Propagation Method

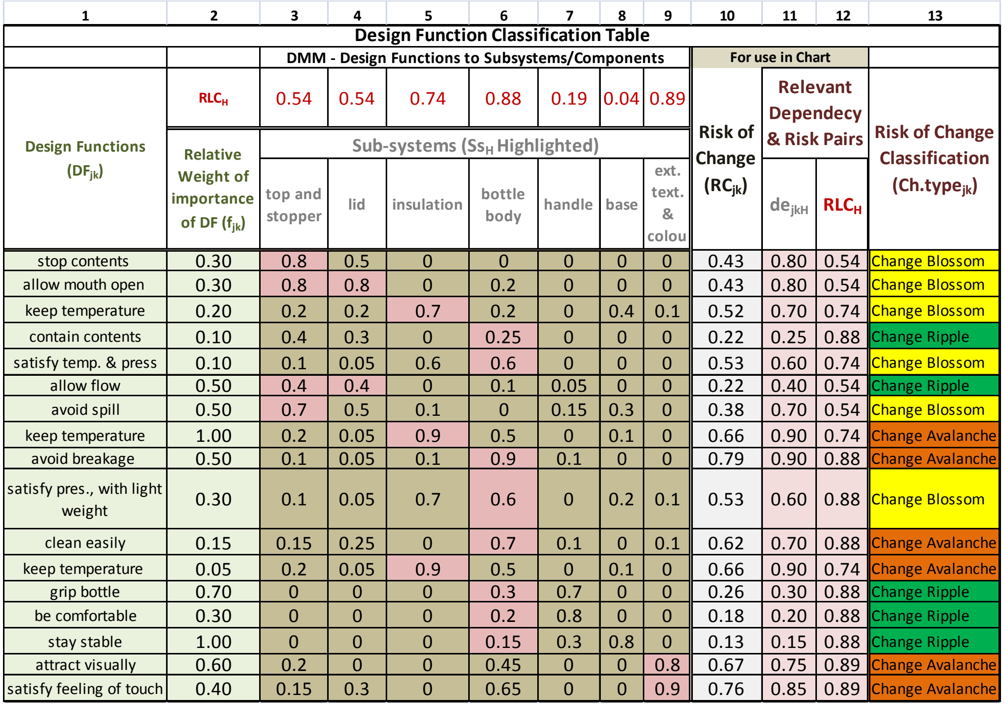
These matrices are fed into the change propagation software to obtain the combined likelihood and impact matrices which are pairwise multiplied to get the combined risk matrix as presented on the right side of Figure 4. A cell in the combined risk matrix represents the risk of change propagation for a component in the column given that the corresponding component in the row is changed. For instance, given that there is a change in the *bottle body* (component d at the left of combined risk matrix), there is 73% risk of change for the *insulation* through all direct and indirect change propagation links.

The matrices presented in Figure 4 are symmetric. This is a valid assumption for simple designs. However, for complex designs there may be one way dependencies between components which would result in a non-symmetric matrix, and may require significantly more computation and time.

Results of Change Propagation Method (ChPM) and Colour Coding

So far, the dependency level (de$_{jkh}$) between components and functions and the combined change propagation risk between the components (R$_{hh1}$) are obtained through the DMM and ChPM respectively. Now, the combined change propagation risk values obtained for each component should be traced back to the functional level.

The first step in tracing the component risk is to obtain the dominant R$_{hh1}$ (RLC$_H$). The calculation of RLC$_H$ is presented in Equation 1 which simply takes the maximum combined change propagation risk (R$_{hh1}$) for each row. These RLC$_H$ values are noted on the top row of the design function classification table as shown in Figure 5.



1	2	3	4	5	6	7	8	9	10	11	12	13
Design Function Classification Table												
		DMM - Design Functions to Subsystems/Components							For use in Chart			
Design Functions (DF$_{jk}$)	RLC$_H$	0.54	0.54	0.74	0.88	0.19	0.04	0.89	Risk of Change (RC$_{jk}$)	Relevant Dependecy & Risk Pairs		Risk of Change Classification (Ch.type$_{jk}$)
	Relative Weight of importance of DF (f$_{jk}$)	Sub-systems (Ss$_H$ Highlighted)										
		top and stopper	lid	insulation	bottle body	handle	base	ext. text. & colou		de$_{jkH}$	RLC$_H$	
stop contents	0.30	0.8	0.5	0	0	0	0	0	0.43	0.80	0.54	Change Blossom
allow mouth open	0.30	0.8	0.8	0	0.2	0	0	0	0.43	0.80	0.54	Change Blossom
keep temperature	0.20	0.2	0.2	0.7	0.2	0	0.4	0.1	0.52	0.70	0.74	Change Blossom
contain contents	0.10	0.4	0.3	0	0.25	0	0	0	0.22	0.25	0.88	Change Ripple
satisfy temp. & press	0.10	0.1	0.05	0.6	0.6	0	0	0	0.53	0.60	0.74	Change Blossom
allow flow	0.50	0.4	0.4	0	0.1	0.05	0	0	0.22	0.40	0.54	Change Ripple
avoid spill	0.50	0.7	0.5	0.1	0	0.15	0.3	0	0.38	0.70	0.54	Change Blossom
keep temperature	1.00	0.2	0.05	0.9	0.5	0	0.1	0	0.66	0.90	0.74	Change Avalanche
avoid breakage	0.50	0.1	0.05	0.1	0.9	0.1	0	0	0.79	0.90	0.88	Change Avalanche
satisfy pres., with light weight	0.30	0.1	0.05	0.7	0.6	0	0.2	0.1	0.53	0.60	0.88	Change Blossom
clean easily	0.15	0.15	0.25	0	0.7	0.1	0	0.1	0.62	0.70	0.88	Change Avalanche
keep temperature	0.05	0.2	0.05	0.9	0.5	0	0.1	0	0.66	0.90	0.74	Change Avalanche
grip bottle	0.70	0	0	0	0.3	0.7	0	0	0.26	0.30	0.88	Change Ripple
be comfortable	0.30	0	0	0	0.2	0.8	0	0	0.18	0.20	0.88	Change Ripple
stay stable	1.00	0	0	0	0.15	0.3	0.8	0	0.13	0.15	0.88	Change Ripple
attract visually	0.60	0.2	0	0	0.45	0	0	0.8	0.67	0.75	0.89	Change Avalanche
satisfy feeling of touch	0.40	0.15	0.3	0	0.65	0	0	0.9	0.76	0.85	0.89	Change Avalanche

Figure 5. Classification of design functions including DMM with risk of change

$$RLC_H = max_{1 \leq h1 \leq s1}(R_{hh1}) \ for\ each\ h = 1, 2, \dots, s \tag{1}$$

where, RLC$_H$: dominant risk of change propagation for hth component and R$_{hh1}$: risk of change propagation for component h1 due to component h.

For instance, the dominant risk for the component top and stopper is 54%, which means that in the worst case scenario there will be a 54% raise in the cost of another component. This value is obtained by taking the maximum of all the outgoing risks which are represented by the rows. This value is noted above the corresponding component in the design function classification table presented in Figure 5.

Then, the risk of change can be traced to the functional level by multiplying the dependency (de$_{jkh}$) with dominant risk level (RLC$_H$) values for each pair of functions and components. Finally, the risk of change values (RC$_{jk}$) are determined by taking the maximum of the resulting calculation for each function as presented in Equation 2.

$$RC_{jk} = max_{1 \leq h \leq s}(de_{jkh} \times RLC_H) \ for\ each\ pair\ of\ jk = 11, 12, \dots, np \tag{2}$$

where: RC$_{jk}$: change propagation risk for jkth design function and de$_{jkh}$: dependency value for jkth function.

For the thermoflask, the maximum risk value, 0.43 (RC$_{11}$) in case of the *top and stopper*, is noted in column 10 under the “For use in Chart” section in Figure 5. For clarity, the dependency value corresponding to the maximum risk value is also highlighted in light pink in the design function classification table. The corresponding dependency and dominant risk values, 0.8 (de$_{11B}$) and 0.54 (RLC$_B$) in the case of *stop contents*, are noted in columns 11 and 12 under the “For use in Chart” section. These steps are repeated for each design function of the thermoflask.

Finally, the values in the “Relevant Dependency & Risk” section are plotted on a colour scaled chart in order to determine the classification of the design functions. As mentioned before, the scale of colour coding is highly product dependent and should be decided by the designers prior to the construction of the chart.

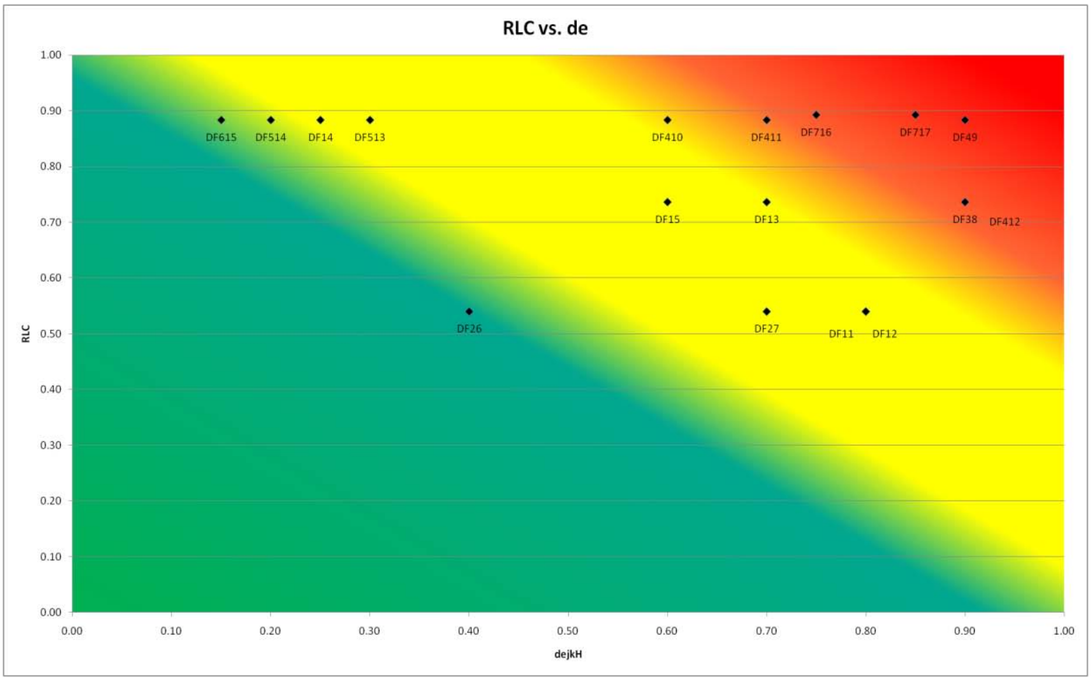


Figure 6. Classification chart (dependency vs. risk of change propagation)

For simplicity, a generic three colour dominant risk vs. dependency chart is presented in Figure 6 where the colour coding is as follows:

- Low Risk: Change Ripples (Green)
- Medium Risk: Change Blossoms (Yellow)
- High Risk: Change Avalanches (Red).

The type of risk of change classification (Ch-type$_{jk}$) for design functions can easily be determined by identifying the colour for each point on the chart.

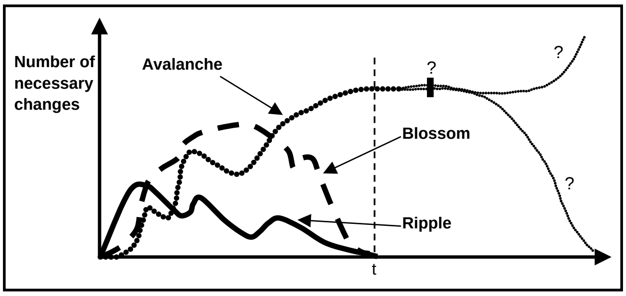
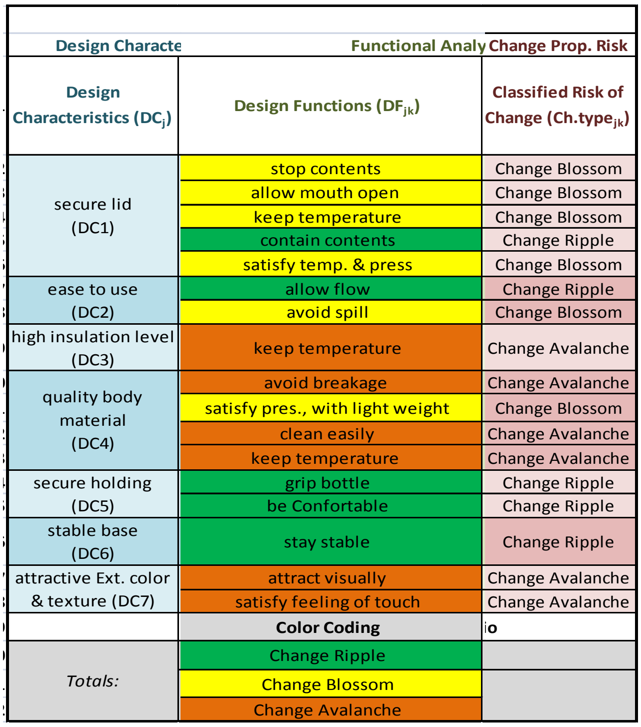


Figure 7. Instantaneous effort requirement in time [20, 21]

For instance in Figure 6, the function *stop contents* is classified as “change blossom” because there is relatively high dependency between it and the component *top and stopper*, and the risk of causing change propagation for the *top and stopper* is medium. This means that a change in the function *stop contents* is likely to cause increased effort in the short term due to change propagating to other components. But, the extra effort will diminish shortly after, when the propagating changes hit change absorbers (components which do not propagate change). In other words, all changes are put under control quickly and completed within the limits of given time and cost. On the other hand, the function *avoid breakage* is classified as a “change avalanche”, which means that the increased effort requirement after the change will be out of control (continually increasing) and likely to cause project and cost overruns. The propagating change passes through many other change multipliers, spreading to more and more components in time. A change avalanche can be catastrophic for a project, causing cost and schedule overruns or quality issues, which result in loss of profit, market-share, and customer satisfaction. The graphical illustration of the change propagation type is presented in Figure 7.

The results of the risk versus dependency chart are then fed into column 13 of the DMM. Then, the design functions, which are in column 2 of Figure 8, are colour coded based on classified risk of change values. For the thermoflask example, a tricolour scale green-yellow-orange is selected and presented in Figure 8.



Design Characte	Functional Analy	Change Prop. Risk
Design Characteristics (DC_j)	Design Functions (DF_{jk})	Classified Risk of Change ($Ch.type_{jk}$)
secure lid (DC1)	stop contents	Change Blossom
	allow mouth open	Change Blossom
	keep temperature	Change Blossom
	contain contents	Change Ripple
	satisfy temp. & press	Change Blossom
ease to use (DC2)	allow flow	Change Ripple
	avoid spill	Change Blossom
high insulation level (DC3)	keep temperature	Change Avalanche
quality body material (DC4)	avoid breakage	Change Avalanche
	satisfy pres., with light weight	Change Blossom
	clean easily	Change Avalanche
	keep temperature	Change Avalanche
secure holding (DC5)	grip bottle	Change Ripple
	be Confortable	Change Ripple
stable base (DC6)	stay stable	Change Ripple
attractive Ext. color & texture (DC7)	attract visually	Change Avalanche
	satisfy feeling of touch	Change Avalanche
	Color Coding	io
Totals:	Change Ripple	
	Change Blossom	
	Change Avalanche	

Figure 8. Highlighted design functions based on their classified risk of change

4 CONCLUSION AND FUTURE WORK

In this paper, the change propagation risk of design functions was evaluated using a FAST diagram, domain mapping matrix (DMM), and component design structure matrices (C-DSM) which were fed into a change propagation software. The design functions were many-to-many mapped onto components through the DMM. The risks were evaluated at the component level in the C-DSMs and traced back to the functional level through the DMM. Finally, the design functions were classified based on their effect on the instantaneous effort requirement.

The classification of design functions in this manner has very important implications for designers because many design performance metrics such as design effort, span time, product cost, and other risk attributes are directly or indirectly affected by propagating changes. A change in a function which is classified as an avalanche can be catastrophic for a design project causing cost and schedule overruns. The evaluation method presented in this paper tackles this issue by providing easy to follow information to managers early in the design process.

Highlighting the design functions based on their change propagation type provides insight into the effects of an engineering change, which may be required in the future. Since the propagating changes have direct and indirect impacts on many project performance metrics, being aware of the change propagation risk structure of a product provides the following mitigation possibilities:

- directing efforts towards modularization of subsystems to reduce change propagation risk,
- choosing a lower risk design solution among possible design solutions.

One limitation of this method is the quantification of the dependencies within the components and between the functions and the components. This issue can be improved by standardizing the assessment procedures and setting defined work instructions. Another limitation is the fact that ChPM does not consider new patterns of change. Although it is an advanced technique for assessing change propagation, ChPM cannot systematically identify and analyze unanticipated changes.

Future work will concentrate on improving and developing standardized risk quantification methods and defining standard work procedures to reduce the variability in estimating dependency, likelihood and impact values. Also, developing a monitoring system to predict new patterns of change is seen as highly beneficial.

ACKNOWLEDGEMENTS

The research reported in this paper was undertaken as part of the "Collaborative Development for Product Life Cycle Management" (PLM2 CRIAQ) project. The authors are grateful for the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC), the Consortium for Research and Innovation in Aerospace in Quebec (CRIAQ), Bombardier Inc., CAE Inc., CMC Electronics Inc., Pratt & Whitney Canada Corp., and Rolls-Royce Canada Limited.

Also, the authors would like to thank Dr. Onur Hisarciklilar and Dr. Khadidja Grebici for their valuable feedback and contributions to the research.

REFERENCES

[1] Duhovnik, J. and Tavcar, J., Reengineering with rapid prototyping. In *TMCE 2002: Proceedings of the Fourth International Symposium on Tools and Methods of Competitive Engineering*. Wuhan, PR China, 22-26 April, 2002. pp117-130

[2] Crossland, R., Williams, J. and McMahon, C., An object-oriented modeling framework for representing uncertainty in early variant design. *Research in Engineering Design*, 2003, 14(3), pp173-183.

[3] Keller, R., Eckert, C.M. and Clarkson, P.J., Using an engineering change methodology to support conceptual design. *Journal of Engineering Design*, 2008.

[4] Park, R., *Value Engineering: A Plan for Invention*. (CRC, 1998).

[5] Young, P. and Tippins, S., *Managing business risk: an organization-wide approach to risk management*. (American Management Association, 2000).

[6] Wang, J. and Roush, M., Risk Engineering and Management. (Marcel Dekker, Inc, 2000).

[7] Wright, I., A review of research into engineering change management: implications for product design. *Design Studies*, 1997, 18(1), pp33-42.

[8] Reidelbach, M.A., Engineering change management for long lead time production environments. *Production and Inventory Management Journal*, 1991, 32(2), pp84-88.

[9] Schach, S. and Tomer, A., A maintenance-oriented approach to software construction. *Journal of Software Maintenance*, 2000, 12(1), pp25-45.

[10] Rajlich, V., Modeling software evolution by evolving interoperation graphs. *Annals of Software Engineering*, 2000, 9(1), pp235-248.

[11] Ollinger, G. and Stahovich, T., Redesignit-a constraint-based tool for managing design changes. *Proc. ASME International Design Engineering Technical Conferences, Design Theory and Methodology, Pittsburgh, USA*, 2001.

[12] Clarkson, P.J., Simons, C. and Eckert, C.M., Predicting change propagation in complex design. In *Proceedings of DETC'01*. Pittsburgh, Pennsylvania. (ASME)

[13] Danilovic, M. and Browning, T.R., Managing complex product development projects with design structure matrices and domain mapping matrices. *International Journal of Project Management*, 2007, 25(3), pp300-314.

[14] Lindemann, U., Maurer, M. and Braun, T., *Structural Complexity Management: An Approach for the Field of Product Design*. (Springer Verlag, 2008).

[15] Steward, D.V., The design structure system: a method for managing the design of complex systems. *IEEE Transactions on Engineering Management*, 1981(28), pp71-74.

[16] Browning, T.R., Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Transactions on Engineering Management*, 2001, 48(3), pp292-306.

[17] Wyatt, D.F., Wynn, D.C. and Clarkson, P.J., A computational method to support product architecture design. 2009.

[18] Sosa, M., Agrawal, A., Eppinger, S. and Rowles, C., A Network Approach to Define Modularity of Product Components. *ASME DTM*, 2005, 85422.

[19] Sosa, M., Eppinger, S. and Rowles, C., Identifying modular and integrative systems and their impact on design team interactions. *Journal of Mechanical Design*, 2003, 125, pp240.

[20] Eckert, C.M., Clarkson, P.J. and Zanker, W., Change and customisation in complex engineering domains. *Research in Engineering Design*, 2004, 15(1), pp1-21.

[21] Jarratt, T., Clarkson, P.J. and Eckert, C.M., Engineering change. *Design Process Improvement: A review of current practice* pp262-285 (Springer, London, UK, 2005).

Contact: Arman Oduncuoglu, Department of Mechanical Engineering, McGill University, Montreal, H3A 2K7, Canada. Tel: 1 (514) 574-3422. Email: arman.oduncuoglu@mail.mcgill.ca

Arman Oduncuoglu is a graduate of McGill University. He completed his Master degree in February 2011 and has started working at Bombardier Aerospace in product lifecycle management.

Vince Thomson is the Werner Graupe Professor of Manufacturing Automation in the Department of Mechanical Engineering, McGill University. He has been involved in manufacturing and information technology related research for the past 30 years. His process management research is focused on new product introduction, concurrent engineering and process monitoring.