INTERNATIONAL DESIGN CONFERENCE - DESIGN 2008
Dubrovnik - Croatia, May 19 - 22, 2008.

INFLUENCE OF FEATURE CHANGE PROPAGATION ON PRODUCT ATTRIBUTES IN CONCEPT SELECTION

E. C. Y. Koh, R. Keller, C. M. Eckert and P. J. Clarkson

Keywords: features, change propagation, concept selection

1. Introduction

Companies have to design new products to compete for market share. Many are *variant designs*, which involve modifying parameters like component geometry and material to change product attributes without changing its configuration [Otto and Wood 2001]. Product attributes can be described as product behaviours that are expected or derived from the product structure [Gero 1990]. In other words, product attributes can be changed when the structure and components of a product are modified. These changes in product attributes can be intentional or unintentional. In this paper, a distinctive characteristic or arrangement of components/modules *intended* to satisfy a *required* product attribute is defined as a design feature. For example, a design feature of a 'blade' component might be 'sharp blade edge', directed at fulfilling the *required* product attribute of '*high* cutting speed'. Usually more than one design feature can be implemented to achieve a required product attribute. An extensive classification of features can be found in Shah and Mantyla [1995].

The selection of design features is usually made during the conceptual design phase by use of concept selection tools. During this phase, concepts with different design features will be evaluated with respect to required attributes. The House of Quality (HoQ) is one possible method that can be used for such evaluation [Hauser and Clausing 1988]. However, there is currently no concept selection method available that consider the influence of feature change propagation on product attributes due to component change propagation (see [Eckert et al. 2004] on change propagation). The consideration of feature change propagation on product attributes is important as unintended changes can affect product performance. Failure to meet the required performance level can in turn result in lengthy and unplanned rework. A method is thus required to help designers consider feature change propagation early in the design process.

Existing change analysis tools include RedesignIT [Ollinger and Stahovich 2004], the Change Prediction Method [Clarkson et al. 2004], and the Property-Driven Development approach [Weber et al. 2003]. RedesignIT looks at the causal relationships in change propagation in terms of changes to product attributes. The Change Prediction Method assesses change risks in terms of changes to product components. These tools, however, do not consider the dependencies between product attributes and components. The Property-Driven Development approach provides a characteristics-properties modelling framework to analyse how changes to product features (characteristics) can influence product attributes (properties) and vice versa. Nevertheless, this approach analyses change propagation directly on a feature-to-feature level and thus can be computational intensive. For instance, in Conrad et al. [2007], a change case for a shaft to collar connection (part of a pulley design) requires an analysis of 11 feature descriptions.

In this paper, an approach which considers feature change propagation will be introduced to predict and analyse the performance of attributes during concept selection. This approach assumes that the introduction of new design features is the only source of unintended changes. Although the level of innovation can vary between different design projects, only *variant design* will be considered in this paper. Section 2 of this paper discusses the interactions between product attributes and components when new design features are implemented. Section 3 provides a brief review of concept selection and change propagation and justifies why current concept selection tools should be improved with change prediction capabilities. A step-by-step approach for predicting performance of attributes is presented in Section 4. Section 5 discusses the feasibility of this new approach on a bicycle case example. Lastly, Section 6 concludes the paper and outlines future research directions.

2. Implementing new design features

It is important to understand the relationship between product attributes, features, and components in order to consider change propagation during conceptual design. A product attribute of a bicycle can be its 'weight' while a *required* product attribute will indicate the goal of the design. For example, a '*low* weight' bicycle. In order to fulfil the '*low* weight' attribute, changes to the existing product must be made. By definition, such a change can be carried out by implementing a design feature which is described by how one or more components can be changed to meet the required product attribute. In an ideal situation, the required attribute will be achieved with no changes to other product attributes and components. However, implementing a design feature can sometimes affect other product attributes. Whether this feature change propagation results in an improvement or deterioration of its attributes depends on the intent of the design.

Figure 1 shows an example of how improving a required product attribute can affect other product attributes for a simple bicycle design. If a required attribute of 'high maximum speed' is needed for a new bicycle, designers can try to achieve this requirement by implementing a design feature of 'large wheels'. Obviously, this design feature will involve changing the 'wheels' component. However, changes to the 'wheels' component can propagate and affect other components connected to it. One possible example is the 'frame' component. Changing the 'frame' component to accommodate the 'large wheels' design feature can in turn result in a feature of 'large frame size'. Since a larger frame size will result in a heavier bicycle, the 'weight' attribute can subsequently be affected. If a required attribute of 'low weight' is necessary for the new bicycle, a heavier bicycle will affect the performance of that attribute in a negative way. It should be noted that the initiating required attribute can also be affected through change propagation. For example, a heavier bicycle can also affect the maximum speed achievable.

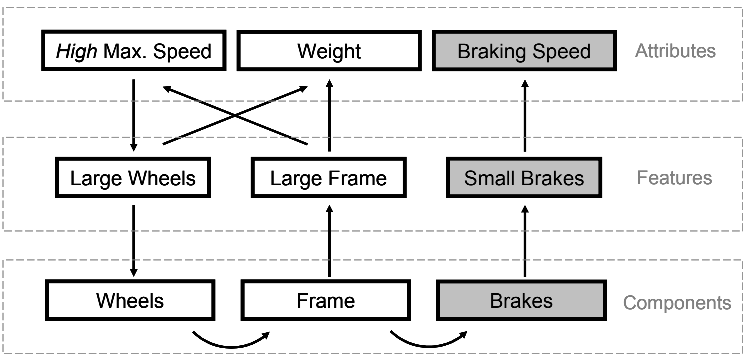


Figure 1. Relationship between Attributes, Features, and Components

The complexity of the problem increases if indirect change propagation between components is considered (grey column on the right of Figure 1). For example, even though the 'wheels' component is not directly connected to the 'brakes' component, changing the 'wheels' can affect the 'frame' which can in turn affect the ‘brakes’. Using the same reasoning mentioned above, the ‘brakes’ have to be downsized in order to accommodate for the design feature of ‘large wheels’. This resulting change, in essence, can be described as a ‘small brakes’ feature and can subsequently affect the ‘braking speed’ attribute in a negative way.

It should be noted that not all propagation paths have the same effect on the product attributes. Some features can affect an attribute more than others and some components are more likely to propagate changes than others. In addition, in most cases, more than one design feature can be implemented when a particular attribute is required. For instance, designers have a choice between implementing a ‘large wheels’ design feature and a ‘light frame material’ design feature to achieve the required product attribute of ‘high maximum speed’. A comprehensive analysis of these design interactions is therefore crucial during the conceptual design stage to minimise unexpected attribute performance and help designers identify the best solution.

3. Concept selection and change propagation

Design concepts can be described as solution alternatives differentiated by their design features. Currently, there are a number of established methods available to rank and evaluate design concepts. The Pugh Charts use a minimal evaluation scale of worse (-), same (s), and better (+) to rank concepts with respect to a reference concept [Pugh 1990]. A two-stage concept selection methodology was developed by Ulrich and Eppinger [1995] to structure the selection process into the concept screening stage and the concept scoring stage. Pahl and Beitz [1996] describe the use of morphological analysis to break down modules systematically and evaluate the performance of each module independently. The House of Quality (HoQ), which is commonly used to translate customer requirements into product attributes, can also be used for concept selection to analyse the tradeoffs between design solutions [Hauser and Clausing 1988]. The ‘roof’ of the HoQ, nevertheless, assumes that the influence of different design features is symmetrical and do not analyse feature interactions at the component level. Since product attributes, features, and components are all connected as shown in Figure 1, this implies that some of the connections can be overlooked: especially the indirect ones which are difficult to detect. The methods mentioned above make use of concept scoring to analyse the tradeoffs between attributes and identify concepts that can produce good overall product performance. However, none of these methods explicitly take into account the effects of change propagation between components which can occur during product redesign [Eckert et al. 2004]. Since a feature change can only be carried out by changing or rearranging product components, the argument that change propagation can affect the overall performance of a product is a plausible one. The ability to predict and minimise unexpected changes during product redesign is therefore important. This can be achieved by adding change prediction capabilities to concept selection tools. Such assessment can alert designers about possible change propagation and help them make better design choices.

The Change Prediction Method (CPM) developed by Clarkson et al. [2004] can be used to predict component change risk during product redesign. This method captures component connectivity using a Dependency Structure Matrix (DSM) to compute the change likelihood and impact between product components (see [Browning 2001] on DSM). Although the CPM can detect indirect component change propagation shown in Figure 1 (grey column), it does not have a concept selection interface which can predict and analyse the overall product performance. A more systematic approach with change prediction capability is therefore required to analyse the performance of different concepts during concept selection. The remainder of this paper introduces an approach that can be used to predict product and attribute performance quantitatively when implementing a new design feature.

4. Analysis of attribute performance due to feature change

The approach for analysing attribute performance is shown in Figure 2. This approach modifies and extends the use of the HoQ by considering change propagation using the CPM method. The key idea is to capture the influence of unintended component changes on product attributes, and present this information as an aggregate performance rating for each product attribute.

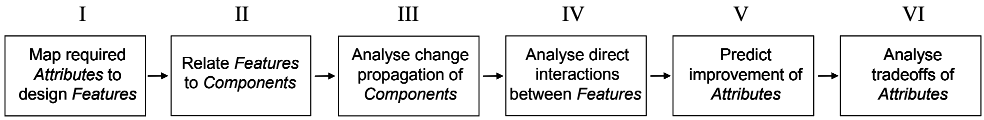


Figure 2. Approach to analyse attributes performance

The approach begins by listing required product attributes against related design features and indicating the influence of each feature with respect to each attribute (Step I). Next, the product components that are related to each of the design features are identified through a feature-component mapping process (Step II). This is followed by an analysis of the likelihood of component change propagation (Step III). Subsequently, the type and strength of interactions between design features are analysed (Step IV). Once the influence and change propagation of all design features have been identified, the predicted performance improvement of each product attribute can be computed by using a performance improvement algorithm (Step V). The final step is to tabulate these performance improvement values into the Attribute Improvement Matrix and analyse the performance tradeoffs between different product attributes (Step VI). The following subsections will provide more explanation on each step of the approach.

4.1 Step I: List and determine feature influence

This step assumes that the designers had successfully translated all the key requirements and constraints into required product attributes and is similar to the HoQ [Hauser and Clausing 1988]. Once the required product attributes have been successfully identified, relevant design features that can affect these product attributes can be listed as shown in Figure 3. The *primary attributes* (PA) section lists all the product attributes that are required to meet the requirements in the new design project. These are the product attributes that the designers want to change intentionally. On the other hand, the *secondary attributes* (SA) are product attributes that are fundamental to the success of the product but are not the main focus of the new project at hand. These attributes can be seen as design constraints that must be observed. The inclusion of these attributes is to ensure that the overall performance of the new product is taken into consideration. The columns are divided into two sections. The *primary features* (PF) section lists all the design features identified by the designers that can improve the primary attributes while the *secondary features* (SF) section lists all the design features that have a strong influence on the secondary attributes. Once the product attributes and design features have been established, the performance impact (*PI*) of each feature on different attributes can be determined through concept scoring and filled into the matrix.

Attribute-Feature Matrix		*Primary Features*			*Secondary Features*			
		Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7
Primary Attributes	Attribute 1	$PI_{1,1}$	$PI_{1,2}$					
	Attribute 2			$PI_{2,3}$				
Secondary Attributes	Attribute 3		$PI_{3,2}$		$PI_{3,4}$	$PI_{3,5}$		
	Attribute 4						$PI_{4,6}$	
	Attribute 5	$PI_{5,1}$					$PI_{5,6}$	$PI_{5,7}$

Figure 3. Attribute-Feature Matrix

It is worth noting that the list of primary product attributes can be different for each design project. However, the secondary attributes can be kept the same for different projects even though they might not be the focus of the new product. This is to ensure that unintentional feature change propagation to these basic attributes is always captured. In addition, a secondary attribute in one project can also be a primary attribute in another depending on the focus of the project.

4.2 Step II: Relate features to components

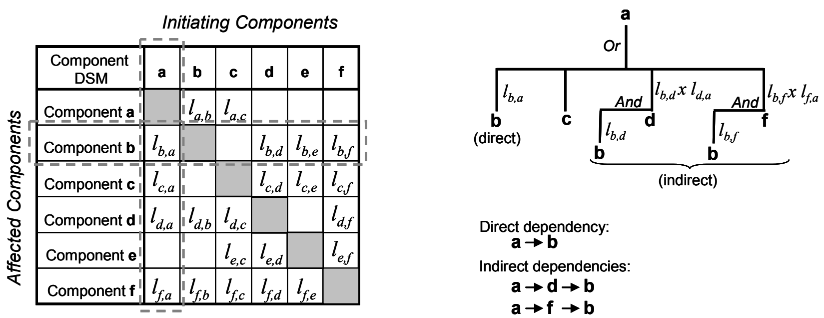
The components relevant to the design features can be identified and marked in a Feature-Component Matrix as shown in Figure 4. This step is important as new design features are always implemented through changes to components or modules. The component breakdown can come from previous product models and the granularity of the component breakdown should be on the same level of abstraction as the design features identified in Step I. The number of components associated with a given design feature depends on the nature of the feature. For example, a design feature of 'high gear ratio' will usually involve changing of more than one gear.

Feature-Component Matrix	Component 1	Component 2	Component 3	Component 4	Component 5	Component 6	Component 7	Component 8
Feature 1	X	X						
Feature 2		X						
Feature 3	X		X	X	X			
Feature 4					X	X	X	
Feature 5								X

Figure 4. Feature-Component Matrix

4.3 Step III: Analyse component change propagation

Once the relevant components have been identified, the next step is to analyse the likelihood of feature change propagation due to a component change. This step uses the Change Prediction Method (CPM) described by Clarkson et al. [2004] to predict the change likelihood between components or modules. Figure 5a shows an example of a direct change likelihood DSM of a product. The column headings represent the change instigating components while the row headings show the affected components. Each cell in the matrix contains a numerical representation of the direct change likelihood (l) between two components. The direct change likelihood can be elicited from the designers to provide an indication of how likely changes are to propagate from one component to another.



(a) (b)

Figure 5. (a) Component DSM (b) Change propagation tree [Clarkson et al. 2004]

All components in the product must be considered in order to analyse indirect change propagation. Figure 5b shows an example of a change propagation tree derived from the direct change likelihood

DSM. By using the CPM method, the combined likelihood (L) of component change propagation which considers direct and indirect change propagation can be calculated (see [Clarkson et al. 2004]).

4.4 Step IV: Determine feature interactions

This step is similar to the “roof” of the House of Quality [Hauser and Clausing 1988] but the analysis of feature interactions is more exhaustive in this method as all product components had been analysed systematically in Step II and III. The strength and type of design interactions (DI) between each design feature will only be considered if they share a product component either directly (Step II) or through component change propagation (Step III). The interaction is positive when the implementation of either design feature facilitates the implementation of the other. Conversely, the interaction will be negative if two design features are related and the implementation of either design feature will conflict with the implementation of the other. When two design features are not related, the interaction will be zero. See Figure 6 for an illustration of a Feature Interaction Matrix. It should be noted that unlike the HoQ, the interaction between design features can be asymmetrical. For example in Figure 6, Feature 5 can affect Feature 2 and 3 but Feature 2 and 3 do not affect Feature 5. This can happen when Feature 5 is not an existing feature. Hence, Feature 5 will only have an effect on other design features if it is implemented.

Feature Interaction Matrix		*Initiating*				
		Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
Affected	Feature 1		$DI_{1,2}$	$DI_{1,3}$	$DI_{1,4}$	
	Feature 2	$DI_{2,1}$			$DI_{2,4}$	$DI_{2,5}$
	Feature 3	$DI_{3,1}$				$DI_{3,5}$
	Feature 4	$DI_{4,1}$	$DI_{4,2}$			
	Feature 5					

Figure 6. Feature Interaction Matrix

4.5 Step V: Predict feature improvement

An improvement algorithm to predict feature improvement will be described in this section. The main assumption of this algorithm is that an emerging change is always implemented to *accommodate* an initiated change. Based on this assumption, the predicted performance improvement of an attribute $\boldsymbol{k}$ by a design feature $\boldsymbol{i}$ ($impr_{k,i}$) can be expressed as shown in Equation 1:

$$impr_{k,i} = PI_{k,i} + \sum_{j=1}^{n}[L_{j,i} \times DI_{j,i} \times PI_{k,j}] \quad (1)$$

where $\boldsymbol{PI_{k,i}}$ represents the performance impact of feature $\boldsymbol{i}$ on attribute $\boldsymbol{k}$ and $\boldsymbol{PI_{k,j}}$ represents the performance impact of feature $\boldsymbol{j}$ on attribute $\boldsymbol{k}$ (Step I); $\boldsymbol{L_{j,i}}$ represents the combined likelihood of component change propagation from feature $\boldsymbol{i}$ to feature $\boldsymbol{j}$ (Step III); $\boldsymbol{DI_{j,i}}$ represents the design interaction between feature $\boldsymbol{i}$ and feature $\boldsymbol{j}$ (Step IV); and $\boldsymbol{n}$ represents the total number of features analysed. The first part of the equation indicates the direct design feature performance predicted by the designers while the second part indicates the indirect design feature performance due to feature change propagation. The indirect design feature performance can either build on or cancel out the direct design feature performance depending on the propagation likelihood, polarity, and strength of the design interactions.

4.6 Step VI: Tabulate Attribute Improvement Matrix

The calculation of predicted improvement for all design features can be carried out using Equation 1 to provide the designers with a better understanding of how each design feature influences a product attribute. This step can be repeated for different attributes by using the appropriate combined likelihood (*L*), design interaction (*DI*), and performance impact (*PI*) values captured in Step I to Step IV. The results can be tabulated in the Attribute Improvement Matrix as shown in Figure 7.

Attribute Improvement Matrix		*Primary Features*		
		Feature 1	Feature 2	Feature 3
Primary Attributes	Attribute 1	$Impr_{1,1}$	$Impr_{1,2}$	$Impr_{1,3}$
	Attribute 2	$Impr_{2,1}$	$Impr_{2,2}$	$Impr_{2,3}$
Secondary Attributes	Attribute 3	$Impr_{3,1}$	$Impr_{3,2}$	$Impr_{3,3}$
	Attribute 4	$Impr_{4,1}$	$Impr_{4,2}$	$Impr_{4,3}$
	Attribute 5	$Impr_{5,1}$	$Impr_{5,2}$	$Impr_{5,3}$
Overall Improvement		*Impr(1)*	*Impr(2)*	*Impr(3)*

Figure 7. Attribute Improvement Matrix

By examining the Attribute Improvement Matrix, designers can analyse the performance tradeoffs between design features within a product attribute and between different product attributes. For example, the designers might have an interest on a particular product attribute and can focus their attention on analysing which is the best design feature that can fulfil the performance level required for that attribute. This can be done by comparing the predicted improvement values of different design features across the columns. For simplicity, the secondary features are not listed in the Attribute Improvement Matrix as these design features do not improve the primary attributes directly. The designers might also want to analyse the effect over different attributes if a particular design feature is changed. This can be carried out by inspecting the predicted improvement values of that design feature against different attributes. By summing up the improvement values for each column, the overall improvement values at the bottom of the matrix can provide a rough indication on the overall product performance when a design feature is implemented.

5. Case example

This section uses a hypothetical bicycle example to demonstrate the feasibility of the approach described in Section 4. Figure 8 shows a simplified component breakdown of a bicycle. Assuming that the product attributes had been finalised, the designers can proceed to fill the Attribute-Feature Matrix as described in the previous section (Step I, see Figure 9a). The rating scale used for the Attribute-Feature Matrix is from -3 to 3. The primary attributes, in this case, are 'high maximum speed' and 'low weight'. The secondary attributes are 'short braking time', 'low product cost', 'high impact strength', and 'low frame deflection'. The possible primary features that can meet the requirements of the primary attributes are identified as 'large wheel size', 'more gears to achieve high speed ratio', 'small frame size and tube thickness', and 'titanium frame'. It should be noted that the geometry and the material of the frame are listed as primary features even though these design features have strong influence on the secondary attributes of 'high impact strength' and 'low frame deflection'. A 'strong and big brakes' feature is also included in this example as a secondary feature for the secondary attribute of 'short braking time'. Assuming that none of the components listed in Figure 8 can significantly increase the cost of the bicycle, no secondary feature is required for the secondary attribute of low cost.

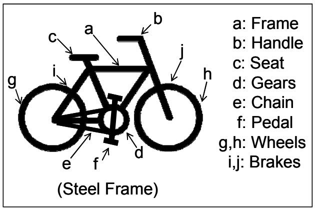


Figure 8. Component breakdown of a bicycle

Attribute-Feature Matrix		PF				SF
		Large Wheels	More Gears	Small Frame	Ti Frame	Strong Brakes
PA	High Max Speed	1	3	2	2	0
PA	Low Weight	-1	0	2	2	0
SA	Fast Braking	0	0	1	1	3
SA	Low Cost	-1	-2	2	-3	-1
SA	High Strength	0	0	-1	0	0
SA	Low Deflection	-1	0	1	-2	0

(a)

Feature Interaction Matrix	Large Wheels	More Gears	Small Frame	Ti Frame	Strong Brakes
Large Wheels		-0.3	-0.9	-0.6	-0.3
More Gears	-0.3		-0.3	-0.3	-0.3
Small Frame	-0.9	-0.3		-0.3	-0.3
Ti Frame	0	0	0		-0.3
Strong Brakes	-0.3	-0.3	-0.3	-0.3	

(b)

Figure 9. (a) Attribute-Feature Matrix of a bicycle (b) Feature Interaction Matrix of a bicycle

The next step (Step II) is to map design features to relevant components. For ease of illustration, only one component is assigned to each design feature as shown in Figure 10a. The following step (Step III) is to analyse the change propagation between components related to each design feature. By filling the direct change likelihood DSM as shown in Figure 10b, the combined change likelihood can be calculated using the CPM method described in the previous section. The rating levels used for the direct change likelihood are 0.3 for ‘low’, 0.6 for ‘medium’, and 0.9 for ‘high’ possibility of change. Figure 10c shows the combined likelihood DSM of relevant components.

Feature-Component Matrix	Wheels	Gears	Frame	Brakes
Large Wheels	X			
More Gears		X		
Small Frame			X	
Ti Frame			X	
Strong Brakes				X

(a)

Direct Change Likelihood	Wheels	Gears	Frame	Brakes	Handle	Seat	Chain	Pedals
Wheels		0.3	0.3	0.3	0	0	0	0
Gears	0.3		0.3	0	0	0	0.9	0.6
Frame	0.9	0.3		0.3	0.3	0.3	0	0.3
Brakes	0.3	0	0.3		0	0	0	0
Handle	0	0	0.3	0		0	0	0
Seat	0	0	0.3	0	0		0	0
Chain	0	0.6	0	0	0	0		0
Pedals	0	0.9	0.3	0	0	0	0	

(b)

Combined Change Likelihood	Wheels	Gears	Frame	Brakes
Wheels		0.44	0.45	0.38
Gears	0.58		0.49	0.29
Frame	0.92	0.63		0.49
Brakes	0.51	0.31	0.39	

(c)

Figure 10. (a) Feature-Component Matrix (b) Direct change likelihood (c) Combined change likelihood of related components

The direct interactions between design features can be captured by completing the Feature Interaction Matrix as shown in Figure 9b (Step IV). Note that the design feature ‘titanium frame’ affects other design features but is not influenced by others as it is not an existing feature. The rating levels used are ±0.3 for ‘low’, ±0.6 for ‘medium’, and ±0.9 for ‘high’ design interactions. Using the improvement algorithm (Step V), the predicted performance improvement of each design feature with respect to each required attribute can be tabulated as shown in Figure 11a (Step VI). The performance improvement of each design feature without considering change propagation is shown in Figure 11b for comparison.

Attribute Improvement Matrix		PF			
		Large Wheels	More Gears	Small Frame	Ti Frame
PA	High Max Speed	-1.2	2.5	1.1	0.7
	Low Weight	-2.7	-0.3	2.4	1.7
SA	Fast Braking	-1.3	-0.5	0.6	0.3
	Low Cost	-2.2	-2.2	2.8	-2.9
	High Strength	0.8	0.2	-1.0	0.3
	Low Deflection	-1.8	-0.1	1.4	-2.0
Total improvement		-8.3	-0.3	7.4	-2.0

(a)

Attribute Improvement Matrix *(without considering change propagation)*		PF			
		Large Wheels	More Gears	Small Frame	Ti Frame
PA	High Max Speed	1	3	2	2
	Low Weight	-1	0	2	2
SA	Fast Braking	0	0	1	1
	Low Cost	-1	-2	2	-3
	High Strength	0	0	-1	0
	Low Deflection	-1	0	1	-2
Total improvement		-2	1	7	0

(b)

Figure 11. (a) Attribute Improvement Matrix
(b) Attribute Improvement Matrix without considering change propagation

By comparing Figure 11a and Figure 11b, it can be seen that the performance ratings for most design features are different after considering change propagation. In some extreme cases, the introduction of a primary design feature can even decrease the performance of a primary attribute. For instance, in this example, it can be seen from Figure 11a that introducing larger wheels to the bicycle does not increase the maximum cycling speed as such a change can propagate to other design features and affect the overall performance of the attribute. This observation is important as it suggests that ignoring the effects of change propagation in the concept selection stage might result in choosing a design feature that might produce counter-intuitive results. This can result in a series of unexpected changes attempting to correct and match the design to the required product attributes. The consequences can be disastrous if the amount of unexpected changes exceeds the available resources.

The inclusion of the secondary attributes in the Attribute Improvement Matrix allows designers to assess the overall performance of each design feature even though these attributes are not the focus of the new project. For example, having more gears was not envisaged to influence the speed of braking. However, by considering change propagation, having more gears can lead to a larger frame to accommodate the new gears and in turn reduce the speed of braking due to higher inertia. Therefore, unintended change propagation to the secondary attributes can also undermine the success of the new product and must be included for a comprehensive analysis.

6. Conclusion

This paper introduced an approach to consider change propagation during the concept selection phase. The approach was applied on a bicycle case example to demonstrate the feasibility of such a framework. The results indicate that the performance rating of design features can be different once change propagation is taken into account. This implies that the selection of design concepts can be influenced by using this approach. The findings in this paper also provide an indication that ignoring change propagation in concept selection can result in project delays due to unexpected changes. In this paper, we assumed that only one design feature will be selected to improve the product. This might not be the case during all design projects. Future work will include the analysis of attribute performance when selecting more than one design feature during the concept selection phase. The current approach is also limited to the assessment of attribute performance and does not consider the impact of implementing new design features in terms of redesign effort. More research is required to link this information with attribute performance to direct designers' attention to the tradeoffs between different design features.

References

Browning, T. R., "Applying the Design Structure Matrix to system decomposition and intergration problems: A review and new directions", IEEE Transactions on Engineering Management, Vol 48, No. 3, 2001, pp. 292-306.

Clarkson, P. J., Simons, C. and Eckert, C. M., "Predicting change propagation in complex design", ASME Journal of Mechanical Design, Vol 126, No. 5, 2004, pp. 765-797.

Conrad, J., Deubel, T., Kohler, C., Wanke, S. and Weber, C., "Change impact and risk analysis (CIRA) – Combining the CPM/PDD theory and FMEA-methodology for an improved engineering change management", Proceedings of ICED'07, Paris, France, 2007.

Eckert, C. M., Clarkson, P. J. and Zanker, W., "Change and customisation in complex engineering domains", Research in Engineering Design, Vol 15, No. 1, 2004, pp. 1-21.

Gero, J. S., "Design Prototypes: a knowledge representation schema for design", AI Magazine, Vol 11, No. 4, 1990, pp. 26-36.

Hauser, J. R. and Clausing, D., "The House of Quality", Harvard Business Review, Vol 66, No. 3, 1988, pp. 63-73.

Ollinger, G. A. and Stahovich, T. F., "RedesignIT- A model-based tool for managing design changes", Journal of Mechanical Design, Vol 126, No. 2, 2004, pp. 208-216.

Otto, K. and Wood, K., "Product Design, Techniques in reverse engineering and new product development", Prentice-Hall, New Jersey, USA, 2001.

Pahl, G. and Beitz, W., "Engineering Design- A systematic approach", 2nd Ed., Springer, London, UK, 1996.

Pugh, S., "Total Design: integrated methods for successful product engineering", Addison Wesley, 1990.

Ulrich, K. T. and Eppinger, S. D., "Product design and development", McGraw-Hill, New York, USA, 1995.

Shah, J. J. and Mantyla, M., "Parametric and feature-based CAD/CAM: concepts, techniques, and applications", Wiley, New York, USA, 1995.

Weber, C., Werner, H. and Deubel, T., "A different view on product data management/product life-cycle management and its future potentials", Journal of Engineering Design, Vol 14, No. 4, 2003, pp. 447-464.

Edwin Chan Yang KOH
PhD Research Student
University of Cambridge, Engineering Design Centre
Trumpington Street, Cambridge, CB2 1PZ, United Kingdom
Tel.: +44 1223 766955
Fax.: +44 1223 332662
Email: yek2@cam.ac.uk
URL: http://www-edc.eng.cam.ac.uk