STUDY ON A DETERMINATION OF DESIGN POLICIES FOR SOLAR-BOATS WITH DIFFERENT DESIGN PHILOSOPHIES

Oizumi, Kazuya; Aoyama, Kazuhiro
The University of Tokyo, Japan

Abstract

As complexity of products increased, management of product design becomes difficult. For better management of product design, a method for the determination of design policy based on extended quality function deployment (QFD) was proposed. To realize computer-aided support for designers to determine design policy, the method makes use of a product model that can be retrieved from the extended quality function deployment table. In this paper, the proposed method was applied to two solar-boats whose design philosophies are different. One is to be designed for sprint race while the other is for endurance race. Comparison of the suggested design policies is discussed whether the difference properly captures change of design policies. As a result, it is verified that proposed method can satisfactory suggest design policies.

Keywords: Decision making, Design management, Project management, Design policy, Quality function deployment

Contact:
Dr. Kazuya Oizumi
The University of Tokyo
Department of Systems Innovation
Japan
oizumi@m.sys.t.u-tokyo.ac.jp

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 20th International Conference on Engineering Design (ICED15), Vol. nn: Title of Volume, Milan, Italy, 27.-30.07.2015

1 INTRODUCTION

Complexity of a product is increasing due to today's diversified market needs and introduction of new technologies. As products become complex, number of recalls are increasing and product developments often face huge delay. Thus, management of product design needs further sophistication.

To manage a design process, holistic understanding of a product is essential. However, as many designers/engineers who have different aspects are involved in the design process, it is difficult to capture the holistic view of the product. While the design process, designers coordinate design of ach part through communications with other designers. In case a product is quite complex, it becomes difficult for designers to have enough communications. As a result, it may cause inconsistencies among parts and ends up in product failure. To avoid such situations, in which direction each part should be designed (in this paper, defined as 'design policy') needs to be deliberated and agreed before actual design of each part starts.

As designers experience a number of product designs, coordination among parts are optimised like self-organization. This type of knowledge is often shared as design standards. Howsoever, when a design philosophy of a product is drastically changed due to introduction of new technologies or new concepts, such understandings that is constructed through the past experiences could be inadequate for the new design. Therefore, logical method to determine design policy for each part should be discussed.

In this paper, agreement of design policy at the very beginning of the embodiment design stage is focused. To support the determination of design policy for each part of a complex product, this paper demonstrates the use of computational method to deduce design policies, which has been proposed in (Oizumi et al. 2015). By deducing rough direction on how to design each part, this method helps a design team to build the agreed design policies. As the formal method elicits the design rationales, reasons of design policies given to parts can be explained rather objectively compared to implicit knowledge.

2 APPROACH

This paper demonstrates the use of a method that makes use of the information depicted in a house of quality (a table described in quality function deployment (Akao 1990)), proposed in (Oizumi et al. 2015). (Oizumi et al. 2015) extended the house of quality so that trade-offs within a product can be captured in a functional aspect. In this paper, how design could be affected by the change of product design philosophy is discussed by the use of the proposed method.

3 REVIEW

Quality function deployment (QFD) has been studied for a long time (Chan and Wu 2002) and applied in real design processes. In this chapter, several studies on QFD, its application to the establishment of design policy, and other approaches are reviewed. Then, the academic position of the method proposed in (Oizumi et al. 2015) is stated.

3.1 Related works

QFD was proposed by (Akao 1990, Ofuji et al. 1990, Nagai and Ofuji 2008). The basic concept of QFD is to deploy customer requirements to several metrics in engineering domains. In original QFD, voices of customer are deployed to required qualities then to parts and production processes (Akao 1990). The deployment is conducted on a duality chart. For example, importance (vector) of required qualities is deployed to quality characteristics by means of linear mapping, which calculates importance (vector) of quality characteristics. Likewise, importance (vector) of voices of customer is reflected to importance of parts characteristics, production processes and so on. Several methods that deploy to technologies, reliabilities, or costs are proposed as well (Chan and Wu 2002). This concept can be regarded as a way to reflect a picture of a certain domain to another domain. As possible application area of QFD is wide, several extensions are proposed (Chan and Wu 2002).

There are different approaches to support design process management by means of a matrix that depicts relationships within a product. For example, (Eckert 2004) proposed a method to grasp how a change on a certain part propagates to other parts. By forecasting how changes propagate in a product, change planning could be sophisticated. (Fei 2011) proposed to construct a structured repository of design changes. Based on the repository, it is possible to plan next design efficiently.

3.2 Position of the applied method

Unlike traditional QFD, the method to determine design policy of each part proposed by (Oizumi et al. 2015) constructs a product model, which can deal with not only matrix calculations but also logical operations. Logical operations can sophisticate computational design management (such as QFD, DSM, and MDM). The proposed method expanded traditional QFD to be able to suggest design policy further in detail by the introduction of logical operations.

In comparison with the change prediction method (Eckert 2004), the proposed method does not deal with change propagation but deduces a direction of change of each design parameter (increase/decrease). Because the proposed method extended QFD, it is rather easier to acquire data. Thus, the method does not require data tailored exclusively to the method or any kind of huge repository like (Fei 2011). Though it might not be precise as (Fei 2011), the easiness of use is an advantage of this method, in case the design team is accustomed to make QFD. This paper demonstrates how the method can help a design team to determine and agree rough design policy.

4 EXTENDED HOUSE OF QUALITY AND PRODUCT MODEL

In this chapter, required information for the determination and agreement of design policy is stated. In regards to the required information, extension of QFD and structure of a product model is discussed.

4.1 Prerequisite for product model in regards to determination of design policy

Nowadays, improvement design, improvement to a product which has already been designed, accounts for a large part of product designs. In case of improvement design, design policy to be determined and agreed is 'What extent which required metrics should be improved by which design parameter of which part.' The proposed method gives how superior/inferior each required metrics are taken into consideration upon changing a specific design parameter. Thus, the problem is, in short, deducing whether each relationship between requirement and design parameter (a cell in a house of quality) should be operated for the purpose of improving the specific required metric. To determine the design policy described above, following two attributes needs to be captured.

- What extent each design parameter contributes to fulfilment of requirement
- Whether a design parameter causes trade-offs among requirements.

The traditional QFD technique can not judge whether each design parameter should be increased/decreased because it uses linear mapping as a way to reflect a picture of required qualities to quality characteristics. If a relationship between a required quality and a quality characteristics specifies whether the quality characteristics should be increased/decreased and how much the value should be changed, it is possible to depict trade-offs, which is essential for the determination of design policies. To deal with such information, a product model that can describe rough parametric relationships between required qualities and quality characteristics and their strength.

4.2 Extension for house of quality

To fulfil the prerequisite stated in section 4.1, a duality chart that can describe both specific metrics of required qualities and specific values of quality characteristics that components have. Thus, the house of quality was extended as in Figure 1. In the extended house of quality and a product model introduced in section 4.3, three types of elements are defined as follows (Oizumi et al. 2015).

- <u>Functional metric</u>: Functional metric is a type of required quality that is broken down to a detailed level enough to be able to describe in a value or its substitute value, which can be measured or calculated. Their required values are determined in a product planning and evaluated in a product test phase. An importance value is given to each functional metric.
- <u>Component</u>: A product is composed by components. Intangible things such as functions or behaviours are not included. A component has one or several design parameters (explained below)
- <u>Design parameter</u>: A design parameter is a feature of a component that can be changed as an object of design. Each design parameter belongs to a component.

In a traditional house of quality, strength of relationship is given in numbers (e.g. 1, 3, 9) or signs (e.g. double circle, circle, triangle). In the extended house of quality proposed by (Oizumi et al. 2015), a relationship between a functional metric and a design parameter is explained by attributes listed in

Table 1. How strongly the functional metric is affected by the design parameter is defined as ‘sensitivity’ while whether the parameter should be increased/decreased is defined as ‘characteristics’. Figure 2 shows conceptual chart of characteristics. Four types of characteristics (larger the better, smaller the better, closer the better, and unknown) are defined. Here, ‘larger the better’ means the design parameter should be larger to improve the specific functional metric.
Furthermore, there are several design parameters that should not be changed due to multiple reasons such as standardization or contracts with suppliers. This type of information is given to design parameters as ‘preconditioned’.
As the knowledge that needs to be described in an extended house of quality is usually dispersed across the design team, they should construct the table incorporating insights of several designers.

	Importance	Component A		Component B		…
		Design parameter a	Design parameter b	Design parameter c	Design parameter d	…
Preconditioned			P			
Functional metric A	4	1 ↑	3 ↑↓		3 ↓	…
Functional metric B	3	3 ↓	3 ↑↓	3 ↑		
Functional metric C	2	3 ↓		1 ↑		
⋮	⋮	⋮				⋱

Figure 1. Extended house of quality

Table 1. Levels of attributes associated with a relationship between a functional metric and a design parameter

Sensitivity	(Toward improvement of function …) 1 : not strong enough 3 : strong enough
Characteristics	↑ : Larger the better ↓ : Smaller the better ↑↓ : Closer the better ? : Unknown

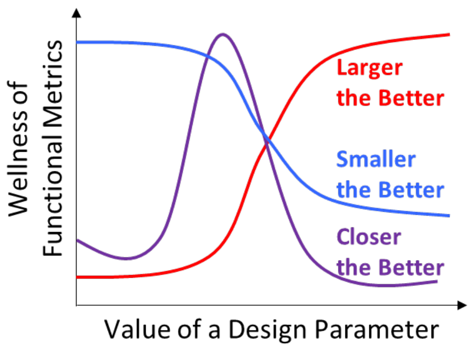

Figure 2. Conceptual chart of characteristics

4.3 Introduction of operational preferences to a product model

A product model shown in Figure 3 is constructed from the extended house of quality (section 4.2). As shown in Figure 3, operational risk and operational preference is introduced to a relationship between a functional metric and a design parameter for the purpose of determining design policy of each component. Their definitions are stated below.

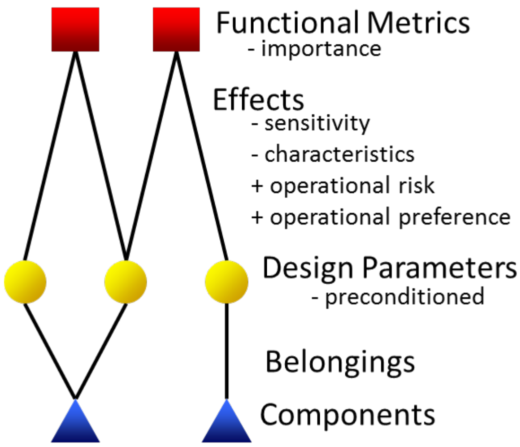

Figure 3.Product model and annotations

- Operational risk: A design parameter may affect several functional metrics that have different characteristics. In this case, when the parameter is changed to improve a specific functional metric, it may cause negative effects on other functional metrics. The possibility of giving negative effects to others is defined as operational risk, which has three possible values (no, low, high). The value is calculated from characteristics of the relationship and importance of functional metrics.
- Operational preference: Operational preference states whether the design parameter should be changed to improve the certain functional metric. Operational preference can have three possible values (preferred, not preferred, controversial). The value is calculated from sensitivity, operational risk, and whether the design parameter is preconditioned. Design policy of each design parameter is expressed as a set of operational preferences (explained in chapter 5).

While information depicted in an extended house of quality needs to be described by designers, operational risk and preference are deduced from the depicted information by means of logical operations explained in chapter 5.

5 DEDUCTION OF DESIGN POLICY

In this chapter, expression of design policies on a product model defined in chapter 4 and its calculation are explained.

5.1 Expression of design policy

Design policy is assumed to be determined to each design parameter. As shown in Figure 4, a design policy of a certain design parameter is expressed as a set of combinations of characteristics and operational preference. For example, in Figure 4, 'the design parameter 1 is preferred to be increased so as to improve the functional metrics B and C, but not preferred to be decreased for the functional metric A nor to be closer to a certain value for the functional metric D' is the design policy for the design parameter 1.

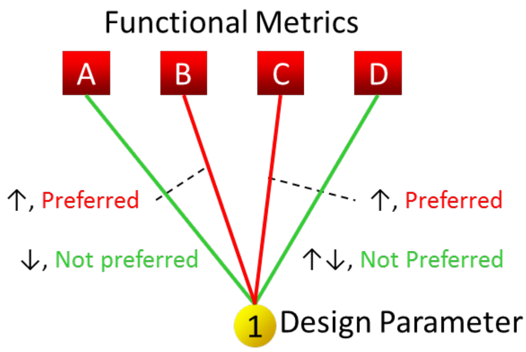

Figure 4. Expression of design policy on product model

Therefore, to determine design policies for design parameters, operational preferences need to be decided. The following sections explain a way to suggest operational preference by means of logical operations.

5.2 Deduction of operational risk

Before suggesting operational preferences, operational risk of each relationship between a functional metric and a design parameter needs to be deduced, as it is one of the essential information for the suggestion of operational preferences. In this section a logical operation to deduce operational risk is explained.

5.2.1 Judgement of trade-off

First of all, whether each design parameter causes trade-offs among functional metrics is judged. In case, characteristics of relationships of a design parameter are all 'larger the better'/'smaller the better', there are no trade-offs as the parameter should be changed to a certain direction. In such a case, trade-off judgement is 'no'. Thus, operational risk as well is judged as 'no', which is stored on all the relationships of the design parameter. In other cases, there are trade-offs.

5.2.2 Judgement of dominant effects

If a design parameter causes trade-offs among functional metrics, dominant effect(s) need to be specified as shown in Figure 5. Dominant effect means that the relationship between a functional metric and a design parameter has the strongest influence on the determination of the design parameter. The dominant effects are relationships whose dominance is maximum, which is calculated by sensitivity multiplied by importance of the functional metric. Characteristics of the dominant effects are defined as a dominant direction. In case, characteristics of the dominant effects are all 'larger the better'/'smaller the better', the dominant direction is 'larger the better'/' smaller the better'. In other cases (e.g. both 'larger the better' and 'smaller the better'), the dominant direction is 'closer the better', as it can be considered that the value of the design parameter is determined in balancing the several functional metrics.

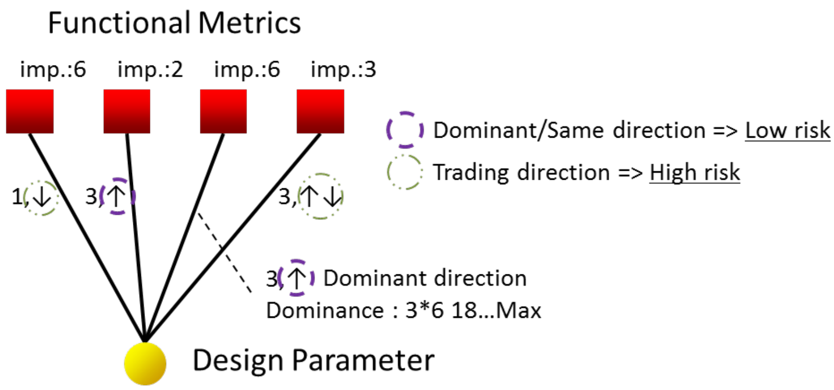

Figure 5. Determination of dominant effects

5.2.3 Judgement of operational risk

For each relationship of a design parameter, in case the relationship is dominant or its characteristics are consistent with the dominant direction, operational risk of the relationship is judged as 'low'. Here, "consistent" means that characteristics of the relationship and dominant direction are both 'larger the better'/'smaller the better'. When a design parameter is changed so as to improve the functional metric, this operation does not cause negative effects on functional metrics whose relationship is dominant. That is why operational risk is judged as 'low'. In case the characteristics of the relationship are inconsistent with the dominant direction, operational risk of the relationship is judged as 'high', because this operation cause or may cause negative effects on functional metrics whose relationship is dominant.

5.3 Suggestion of operational preference

Each relationship between a functional metric and a design parameter has three attributes; sensitivity, operational risk and whether the design parameter is preconditioned. By using these attributes, operational preference, whether the design parameter should be changed in accordance with the direction that characteristics indicate, is suggested. Though three different types of logical operations have been proposed in (Oizumi et al. 2015), one type that is adopted for the following case study is explained on this paper.

The logical operations for the suggestion of operational preference consist of two rounds. In the first round operation, operational preferences of relationships are tentatively deduced. In this round the operation is completed in each design parameter. As for the second round operation, operational preferences deduced in the first round is revised by assessing tentative operational preferences in each functional metric.

5.3.1 First round operation: deducing operational preference in each design parameter

Table 2 indicates the logical operation to deduce tentative operational preferences. If a design parameter is 'preconditioned', operational preferences of all the relationship the design parameter has are suggested as '0 - not preferred'. In contrast, if a relationship has '3 - strong sensitivity', 'no risk', and is 'not preconditioned', the operational preference of the relationship is suggested as '2 - preferred'. When sensitivity is '1 – not strong' and risk is 'no or low risk', the operational preference of the relationship is suggested as '1 - controversial'.

Table 2. First round logical operation to deduce tentative operational preferences

	Not Preconditioned		Preconditioned	
	Sensitivity: 3	Sensitivity: 1	Sensitivity: 3	Sensitivity: 1
No Risk	2	1	0	0
Low Risk	2	1	0	0
High Risk	0	0	0	0

5.3.2 Second round operation: revising operational preference in each functional metric

After the first round, operational preferences are tentatively suggested to all relationships. In the second round, operational preferences are revised so that each functional metric has at least one relationship that is 'preferred' to be operated. Table 3 indicates the logical operation to revise operational preferences. Operational preferences are decided by the maximum and minimum values of tentative operational preferences (in first round) of each functional metric. For example, in case maximum and minimum tentative operational preferences of a certain functional metric are '1' and '0', those its tentative operational preference is 1 is revised as 'preferred'.

Table 3. Second round logical operation to revise operational preferences

For each Functional metric		Operational preference of each effect		
Max	Min	2	1	0
2	0	Preferred	Controversial	Not Preferred
2	1	Preferred	Controversial	N/A
2	2	Preferred	N/A	N/A
1	0	N/A	Preferred	Not Preferred
1	1	N/A	Preferred	N/A
0	0	N/A	N/A	Preferred

6 CASE STUDY: TWO SOLAR-BOATS WITH DIFFERENT PHILOSOPHIES

The proposed method explained above was applied to a case study on solar-boat development projects. In the case study, two different solar-boats whose design philosophies (design policy of a whole product) are different are studied. Design policies are suggested to those two products by the proposed method. Comparison of the suggested design policies is discussed whether the difference properly captures change of design policies.

6.1 Overview of solar boats and their house of quality

A solar-boat was developed in the past. A picture and its rough drawing are shown in Figure 6. In this development project, a house of quality is made as in Figure 7. The house of quality contains 12 functional metrics and 10 components with 23 design parameters.

In this case study, product structure is common within products. Two different design philosophies are depicted as different importance (vector). Importance given to two products is shown in Table 4. Product A is to be designed for sprint races where speed and acceleration are essential competitiveness, while Product B is for endurance races where reachable distance is essential.

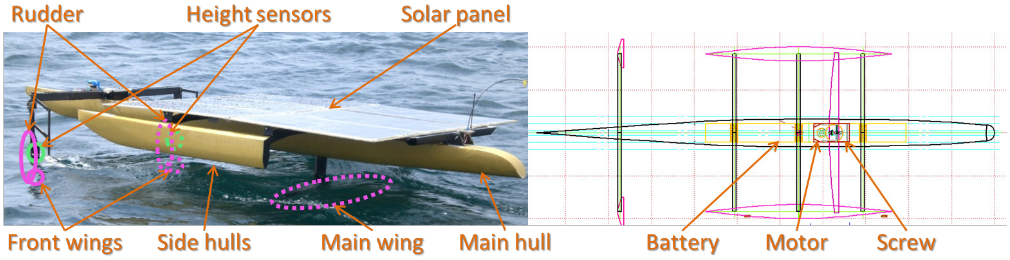

Figure 6. Picture and rough drawing of solar-boat

		part		Main Hull			Side Hull			Main Wing			Front Wing			Height Sensor	Solar Panel		Battery		Motor		Screw		Rudder	
		precondition		●												●				●						
			Importance	Length	Width	Stiffness	Length	Width	Stiffness	Length	Width	Stiffness	Length	Width	Stiffness	Sensitivity	Length	Width	Output	Capacity	Torque	Revolution	Size	Number	Length	Depth
Performance	Speed	Maximum Speed	6	1↓	3↓	1↑	1↓	1↓	1↑	1↓	1↓	1↑	1↓	1↓	1↑	–	–	–	3↑	1↓	1↑	3↑	3↑	3↑	1↓	1↓
		Acceleration	9	1↓	1↓	1↑	1↓	1↓	1↑	3↓	3↓	1↑	1↓	1↓	1↑	–	–	–	3↑	1↓	3↑	1↑	3↑	3↑	1↓	1↓
	Stability	Role Stability high speed	7	–	1↑	1↓	1↓	1↓	–	1↑	3↑	1↑	1↑	3↑	1↑	3↑	1↓	1↓	–	–	1↑	–	–	–	1↑	1↑
		Role Stability low speed	4	–	1↑	1↓	1↑	1↑	–	1↑	3↑	1↑	1↑	3↑	1↑	3↑	1↓	1↓	–	–	1↑↓	–	–	–	1↑	1↑
		Yaw Stability high speed	3	3↑	1↑	1↑	–	–	–	–	1↑	1↓	–	1↑	1↓	–	–	–	–	–	1↑	–	–	–	3↓	1↓
		Yaw Stability low speed	2	3↑	1↑	1↑	3↑	1↑	1↑	–	1↑	1↓	–	1↑	1↓	–	–	–	–	–	1↑↓	–	–	–	3↓	1↓
	Distance	Max reachable Distance low speed	3	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	–	3↑	3↑	3↓	3↑	1↓	3↓	1↓	1↓	1↓	1↓
		Distance in high speed	4	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	–	3↑	3↑	3↓	3↑	1↑↓	1↑↓	1↑↓	1↑↓	1↓	1↓
		Max reachable Distance low speed cloudy	2	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	–	1↓	1↓	3↓	3↑	1↓	3↓	1↓	1↓	1↓	1↓
		Distance in high speed cloudy	3	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	1↓	1↓	–	–	1↓	1↓	3↓	3↑	1↑↓	1↑↓	1↑↓	1↑↓	1↓	1↓
	Curve	Minimum Radius low speed	8	3↓	–	–	3↓	–	–	1?	1?	1↑	1?	1?	1↑	–	–	–	–	–	–	–	–	–	3↑	3↑
		Minimum Radius high speed	4	3↓	–	–	–	–	–	1?	1?	1↑	1?	1?	1↑	–	–	–	–	–	–	–	–	–	3↑	3↑

Figure 7. Extended house of quality for solar-boat

Table 4. Importance (vector) of solar-boats with different design philosophies

	Maximum Speed	Acceleration	Role stability high speed	Role stability low speed	Yaw Stability high speed	Yaw Stability low speed	Max reachable dist. low speed	Max reachable dist. high speed	Max reachable dist. low speed cloudy	Max reachable dist. high speed cloudy	Minimum Radius low speed	Minimum Radius high speed
Product A	6	9	7	4	3	2	3	4	2	3	8	4
Product B	2	3	7	4	3	2	9	6	7	5	8	4

6.2 Comparison of suggested design policies

As an example, a design policy suggested for 'solar panel / length' (dp14) is illustrated in Figure 8. The matrix on left side is a respondent part of the house of quality with suggested operational preferences. A red cell means that the design parameter is preferred to be change so as to improve a functional metric on the row. A green cell is not preferred. Right side of Figure 8 illustrates its graphical visualization. In this case, ‘solar panel / length’ is preferred to be change so as to improve maximum reachable distance in low and high speed, but not be preferred to be changed to improve other four functional metrics.

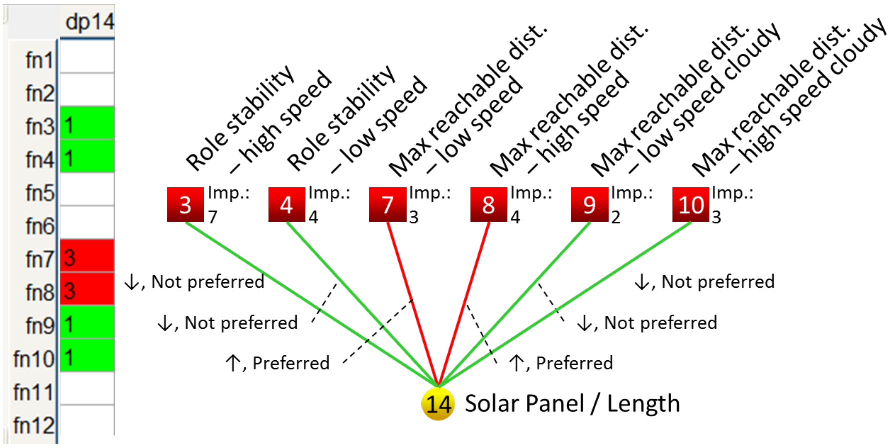

Figure 8. Suggested design policy for solar panel length

Figure 9 shows design policies suggested to both product A and B. It is possible to see that there are several design parameters whose design policies are common between products A and B. It shows that determination of those design parameters are not affected by the difference of design philosophies on this case, because they have minor effects to functional metrics whose importance is changed.

As shown in Figure 10, design parameters whose design policies were significantly different were ‘battery / output’ (dp16), ‘motor / revolution’ (dp19), and ‘main wing/ width’ (dp8). As for ‘battery / output’ and ‘motor / revolution’, their preferred operations are to improve functional metrics that are wanted to be enhanced for each product (speed and acceleration of product A, distance-related metrics for product B). In case of ‘main wing/ width’, a suggested design policy for product B is to improve stability-related metrics, because these metrics become more important than speed and acceleration.

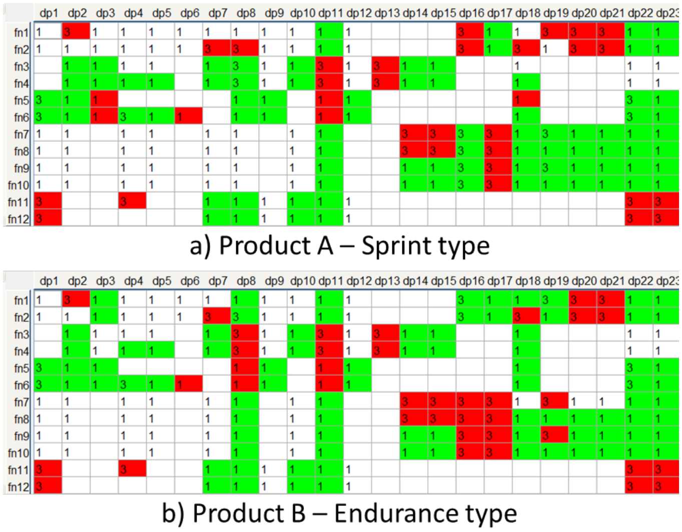

Figure 9. Comparison of design policies deduced for product A and B

6.3 Discussion on the case study

The case study shows that the method to deduce design policy for each part of a product could successfully capture the change of design policies between different design philosophies adopted for two products. When a design philosophy is changed from sprint to endurance, the necessary changes of design policies could be indicated. First, ‘battery / output’ should be changed to smaller one so as to improve maximum reachable distances. Second, ‘motor / revolution’ should be changed to smaller one as well.

In addition, the method could capture the change which is not necessary and tends to be overlooked. Though 'main wing/ width' is not related to the change of design philosophy from sprint to endurance, as product B does not need to improve speed and acceleration, opportunity to improve stabilities emerges. As this kind of change tends to be overlooked, the computational method could be an alert for designers.

As a result, the design policies deduced for two products with different design philosophy match quite well with the thought that the designers have. To validate the proposed method, (Oizumi et al. 2015) conducted a comparison between design policies that the proposed method deduced and that of designers given in a same manner as the proposed method. This comparison showed quite high agreement. The result of this paper supplements the proposed method with wider applicability.

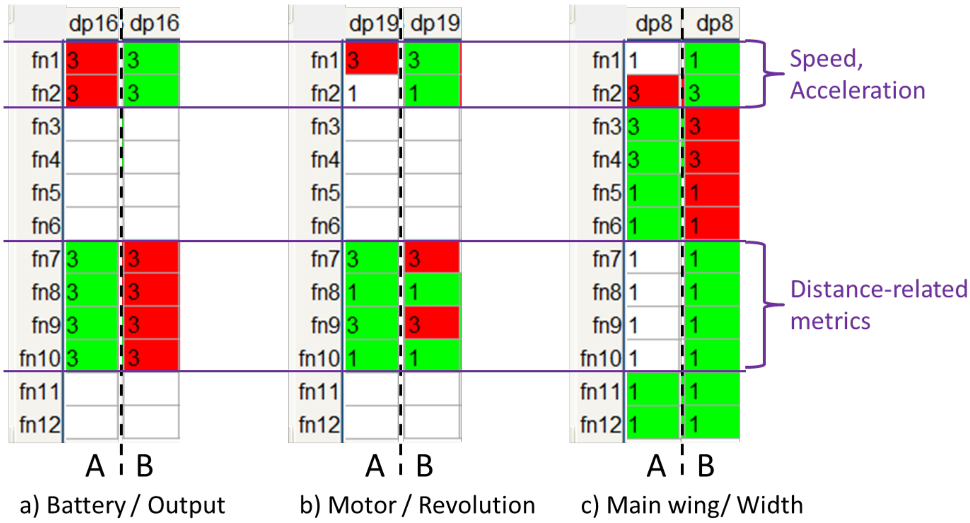

Figure 10. Illustrative difference of design policies between product A and B

7 CONCLUSION

This paper demonstrated a method to deduce design policy for each part of a product proposed in (Oizumi et al. 2015). To evaluate the method, a case study is conducted on two different solar-boats whose design philosophies are different. As a result, the method could capture the change of design policies between different design philosophies successfully. The suggested design policies are quite consistent with the thought of designers.

It means that following statement is proven on this certain case;

- Which direction each design parameter should be changed is influenced by risks that causes negative effects to functional metrics (required qualities) and sensitivities of functional metrics toward the design parameter.

In conclusion, the logical operations that take into consideration of the risk of negative effects and sensitivities can contribute to the determination of design policies. Further, it might be possible to say that the logical operations adopted in this paper reflects how designers think in determining design policies at the early stage of design

REFERENCES

Akao, Y. (1990) Introduction to Quality Function Deployment (in Japanese), Tokyo:Union of Japanese Scientists and Engineers.

Chan, L., K. and Wu, M., L (2002) Quality function deployment: A literature review, European Journal of Operational Research, Vol 143, pp. 463-497

Eckert, C., Clarkson, P. J. and Zanker, W. (2004) Change and Customization in Complex Engineering Domains, Research in Engineering Design, Vol. 15, No. 1.

Fei, G., Gao, J., Owodunni, O. and Tang, X. (2011) A Method for Engineering Design Change Analysis us-ing System Modelling and Knowledge Management Techniques, International Journal of Computer Inte-grated Manufacturing, Vol. 24, No. 6, pp. 535.

Nagai K. and Ofuji T. (2008) The Third Generation QFD (in Japanese),Tokyo:Union of Japanese Scientists and Engineers.

Ofuji T., Ono M. and Akao Y. (1990) Quality Function Deployment (1) (in Japanese), Tokyo:Union of Japanese Scientists and Engineers.
Oizumi K., Koga T., Kitajima K. amd Aoyama K. (2015) Determination of Design Policy Utilizing Information on Quality Function Deployment (in Japanese), Sekkeikougaku (Design Engineering), Vol. 50, No. 3, 2015.