개념적 설계 지식과 프로세스 통합 모델
Integrated Knowledge and Process Model for Conceptual Design

이재현 [1], *,#서효원 [2]
J.H. Lee, H.W. Suh(hwsuh@kaist.ac.kr)
[1,2] 한국과학기술원 산업공학과

Key words : conceptual design knowledge, conceptual design process

1. Introduction

A conceptual design has an evolutionary and iterative process. In conceptual design, design knowledge evolves from incomplete and imprecise knowledge, so that many decision changes are bound to happen in the process. Although a conceptual design is an early stage of a product design, it is the most crucial task in a product development cycle. According to researches in concurrent engineering [1], most of a product life cycle cost is fixed early in its lifecycle. Moreover, a design change or improvement during a conceptual design is comparatively inexpensive. In spite of its influence to overall design stages, a conceptual design knowledge model is not well established considering the knowledge evolution and the iterative process. Although many previous researches [6-13] proposed knowledge (information) models, they did not explicitly describe the relationships between their knowledge model and process model.

Therefore, we propose an integrated knowledge and process model. It includes a process model which can manage the unexpected design changes or side-effects. It also includes a knowledge model (so called a knowledge map) which is developed based on the process model. The map consists of knowledge primitives and their relations. We define their semantics and representations explicitly. In section 2, we provide an analysis of previous researches' knowledge models. Our knowledge model and our process model are defined in section 3 and 4 respectively. Lastly, we summarize this paper and discuss advantages and shortcomings of our model in section 5.

2. Previous Research

Conceptual Design Process Model: According to Chakrabarti and Bligh's work [4], we can classify the evolutionary process model into two types. First type is an *iterative design model*, and second type is a *systematic design model* proposed by Pahl and Beitz [5]. They have different approach to the function and solution design (see [4] for details). Chakrabarti and Bligh [4] said the systematic design model has limitations that it cannot be used to generate the function structure in a useful way without being guided by the knowledge of existing solutions. So we think the iterative design model is more applicable to a practical situation.

In 1990, Gero [2] defined an iterative design process model. The model defined five information types. After the Gero's model, many previous researches in conceptual (functional) design area proposed knowledge models. They supported some activities of a design process or whole of the process. They defined different knowledge primitives (i.e. information types). We analyze the previous knowledge models in the following sub-section.

Conceptual Design Knowledge Model: One of the issues in knowledge (information) modeling research is how to describe the conceptual causal mechanism of a solution. Sembugamoorthy and Chandrasekaran[6] proposed a functional representation, which hierarchically represents a device's functions and behaviors. Iwasaki and et al. [7] extended it and proposed the causal functional representation language (CFRL). McDowell and et al. [8] proposed an elaborated version of CFRL. However, to allow freedom in design and to make the selection of other alternatives, function should be defined as an independent concept from methods, principles, or behaviors [3]. So other previous researches proposed knowledge models that defined a function independently from behavior. Bracewell and Sharpe [9] proposed a model that consisted of functions, components and functional embodiment (means and working principles). Umeda and Tomiyama [10] proposed a Function-Behavior-State (FBS) model. Brunetti and Golob [11] designed an information structure in conceptual design. Roy and et al. [12] proposed a functional model that consisted of function, behavior and artifact. The artifact includes a form that is a collection of sketches, features, tolerances, and materials. Roy and Bharadwaj [13] studied the nature of relationship between function, geometry and part behavior, and they proposed a part function model (PFM) and behavior-PFM map.

We can not say which knowledge model is better than others, but we think a model including more semantically classified primitives is better applicable to define a conceptual design process. Therefore, we propose an integrated model that consists of a design process model as well as a knowledge model.

3. Conceptual Design Knowledge Model

In this section, we describe conceptual design knowledge (CDK) primitives briefly because of the page limit.

Requirement primitive: A requirement is a something that is requested by customers. Initial requirements can be vague and incomplete. We classify requirements into two types according to customer's intention. A proactive requirement is a type of a requirement. It is defined by customer's requests or needs. Another type is a reactive requirement. It is not defined by a customer, but it is requested by an engineer to reduce side-effects of a designing solution.

Function primitive: Engineers generally agree that function is the most important concept in designing product systematically. However, there is no clear and uniform definition of function because function is an intuitive concept depending on the designer's intention.

We define a function based on previous researches [10,11]. We define function as a relationship between a functional verb and I/O. In addition, we discriminate between a function and a behavior. Kitamura and *et al*. [15] defined a behavior of a device as situation-independent conceptualization of the change between input and output of the device. So, we can say that behavior is owned by a device (a solution in our view), but function is not. This ontological difference makes us to design a solution model for a function, not behavior model.

Solution primitive: According to Urich and Eppinger [14], a solution (so called a product concept) is an approximate description of the technology, working principles and form of the product. It is a concise description of how the solution will satisfy the customer requirements.

A solution primitive should be applied into any abstraction level of design. So we classify solutions into two types; abstract solution and physical solution. When an engineer develops a new product, the design begins from an incomplete and rough solution; abstract solution. An abstract solution becomes a detail and precise solution as design proceeds; a physical solution.

We define that a solution is a composite of behaviors, structures, operations, and mfg. processes. The behavior of a solution is situation independent conceptualization of the change between input and output of the solution [15].

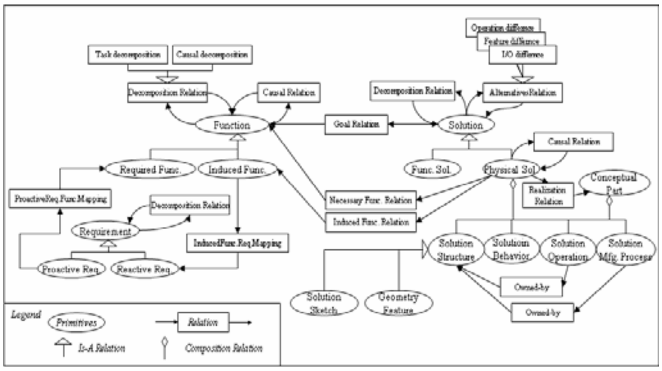

Fig 1. Knowledge Map of Conceptual Design

Conceptual part primitive: We define a conceptual part as a fully designed physical solution which is decided not to search further sub-solutions or required functions by an engineer. Therefore, our conceptual part model has a same representation of the solution model. An artifact model proposed by Roy and et. al. [12] has very similar structure with our conceptual part model. It includes function, form (sketches and features) and behavior information. They did not differentiate two concepts; a developing solution and a developed solution (conceptual part). However, this difference is very helpful for an engineer in practical situations. If an engineer can recognize easily conceptual parts, he can find easily which solutions need to be further developed or not. In addition, a fast recognition for the conceptual parts helps an engineer to reuse them to develop other solutions.

Figure 1 shows all primitives and their relationships for our knowledge model, and we call it as a knowledge map. An ellipse represents a primitive concept, and a square represents a relationship.

4. Conceptual Design Process Model

Since many decision changes are bound to happen in a design processes, a conceptual design process is a complex and evolutionary process. One of the decision changes is provoked by unexpected side-effects of a designed solution. So we design a process model that allows a design change flow for the side-effects. We define a conceptual design process as a set of design flows (i.e. activities) between the CDK primitives; requirement, function, solution, and part concept. The requirement, the function and the solution are further classified to specify the design change flow. The design flow between the primitives is a design activity. We define nine design activities.

In the specification activity (A0), an engineer defines customer’s requirements and its attributes. If he can divide the requirements without considering functions or solutions, define sub-requirements. Then, he performs the formulation activity (A1) to define required functions for each requirement. Next, he designs a solution for each required function. This is a synthesis activity (A2).

For a designed solution, an engineer should specify its behavior, structure, operation, or mfg. process through analyzing the solution (A3). So a functional solution or a physical solution can be decomposed into sub-solutions. While performing analyzing a solution, auxiliary functions might be required to make the solution work well. If the auxiliary function is required, an engineer performs the re-formulation I activity (A4-1). If he defines auxiliary functions, he has to design solutions for the functions. So he should perform again the synthesis activity for the auxiliary functions.

Meanwhile, an engineer could define (unexpected) side-effects of a solution as induced functions. We call this activity as the re-formulation II activity (A4-2). For the induced functions, he has to define reactive requirements to define constraints for a solution selection. So we call this activity as a re-specification activity (A5).

While an engineer analyzes a solution or synthesizes solutions, he has to select a solution among many alternative solutions. The requirements are measurements of the solution evaluation (A6). If a physical solution is developed enough to satisfy all requirements, he defines the solution as a part concept. We call this activity as a description activity (A7) because it generates a description of the part concept.

5. Conclusions

We design an integrated knowledge and process model that supports an evolutionary and iterative design process. The knowledge model can be utilized for a development of a conceptual design support system. The model has a process model and a knowledge map. The process model includes the management of unexpected change of specifications as well as traditional conceptual design activities. We define a knowledge map to support the process model. It has CDK primitives and relations among them. We specify the definitions of the primitives considering their roles in the process model. We do not argue that the proposed knowledge map can cover all activities which are relevant to conceptual design. However, the map can be easily expanded or modified to adapt into a new design environment, because its primitives are divided by clear semantics.

Acknowledgement

We acknowledge KITECH which funded this work under the contract “Web-based SMART manufacturing system.”

Reference

1. Cyan, CS. and Menon, U., Concurrent engineering: concepts, implementation and practice. Springer 1994.
2. Gero, JS., “Design prototypes: a knowledge representation schema for design,” AI Magazine, **11**(4), 26–36, 1990.
3. Kiatamura, Y., Kashiwase, M., Fuse, M., and Mizoguchi, R., “Deployment of an ontological framework of functional design knowledge,” Advanced Engineering Informatics, **18**, 115-127, 2004.
4. Chakrabarti, A. and Bligh, TP., “A scheme for functional reasoning in conceptual design,” Design studies, **22**(6), 493-517, 2001.
5. Pahl, G. and Beitz, W., Engineering design, a systematic approach, 2nd ed. London: Springer, 1996.
6. Sembugamoorthy, V. and Chandrasekaran, B., “Functional representation of devices and compilation of diagnostic problem-solving systems,” Experience, Memory and Reasoning. Lawrence Erlbaum Associates, Hillsdale, NJ, 1986.
7. Iwasaki, Y., Fikes, R., Vescovi, M., and Chandrasekaran, B., “How things are intended to work: capturing functional knowledge in device design,” In: proceedings of IJCAI, 1516-1522, 1993.
8. McDowell, JK., Lin, Y., Zhou, K., and Lenz, TJ., “Conceptual Design for polymer composite assemblies,” AI system Support for Conceptual Design, Springer-Verlag, 377-389, 1996.
9. Bracewell, RH., and Sharpe, JEE., “Functional descriptions used in computer support for qualitative scheme generation – schemebuilder,” Artificial intelligence for engineering design, analysis and manufacturing, **10**(4), 333-345, 1996.
10. Umeda, Y. and Tomiyama, T., “Functional Reasoning in Design,” IEEE expert, **12**(2), 42-48, 1997.
11. Brunetti, G. and Golob, B., “A feature-based approach towards an integrated product model including conceptual design information,” Computer-Aided Design, **32**, 877-887, 2000.
12. Roy, U., Pramanik, N., Sudarsan, R., Sriram, RD., and Lyons, KW., “Function-to-form mapping: model, representation and applications in design synthesis,” Computer-Aided Design, **33** 69-719, 2001.
13. Roy, U. and Bharadwaj, B., “Design with part behaviors: behavior model, representation and applications,” Computer-Aided Design, **34**, 613-636, 2002.
14. Ultich, KT. and Eppinger, SD., Product design and development, McGraw-Hill, Third edition 2003.
15. Kitamura, Y., Sano, T., Namba, K., and Mizoguchi, R., “A functional concept ontology and its application to automatic identification of functional structures,” Advanced engineering informatics, **16**(2), 145-163, 2002.