

UXDs-Driven Contradiction Solving For Conceptual Design Using CAIs

Tan Runhua

Institute of Design for Innovation, Hebei University of Technology,
Tianjin, 300130, P.R. China
Email: rhtan@hebut.edu.cn

Abstract. Design is situated, which means the explicit consideration of the state of environment, the knowledge and experience of the designer and the interaction between the designer and the environment during designing. Central to the notion of situated design is the notion of design situation and constructive memory. When Computer-aided innovation systems (CAIs) are applied in the design, the environment and the situation are different from the traditional design process and environment. The basic principles of some CAIs in the world market are directly related to theory of inventive problem solving (TRIZ). Special TRIZ solutions, such as 40 inventive principles and the related cases, are medium-solutions to domain problems. The second stage analogy process is used to generate domain solutions and in this process the TRIZ solutions are used as source designs of analogy-based process. Unexpected discoveries (UXDs) are the key factors to trigger designers to generate new ideas for domain solutions. The type of UXDs for the specific TRIZ solutions is studied and an UXDs-driven contradiction solving for conceptual design is formed. A case study shows the application of the process.

Keywords. unexpected discovery, Contradiction solving, Conceptual design, computer-aided innovation

1 Introduction

A design situation models a particular state of interaction between a design agent or a designer and the environment at a particular point in time [1]. Memory construction occurs whenever a design agent or a designer uses past experiences and knowledge within the current design environment in a situated manner [2]. When computer aided innovation systems (CAIs) [3] are applied in design process the design situation becomes specific. The interactions mainly happen among designers and a serial of interfaces produced from CAIs.

The basic theory or method of some important existing CAIs [3] is TRIZ, that is, theory of inventive problem solving [4]. TRIZ is developed by analyzing and inducing a great deal of patents within the world database. TRIZ has put forward the concepts, models and tools of inventive problem solving. But the TRIZ special solutions, such as selecting forty or less inventive principles from all the forties, are not domain solutions needed by designers. It's still a problem how to convert the TRIZ special solutions into domain solutions when TRIZ is applied.

One of the converting processes is an analogy-based process [5]. Unexpected discoveries (UXDs) [6] are the key factors to trigger designers to generate new ideas for domain solutions during this process [5]. How to find UXDs from the TRIZ special solutions is becoming an important step for obtaining the domain solutions.

There are a few inventive problems to be solved in TRIZ, such as technological maturity mapping, technology evolution, function solving, contradiction solving etc. But the contradiction solving is the most important kind of problems to be solved and it is rooted philosophy in TRIZ [7]. This study will be restricted to find solutions for this kind of inventive problems. The types of UXDs from the TRIZ special solutions will be studied. And an UXDs-driven analogy based process for conceptual design will be formed.

2 Design situation for contradiction solving using CAIs

The development of different CAIs based on TRIZ has made the TRIZ more applicable and practical. There are one or more knowledge bases in CAIs, in which much knowledge is abstracted from the world patent bases. The knowledge is arranged by the framework of TRIZ.

Fig. 1, which shows contradiction solving method in TRIZ, is developed for explaining the principle of CAIs. There are two parts in the figure, TRIZ world and outside world. In the outside world, the designers find a domain technical contradiction for a domain problem and input it into the TRIZ world. In TRIZ world the contradiction is firstly to be transformed into a standard contradiction using 39 engineering parameters in TRIZ and then forty or less inventive principles are selected through the matrix. Also, some design cases following the principles are contained. The selection of these principles and the cases with these principles are TRIZ special solutions. The design cases are the results of analyzing patent bases from outside world. The TRIZ world in Fig. 1 has been programmed as a kind of arithmetic and a module of CAIs, such as in the Goldfire Innovator and InventionTool, which is then developed. Interaction between TRIZ world and outside world is realized by the interfaces of the CAIs.

The knowledge, which is tacit in different domains of a patent base, is difficult to be applied by designers because it is a problem to find a useful one in different domain. If a patent abstracted in any domain is stored in the case base of TRIZ it becomes explicit knowledge or codified knowledge which can be easily found and applied for idea generation. In the knowledge base of CAIs, a case is described using a sketch with text to explain the working principle of that sketch. When one principle as a TRIZ special solution is selected all the cases relevant to that principle can be browsed one by one. New ideas for the domain solutions may be formed from

designers' mind during the browsing process. Fig. 2 shows the model of design situation for this specific environment.

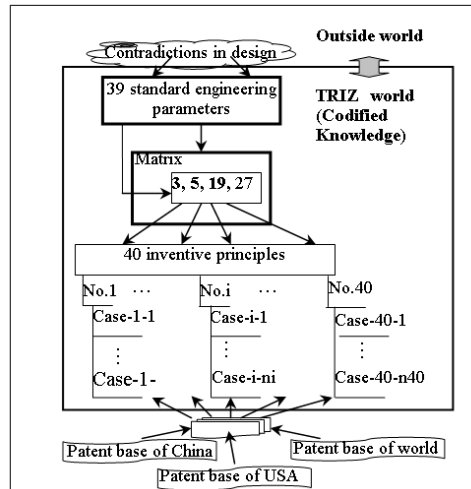


Fig. 1. Contradiction solving model in TRIZ

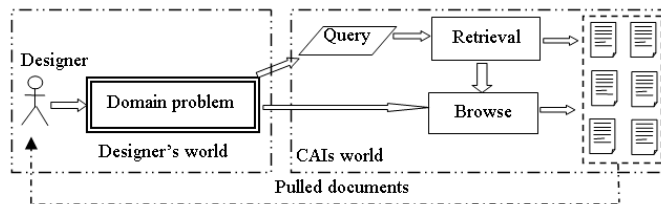


Fig. 2. Design situation using CAIs for technical contradiction

In the designer's internal world, the selected principles and cases as codified knowledge are apperceived, and some cues are found, which trigger for designers to form domain solutions. In Fig. 2, the creative ability for designers, that is, ability to produce ideas, should be increased.

3 A macro-analogy-based process Using CAIs

Analogies are partial similarities between different situations that support further inference. Analogy-based design (ABD) means the application of analogy to design. The ABD is used not only for the normal traditional design, but also for the innovative and creative design [8]. In the process of ABD, the existing designs and the designs to be carried out are source designs and goal designs respectively. One of

the conditions to carry out ABD is the existence of source or base designs in different domains in large number.

Designers find domain problem and then convert them into TRIZ problems, such as contradictions. Then, the TRIZ special solutions are determined from TRIZ, such as using the matrix for contradiction solving. The TRIZ special solutions are not the domain solutions needed by designers. Last, the designers themselves convert TRIZ special solutions into domain solutions. There are two mappings in this process, from domain problem (DP) to TRIZ special solution (TSS) and from TRIZ special solution to domain solution (DS). Both of the two mapping process are analogy processes. The first mapping process is called the first stage analogy process (FSAP) and the second mapping is called the second stage analogy process (SSAP) [5]. The first stage analogy process is completed by the application of CAIs and the outputs are TRIZ special solutions. The second stage analogy process is a human-based process.

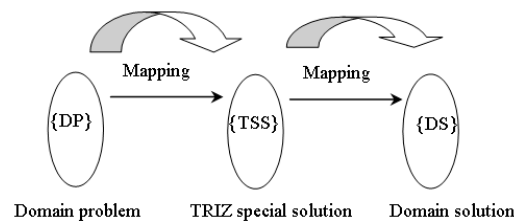


Fig. 3. Three domains of the design world using TRIZ

Suwa and Gero [9] have developed a concept “situated-invention (S-invention)”, which means a designer generates the issue or requirement for the first time in the current design task in a way situated in the design setting. Gero et al. have studied the generation of S-invention, and summarized a design process [6, 8, 10, 11]. Firstly, the design agents apperceive the domain problem and determine source design and goal design. Then design agents find unexpected discoveries (UXDs) through the matching of source design and goal design. UXDs are transferred to goal design by mapping, and new goal is generated. Then modified goal design is produced. There may be multi-source designs, and the last modified goal design is the concept of solving domain problems through modifying goal design continually.

Gero and his group are major in architectural design. So the source designs are drawings of different kinds of architectures. If the source designs are substituted by TRIZ special solutions and the cases corresponding to them the design process for generation of S-invention can be applied to generate the domain solutions. Designers find several UXDs and modify goal design depending on their design experience, the comprehension of domain problems and the situation. At last some modified goal designs are domain solutions. The macro-process of ABD for contradiction solving using TRIZ is shown in Fig. 4.

The contradiction analogs are the standard contradictions selected from the 39 engineering parameters, which have similar meanings to the domain contradictions. The UXDs about solving contradiction analogs are found from the principles solving contradiction analogs and cases having solved contradiction analogs.

The main processes of implementation for the process of Fig. 4 are how to find UXDs about solving the contraction analogs and convert these UXDs into ideas for solving the domain contradictions. UXDs enlighten designers on invention and make new concepts or ideas appear, so discovering and transferring UXDs are the key to the success. According to the concept of constructive memory [2], the memory is not direct reappearance of former experience but a function of former experience, which changes after producing these experience and situation of memory requirement. An UXD is a “new” perceptual action that has a dependency on “old” physical action(s) [9]. This means that if a designer traces or pays attention to the existence of source designs, the perceptual action is an instance of UXD. Experiences and UXDs drive designers to generate new concepts or ideas. The new concepts are mapped to the goal design to produce a new goal design.

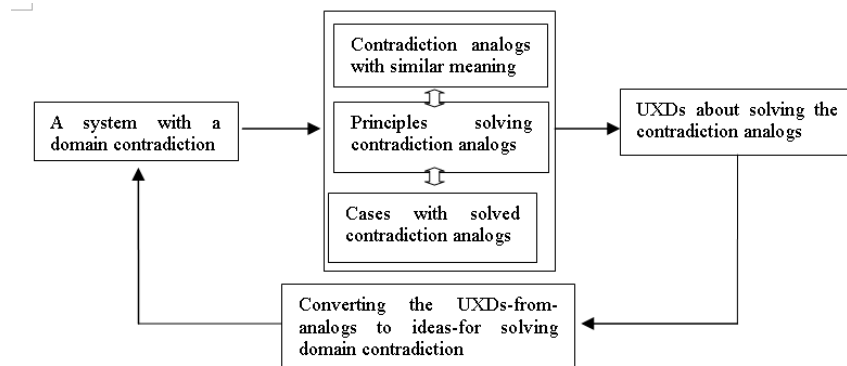


Fig. 4. The macro-process of ABD for contradiction solving using

Perceptual actions [9] are operated for architectural drawings. They must be extended to TRIZ special solutions and cases for TRIZ based design.

4 Types of UXDs for contradiction solving

Currently, there are several well-known design theories and methodologies developed for general (macro-scaled) systems, such as Systematic Design Methodology [12], Axiomatic Design Theory [13] etc. Essentially, the design knowledge representation in these theories is based on the Function-Behavior-Structure (FBS) model [14-17]. There are different kinds of functions, behaviors and structures [18-21]. Functions are divided into atomic, source, destination and transfer functions. And behaviors are divided into three kinds, continuous-time-behavior, discrete-time-behavior and state-transition-behavior. According to the specific design context, a structure may refer to a sub-system, a sub-assembly, a component, a feature, or a geometric entity, and a physical relationship.

For new designs there will be a transition from functional model to a structure model at some point during design process. The transition is called mapping. There

are two kinds of mapping, function-structure mapping and function-behavior-structure mapping. The former is suitable for the mapping of extrinsic functions and the latter is suitable for intrinsic functions.

In order to make new ideas for innovation, the designers need to apply the knowledge of different domains, especially with which the designers are unfamiliar. Generally, the first step for latent in any problem solving is the definition of a space in which the solution of the problem is believed to exist. This solution space is shaped through the requirements, restrictions and constraints imposed on the situation by the problem-solver interpreting user and company needs and functional, structural and intentional. When TRIZ is applied, TRIZ special solutions are sources to extend solution spaces including extended function spaces, behavior spaces and structure spaces, as shown in Fig. 5.

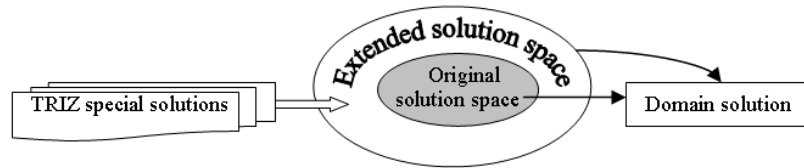


Fig. 5. Extended solution space

Designers generate domain solutions under the search in original and extended solution spaces. The core cues to designers are functions, behaviors or structures implicated in the TRIZ special solutions. The original solution space that the designers have known is used as a background for the domain solutions. The reason is that the designers could not generate new solutions using the original solution space that they have known.

Suwa and Gero [9] have divided UXDs into three types, depending on what types of visuo-spatial feature that the designer discovers. One is the discovery of a visual feature such as shape, size or texture of a previously-drawn element. The second is the discovery of a spatial or organizational relation among more than one previously-drawn element. The third is the discovery of a space that exists in previously drawn elements. The types are suitable for design of an architecture, in which the basic elements are dots, lines, rectangles, circles, arrows and so on. For complex system design, such as complex mechanical system design, the basic elements are more complex. The types divided are not suitable for them. New types are needed.

Because function, behavior and structure are basic knowledge representations and they are also major UXDs implicated in the TRIZ special solutions. For contradiction solving using TRIZ and CAIs, the UXDs are divided into four types that designers discover from the CAI interfaces. Table 1 shows the types, definition of each type, and the instances of each type and how to find an UXD.

Table 1. Types of UXDs

Types	Definition	Instances	How to find
UXD-1	An inventive principle which is suitable for solving the contradiction faced	One to four inventive principles	Check the matrix using two engineering parameters
UXD-2	A function that one case implicated	atomic, source, destination and transfer functions	Physical actions
UXD-3	A behavior that one case implicated	continuous-time-behavior, discrete-time-behavior, state-transition-behavior	Physical actions
UXD-4	A structure that one case implicated	A sub-assembly, a component, a feature, or a geometric entity, and a physical relationship.	Physical actions

For the designers using CAIs, the physical actions to find a UXD are only looking, which means designers looking at the computer screen: the principles of contradiction solving and the cases showing by pictures and contexts, shown in Fig. 2. By looking actions designers discover the UXDs implied.

5 UXDs driven analogy based design using TRIZ

A new idea is not direct reappearance of a designer's former experience but a function of former experience, changes after producing these experience and situation of memory requirement. A new idea is a memory constructed by some stimulus from the situation faced by designers. After the first stage of analogy process using TRIZ four or less inventive principles and related cases are at hand. Designers analyze the principles and cases through physical actions and find some UXDs of different kinds. Now the situations faced by designers are these UXDs. The designers construct new memories under stimulus of UXDs, and as result a domain solution is constructed suddenly. Fig. 6 shows the process which is driven by UXDs analogy process.

According to the Fig. 6, the process is divided into 7 steps, which are as following:

- Step 1: Identify domain problem. Analyze social needs, customer needs or products existed and identify domain problems in the form of contradictions.
- Step 2: Find TRIZ special solutions. Under the situation CAIs find principles and relevant cases.
- Step 3: Identify primary goal design. It can be formed from the existed products.
- Step 4: Find UXDs. The physical actions are performed for the principles and cases.
- Step 5: Transfer UXDs to goal designs. Generate constructive memories and modify goal designs.
- Step 6: Make a judgment. If modified goal designs do not satisfy the needs or the numbers of satisfied goal designs are scarce, turn to step 4, whereas the step continues.
- Step 7: Following design: evaluation, embodiment and detail design.

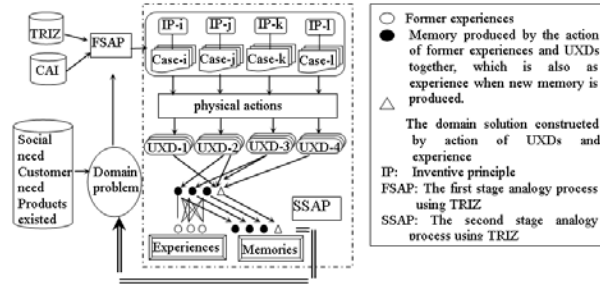


Fig. 6. A process of UXDs driven analogy based design

The step 2 belongs to the first stage analogy process and the step 3 to 6 belongs to the second stage analogy process. The evaluation is included in step 7, and many methods have been developed for this application. The result of the second stage analogy process may be a domain solution or multi-solutions. For the multi-solutions, it needs to determine one or a few domain solutions through evaluation. The result of detailed design is design files, drawings or data files, which are applied for manufacturing.

6 Case study

Dropping pills, produced by dropping pill machines, are a kind of Chinese traditional medicine. After dropping and drying, they should be put into little bottles for selling in the market. The machine to put the pills into bottles is a kind of package machines. There are no standard machines of this kind. A few machines have been developed by one or two firms in China. But new principles for the kind machines are needed by firms of medicine production.

The main functions of the machine are distributing pills, discharging pills, bottling and lidding. Here, the concentration will be on the ideas generation for structure of discharging pills.

Step 1: Identify domain problem. The function of distributing pills is implemented by a structure of tumbling cylinder shown in Fig. 7. The pills are distributed in the several circles. The implementation of discharging pills should be based on the principle of distributing pills. Fig. 8 shows a principle for discharging pills being currently used. If the roller is redesigned and used inside the cylinder a transmission mechanism is also needed and that is not easy. So a contradiction is between adaptability (No.35) and complexity of a device (No.36).

Step 2: Find TRIZ special solutions. InventionTool3.0, which is CAIs, is applied in this step. There is a model in the system which is contradiction solving. Select the improved parameter 'adaptability' and worse parameter 'complexity of a device', then, the interfaces show the TRIZ special solutions, which are No.29(Pneumatic or hydraulic construction), No.15(Dynamicity), No. 28(Replacement of mechanical system), No. 37(Thermal expansion). The four principles and the relevant cases in the case base are the TRIZ special solutions.

Step 3: Identify primary goal design. The Fig. 7 is identified as the starting point.

- Step 4: Find UXDs. By browsing the cases the designer may find several UXDs and generate several ideas under the stimulants of UXDs. Fig. 8 is a case which shows the principle of a component used in a kitchen machine. When working, the ball moves up and down and releases the exhausted gas produced during frying. The principle implies an UXD, which is the ball moves under the pressure of exhausted gas. The UXD is a kind of behavior which is an UXD-3 in table 1.
- Step 5: Transfer UXDs to goal designs. Convert the UXD into the new ideas and generate a principle for discharging pills. The pill moves under the pressure of air flow if a pill is the ball of Fig. 9. A possible structure for this principle is shown in Fig. 10.
- Step 6: Make a judgment. More ideas may generated by finding more UXDs. The structure shown in Fig. 10 is usable.
- Step 7: Following design. Fig. 11 is the conceptual design in which the four main functions of the machine are implemented.

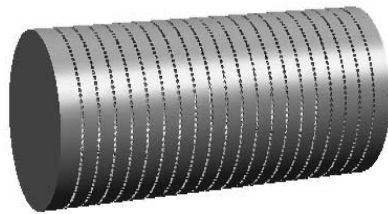


Fig. 7. The cylinder for distributing pills

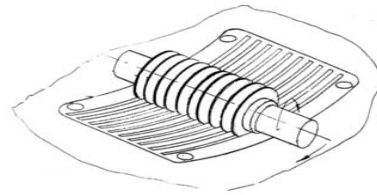


Fig. 8. A principle for discharging pills

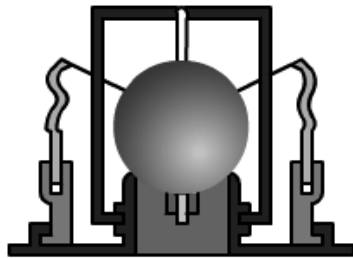


Fig. 9. A case in No.15

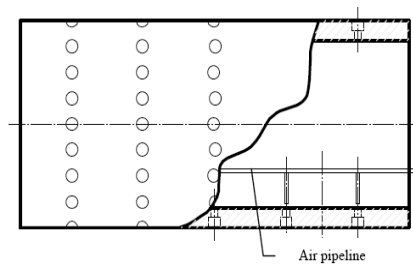


Fig. 10. Structure for discharging pills

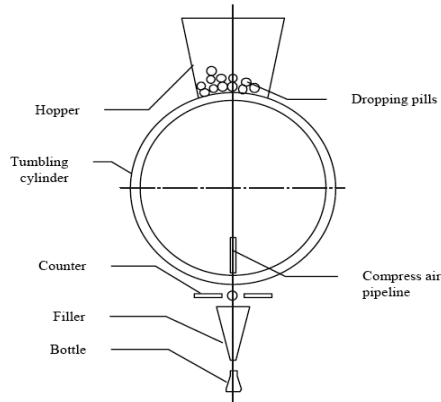


Fig. 11. A conceptual design of packaging machine for dropping pills

7 Conclusions

CAIs show all inventive principles and cases, which are the source designs of ABD. The databases of CAIs are a fruit of TRIZ researchers for many years, which have broadly applicability. The application of the database will improve the validity of ABD that has been extensively accepted by designers.

When TRIZ is applied to solve a contradiction in design the first and second stage analogy process are existed. The results of the first stage analogy process are source designs of the second stage analogy process. To find UXDs from the sources is the key step to generate successful ideas for innovation. Four types UXDs have been divided. The physical action for finding UXDs from the computer screen of CAIs is only looking.

A seven step process model is formed for design, in which UXDs are driving force for generating new ideas. Designers find UXDs from the TRIZ special solutions and react with experiences that designers have to construct memories suddenly. Then new ideas for domain solutions are formed.

The model put forward is only related to contradiction solving of TRIZ. It needs to extend the model to technological evolution, effects, and standard solutions of TRIZ in order to effective application of CAIs.

8 Acknowledgement

The research is supported in part by the Chinese Natural Science Foundation under Grant Numbers 50675059 and Chinese national 863 planning project under Grant Number 2006AA04Z109. No part of this paper represents the views and opinions of any of the sponsors mentioned above.

9 References

1. G.J. Smith and J.S. Gero, What Does an Artificial Design Agent Mean by Being Situated?, *Design Studies*, 26, 535-561 (2005).
2. J.S. Gero, Constructive Memory in Design Thinking, *In: Design Thinking Research Symposium: Design Representation*, Goldschmidt G and Porter W, eds. MIT, Cambridge, 29-35 (1999).
3. S. Kohn, S. Husig and A. Kolyla, Development of an Empirical Based Categorisation Scheme for CAI Software, *In: 1st IFIP TC-5 Working Conference on CAI*, ULM, Germany, Nov.14-15, 143-157 (2005).
4. G. Altshuller, The Innovation Algorithm, TRIZ, Systematic Innovation and Technical Creativity, Worcester, *Technical Innovation Center, INC* (1999).
5. R. Tan, Process of Two Stages Analogy-based Design Employing TRIZ, *International Journal of Product Development*, 4(1/2), 109-121 (2007).
6. J. Kulinski and J.S. Gero, Constructive Representation in Situated Analogy in Design, *In: CAAD Futures 2001*, Kluwer, Dordrecht, Vries B and Achten H, eds, 507-520 (2001).
7. G.M. Martin, How Combinations of TRIZ Tools are Used in Companies – Results of a Cluster Analysis, *R&D Management*, 35(3), 285-296 (2005).
8. J.S. Gero, Concept Formation in Design: Towards a Loosely Wired Brain Model, Candy L and Hori K. eds. *In: Strategic Knowledge and Concept Formation Workshop*, Loughborough University of Technology, 135-146 (1997).
9. M. Suwa and J.S. Gero and T. Purcell, Unexpected Discoveries and Inventions of Design Requirements, *Design Studies*, 21, 539-567 (2000).
10. S. Ricardo and J.S. Gero, Computational Models of Creative Situations, Gero J S and Brazier FMT, eds. *In: Agents in Design 2002 Key Centre of Design Computing and Cognition*, University of Sydney, 165-180 (2002).
11. J.S. Gero, Design Tools as Situated Agents that Adapt to Their Use, Dokonal W and Hirschberg U, eds. *In: eCAADe21, eCAADe*, Graz University of Technology, 177-180 (2003).
12. G. Pahl and W. Beitz, *Engineering Design – a Systematic approach*, 2nd Edition, London, Springer (1996).
13. N.P. Suh, *Axiomatic Design- Advance and Application*, New York, Oxford University Press (2001).
14. A. Chakrabarti and T.P. Bligh, An Approach to Functional Synthesis of Mechanical Design Concepts: Theory, Application, and Emerging Research Issues, *AIEDAM: Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 10(5), 313-332 (1996).
15. L. Qian and J.S. Gero, Function-behavior-structure Paths and Their Role in Analogy-based Design, *Artificial Intelligence for Engineering Design, Analysis Manufacturing*, 10(4), 289–312 (1996).
16. X.F. Zha and H. Du, Mechanical Systems and Assemblies Modeling using Knowledge Intensive, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AIEDAM)*, 15(2), 145-171(2001).
17. G. Cao and R. Tan, FBES Model for Product Conceptual Design, *International Journal of Product Development*, 4(1/2), 22-36 (2007).
18. F. Zhang and D. Xue, Distributed Database and Knowledge Base Modeling for Concurrent Design, *Comput-Aided Design*, 34(1), 27–40 (2002).
19. D. Xue and H. Yang, A Concurrent Engineering-oriented Design Database Representation Model, *Computer-Aided Design*, 36, 947–965 (2004).
20. R.B. Stone, and K.L. Wood, Development of a Functional Basis for Design, *Transactions of the ASME, Journal of Mechanical Design*, 122(4), 359–370 (2000).
21. A.K. Goel, Design, Analogy and Creativity, *IEEE Expert*, 12(3), 62-70 (1997).