N° d'ordre :

École Doctorale Mathématiques, Sciences de l'Information et de l'Ingénieur

UdS – INSA – ENGEES

THÈSE

présentée pour obtenir le grade de

Docteur de l'Université de Strasbourg
Discipline : MECANIQUE DES SOLIDES, GENIE MECANIQUE, PRODUCTIQUE
Spécialité : Génie Mécanique

par

Huichao SUN

Improving product performance by integration use tasks during the design phase: a behavioural design approach

(L'amélioration de la performance du produit par l'intégration des tâches d'utilisation dès la phase de conception: une approche de conception comportementale)

Soutenue publiquement le 06 mars 2012

Membres du jury

Directeur de thèse :	M. Mickaël GARDONI, Professeur, HDR, INSA de Strasbourg
Rapporteur externe :	M. Samuel GOMES, Professeur, HDR, Université de Technologie de Belfort-Montbéliard
Rapporteur externe :	M. Kondo Hloindo ADJALLAH, Professeur, HDR, Ecole Nationale d'Ingénieurs de Metz
Examinateur :	M. Patrick MARTIN, Professeur, HDR, Arts et Métiers ParisTech centre de Metz
Examinateur :	M. Dominique KNITTEL, Professeur, Université de Strasbourg
Examinateur :	M. Rémy HOUSSIN, Maître de Conférences, Université de Strasbourg
Invité :	M. Amadou COULIBALY, Maître de Conférences, HDR, INSA de Strasbourg

Laboratoire de Génie de la Conception (LGéCo) EA 3938

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Prof. Dr. Mickaël GARDONI, for his continuous engorgement, invaluable insights and careful inspection for every detail during my PhD study. I have greatly benefited from his supervision style that provides intense training on how to think carefully and how to present logically. Secondly, I would like to thank my assisting supervisor, Prof. Dr. Rémy HOUSSIN, for his open-minded mindset as a leader and his trust in my choices throughout this research work. Both Mickaël and Rémy have provided a creative work atmosphere in the research group and always encourage one to strive beyond one's limits. Thank you both for all your discussions and guidance!

I would also like to express my appreciation to the wonderful faculty members and stuff of the Design engineering laboratory (LGéCO) at the National Institute of Applied Science (INSA) of Strasbourg, for their support, understanding, and kindness throughout the course of my doctoral studies.

I would especially like to thank Prof. Amadou COULIBALY, Dr. Jing XU, Dr. Qiang ZHANG, Dr. Wei YAN, and Dr. Huan HE for their commitment to our common research interest of performance in product development. Our common goal and different backgrounds became the basis for inspiring creative thinking. Thank you very much. I am looking forward to continuing our collaboration!

Last, but not least, I would like to send my love to my family and friends for their friendship and support in my "other" life which runs beside the "research life".

Abstract

Mechanical engineering design processes are often technology-centered and have difficulties to integrate user's behaviour in term of using the product adequately. This problem is encountered along the whole life cycle of a project, and is especially noticeable during the early design phase. Although, industry and academia agree that human aspects are important for the success of the product, there are few methods that support the designers concerning these factors in the synthesis part of the design works.

Mechanical engineering design is connected with human behaviours targeted at and eventually leading to the development of the product. These behaviours take place all over the product lifecycle. In order to improve product performance, our research carefully thinks out a piece of research linking the user cantered and functional engineering design approached into an integrated package. It aims to a better integration of product and user behaviour during the early design phase. Designers have been obliged to set aside their dreams of a 100% machine due to the vital requirement of the user to perform some definite tasks with machines. While machine productivity and use conditions are the main reasons for automating production systems, human intervention on such systems remains a critical need and the tasks performed by the user remain poorly defined at the early design stage.

The focus of this research is the development and evaluation of a top-down technical and socio-technical framework for engineering design, which integrates various knowledge bases and the task model. The rationale behind such a framework is to develop a behavioural design approach not in a technology-centered approach, but with a socio-technical approach, in order to help designers to optimize the product performance through taking into account using conditions and requirements during the early design phase. We propose here a design approach that integrates user's and system's behavioural data as design specifications. We attempt to provide seamless integration means by merging engineering data and user-centered data within the engineer's toolkit. Otherwise, classical user-centered approach may seem difficult to handle by the whole design team: in this respect, this work provides a formal integration model in the framework in mechanical engineering design.

This paper covers the multi-trade engineering design, and deals with the development of a behavioural design approach to help designers to optimize the product performance in the early design phase through taking into account utilization conditions and requirements. Finally, a software application is in development to support and allow a systematic utilization of the "behavioural design approach" by integrating it into the daily work of the designer.

Key words: Design method; Behavioural design approach; Use conditions; Task; User behaviour; Structure behaviour

Résumé

Les processus de conception d'ingénierie mécanique sont souvent centrés sur la technologie et ont des difficultés à intégrer de façon adéquate les comportements des utilisateurs lors des utilisations du futur produit. Ce problème existe tout au long du cycle de vie du projet de développement du produit et est particulièrement visible lors des phases préliminaires de conception. Bien que les industries et les universités conviennent que les aspects humains sont importants pour le succès du produit, il existe peu de méthodes qui soutiennent les créateurs/concepteurs pour la prise en compte de ces facteurs lors des travaux de conception.

Afin d'améliorer la performance du produit, notre recherche vise à apporter ou complémenter la conception technique fonctionnelle par une approche plus intégrée. Elle vise en particulier une meilleure intégration du comportement du couple produit/utilisateur dès la phase de conception. En effet, les concepteurs ont été obligés de mettre de côté l'objectif d'une machine entièrement automatisée et doit continuer à faire appel à l'utilisateur pour effectuer certaines tâches. Même si pour améliorer la productivité de la machine l'automatisation des systèmes de production est une voix intéressante, l'intervention humaine sur ces systèmes reste un besoin critique, or elle reste mal définie au stade de la conception. Dans notre cas, le système mécanique pourrait être un système de production, une machine, un produit ou tout autre outil manipulé par un utilisateur. Les conditions d'utilisation sont directement influencées par les travaux de conception, qui constituent également le principal facteur d'amélioration des performances du système.

L'objectif de cette recherche est le développement d'une évaluation technique « top-down » et d'une conception d'ingénierie socio-technique pour intégrer les diverses bases de connaissances et en particulier le modèle de tâche. L'objectif est donc de développer une approche de conception comportementale non pas uniquement centrée sur la technologie mais aussi sur une approche socio-technique, afin d'aider les concepteurs à optimiser la performance du produit globalement dès les premières phases de conception. Ainsi, nous proposons une approche qui intègre les données comportementales système technique, utilisateur et utilisateur/technique.

Ce travail porte sur la conception d'ingénierie multi-métiers et traite de l'élaboration d'une approche de conception comportementale pour aider les concepteurs à optimiser la performance du produit globalement dès la phase de conception grâce à la prise en compte des conditions d'utilisation et de la présence de l'utilisateur. Pour expérimenter ces travaux, un logiciel est en développement pour soutenir et permettre une utilisation systématique de cette «approche de conception comportementale» en l'intégrant dans le travail au quotidien du concepteur.

Mot clés : Méthode de conception, une approche de conception comportementale, Conditions d'utilisation, Tâche, Comportement des utilisateurs, Comportement de structure.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

Along with 1) increased product complexity, 2) the urgency of market equipment, 3) standards requirements, and 4) the spread of the networked product development mode, companies must guarantee a more important estimation of the user activities required to use the system. The designers have to set aside the 100% automated machine because of the vital requirements of the user to perform some definite tasks with machines. In order to improve mechanical system performance, our research targets a better integration of system and user behaviour during the early design phase. In our case, the mechanical system could be a production system, a machine, a product or any tool handled by a user. Use conditions (each model of the mechanical system is designed for and intended to be used in a specific set of conditions) are directly influenced by design results, which also constitute the major factor for improving system performance.

Mechanical engineering design processes are often technology-centered and have difficulties to integrate user's behaviour in term of using the product adequately. This problem is encountered along the whole life cycle of a project, and is especially noticeable during the early design phase. Although, industry and academia agree that human aspects are important for the success of the product, there are few methods that support the designers concerning these factors in the synthesis part of the design works.

The focus of this research is the development and evaluation of a top-down technical and socio-technical framework for engineering design, which integrates various knowledge bases and the task model. The rationale behind such a framework is to develop a behavioural design approach not in a technology-centered approach, but with a socio-technical approach, in order to help designers to optimize the product performance through taking into account using conditions and requirements during the early design phase. We propose here a design approach that integrates user's and system's behavioural data as design specifications. We attempt to provide seamless integration means by merging engineering data and user-centered data within the engineer's toolkit. Otherwise, classical user-centered approach may seem difficult to handle by the whole design team: in this respect, this work provides a formal integration model in the framework in mechanical engineering design.

This thesis is accomplished at the Laboratoire de Génie de la Conception (LGéCo) and University of Strasbourg and financed by the China Scholarship Council (CSS).

1.1 Objective and position of our work

The objective of this research is to develop a design methodology for mechanical engineering design during the early design phase, which takes care of the structure behaviour and user behaviour, and provides a modelling formalism which can realize the behaviour comparison. As a result, the design quality can be assured, the use conditions can be safety.

We propose an approach to help the designer optimize product performance from the early design phase, taking into account utilization conditions and requirements. This approach is based on a Task Model and the fact that the behavioural system (system and end-user) must be studied and defined from the early design phase. We focus on a production system design, and so, to complete the mechanical system design method, we propose a global view of the behavioural design approach in the early design stage.

Our intention in that regard is to help the designer answer the following questions early in the design process:

- What does the system do to fulfil its automated functions?
- What does the user do to fulfil manual functions?
- How does the system fulfil its function according to both the designer's point of view and the user's point of view?
- What is the interaction between the user and the system when fulfilling the functions?
- What are the parameters needed to:
- Define the place, the duration, and the nature of the task carried out by the user and the system?
- Have maximum global system performance by minimizing the influence of user ability on global system performance (Buzacott 2002)?

To answer these questions, we propose the behavioural design approach. Where, as noted by Darses (Darses and Wolff 2006), the designer does not have many direct inputs concerning the real needs of end-users, indirect inputs, such as human factors, information provided by ergonomics guidelines, and task analysis, can bridge this gap. However, this does not prevent the designer from referring to his own experience and knowledge of the user's behaviour. These representations of use, whatever their nature, play a decisive role in the choice of a solution (system structure).

This behavioural design approach relates both to tasks done by the system and the users. For example, to load a machine (function identified by FA) what must the machine do, as a task, to be loaded? What will the user do, as a task, and how will he achieve this task to effectively fulfill this function? What are the consequences of their interaction? For example, must the user stop the machine in order to load it, or is it possible to load it otherwise? Is this interaction dangerous? Does it require safety measurements? If so, what could these safety measurements be? What are their effects on machine performance? Specifically, the actual integration of aspects regarding safety, accessibility, usability and user ergonomics is carried out too late in the process to meet the requirements and propose some safety procedures to be applied by the user; moreover, these procedures are often hard to apply, and sometimes require that the machine be stopped (Hasan, Bernard et al. 2003; Houssin, Bernard et al. 2006). This late integration leads to reduced system performance. Some investigations have been performed to consider other lifecycle constraints at the early stage of the design process. We intend to optimize the interaction between the system and its potential users from a socio-technical point of view, starting from the early design phase (Carayon 2006). To that end, a global view of the behavioural design approach is proposed as a feasible solution to improve system performance starting from the early design phase.

The ultimate goal of this work is to realize a computer aided design tool for mechanical engineering design that can provide higher design efficiency in terms of design lead time and lead designers to better design quality, as shown in figure 1.1.

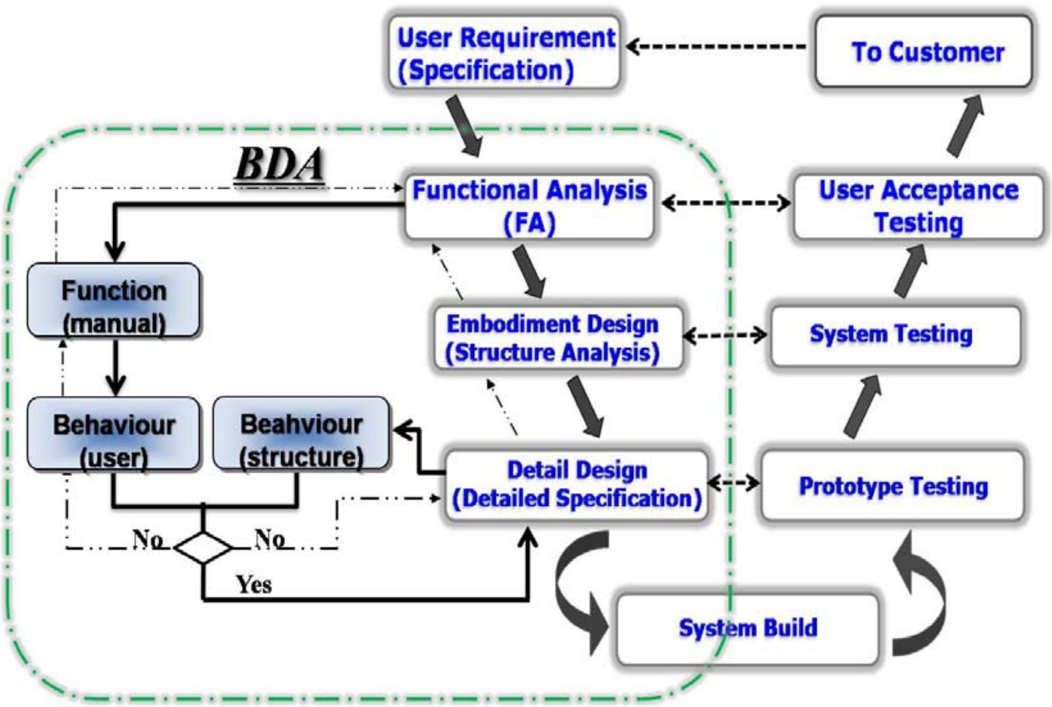


Fig 1.1 BDAS in the engineering design process

In the following paragraphs we will justify our objectives.

1.2 Background

Mechanical engineering design is a process of divergent thinking and routine design. It contains multi-trade approaches which allow taking into account different kinds of criteria. It is the solution of a function used to meet different kinds of techno-economic indicators, and is intended to establish the optimal plan from different possible proposals (Feng and Song 2003). It is widely known that the bulk of the production cost is incurred at the end of the design process (Hsu, Chuang et al. 2000), and as a result it is therefore crucial to avoid errors in the design stage. As shown in figure 1.2, the impact of design decisions is initially very high and declines steeply as the design matures (Wang, Shen et al. 2002). A great opportunity exists at the early design stage to make a few reading decisions.

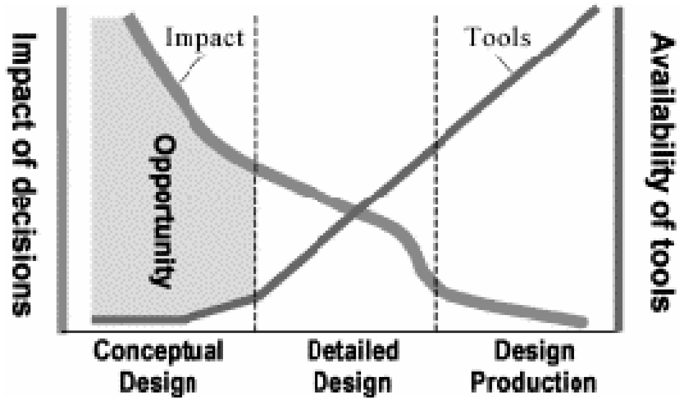


Fig 1.2 Impact of decesions at the early design stage after (Wang, Shen et al. 2002)

Several relevant concepts are rising from the early design stage. Designers have to perform a first choice among these concepts. In order to limit the duration of the design process and to decrease the risks, they tend to focus very early towards a solution they can handle. So, the alternatives might be eliminated, because they are unknown, unused or non-evaluated. Then, starting from a chosen concept, embodiment design aims to define rough arrangements and structural dimensions of the intended product, in accordance with technical, economic and aesthetic considerations (Pahl and Beitz 1996). Designers have to identify key parameters relevant for this definition. Even if there are few parameters and standard elements for a simple mechanism, there is however a great number of possible combinations. Then, designers make their choices (Hicks and Culley 2002), based on their own experiences and knowledge. So, design iterations cannot be avoided between the early design phase, before reaching design solutions (Ashby, Décarroux et al. 2000; Dieter, Schmidt et al. 2009). Moreover, the embodiment solutions are depending on these initial decisions, which greatly limit the field of design investigations. Another difficulty in an integrated engineering context is the disagreement, as soon as possible, among the different points of view of the designer, user and technical skills (technical data, marketing data, company identity, environmental data, ergonomics, cultural and symbolic aspects).

In industry, current design approaches, such as Concurrent Engineering (Moustapha 2006), are often focused on the technical satisfaction required by customers. A range of methods for studying users and involving them in the design process have been developed based on User-Centered Design (Redstrom 2006; Redstrom 2008). However, user-centered design is rather difficult and expensive to apply because it takes time to gather data from and about users, especially if the idea is to understand the environment in which they will be using the products. The process requires resources, both financial and human. Another approach is Axiomatic design, which is proposed as a systems design methodology using matrix methods to systematically analyze the transformation of customer needs (Suh 2001) into functional requirements, design parameters and process variables. Finally, user needs (facility, safety, etc.) are often taken into account at the last phase of the design process in a bid to respect legislation and standards.

Many works in the literature confirm the fact that human integration is necessary and crucial in the design process (Obradovich and Woods 1996; Potter, Roth et al. 2000; Hendrick. 2003; EIMaraghy and Urbanic 2004). Marsot (Marsot 2005) holds that imposing the least possible demand on the user (do the job harmlessly, effortlessly, comfortably) now complements the initial aim involving performance-related efficiency (do the job better and quicker than by hand). To fit users' needs and preferences (i.e., to have a high degree of user fitness), rather than focusing just on the functional factor, designers must also consider psychological, cultural, social and ideological factors (Siu 2003). Wu et al. (Wu, Lo et al. 2006) call for "establishing design processes in which the end-users themselves can influence the design so that it is compatible with their goals and beliefs, etc". Sundin et al. (Sundin, Christmansson et al. 2004) claim that it is not enough to improve workplaces and production systems themselves; it is also necessary to involve "the earlier step that affects the production system, i.e., the system design". Sagot et al. (Sagot, Gouin et al. 2003) hold that only a multidisciplinary approach combining social sciences and engineering sciences can respond to the challenge of the human factor being given greater consideration in the design of products. Other works propose to improve existing system performance using information systems. These information systems take into account the user behaviour (Hayman and Elliman 2000), modelling of users' capabilities and the organization (Juran and Schruben 2004), and using dynamic task allocation (Buzacott

2002). Moreover, users' expertise often allows considerable autonomy over the design and execution of their work (Lind and Sulek 2000; Pilemalm, Lindell et al. 2007). However, in all these works, the technical system is already designed and is theoretically in use in the designer's imagination. The utilization aspects are only taken into account in the last step.

Through a case study, Cullen (Cullen 2007) shows that to facilitate the identification and assessment of user requirements, human factors must be integrated into the design from the start of the design lifecycle. However, sometimes it is difficult to integrate user factor into design because of his unknown. In Product Life-cycle Management (PLM), product design is a complex process during which stakeholders in various trades try to take into account all the phases of the product lifecycle (Gardoni 2005). Also, Noël (Frederic Noel 2006) focus on the need to have a multi-view model for sharing the product behaviour throughout the entire product lifecycle, including the utilization phase, where the system is used by the end-user. He distinguishes two behaviour paradigms: derived behaviour, issuing from an analysis activity, and expected behaviour, as a functional requirement for product design. So we propose, in this work, to concern the expected behaviour and derived behaviour during the early design phase.

Over the last two decades, several computer-based design methodologies have been proposed to increase the effectiveness and achieve better control over the design process. Computers have been used extensively in areas such as simulation, analysis and optimization, but relatively few applications exist at the early design stage. This is because knowledge of the requirements and constraints during this early phase of a product's lifecycle is usually imprecise and incomplete. This lacuna makes it difficult to utilize computer-based systems or prototypes (Hasan, Bernard et al. 2003). There are many software applications and tools which could be used for design, such as CAD, CAM, CAE, CAPP, but which still rely primarily on geometric data (Houssin, Bernard et al. 2006). However, these commercial CAD systems are normally applicable on the product structure and its movement from a technical point of view, as the evaluation of the reliability or the kinematic. This evaluation is only half the track of a whole design process, and some critical issues are still present, such as a shortage of data on the consequences of this movement for the use of the product, like hazards and/or dangerous zones generated by the movement. Houssin and Gardoni (Houssin and Gardoni 2009) propose a Computer Aided Safety Integration in Design process (CASID) approach to help designers integrate users' safety in the design process. This lack of data is at the root of the loss of performance noted at the user site. In reality, standards require that the system be stopped if the user has to intervene in a dangerous zone or could be exposed to a hazard in performing his task. CAD technology is not yet at the level of development allowing it to take this type of data into account. To minimise this lack, a software application is in development to support and allow a systematic utilization of the "behavioural design" approach by integrating it into the daily work of the designer.

Process modelling describes design work, which is usually supported by some known methods. Design process modelling is implemented mainly through the mapping of function and structure. Most processes modelled are extended from two basic frameworks: Function-Structure (FS) (Pahl and Beitz 1996; Suh 2001) and Function-Behaviour-Structure (FBS) (Gero 1990; Umeda and Tomiyama 1997; Labrousse 2004). The FBS framework was first advanced by Gero (1990), who pointed out that the structure expresses the internal and external states of a physical element. This type of framework considers that function and structure must be linked through behaviour 1) to depict the action that is executed to complete a function, and 2) to indicate "how structure fulfils the function". Nowadays, early system design is regarded as an ordinal process of mapping of function, behaviour and structure. Stalker (Stalker 2002) improved the FBS framework to take into consideration the elaboration of product models in terms of Function, Behaviour and Structure. In this framework, the author illustrates that when a product exists physically; its situation and use conditions are easily identifiable and may be affected by its function. This framework presents steps (Function, Behaviour and Structure), with the designer going from one step to another in completing processes (formulation, synthesis, analysis, comparison, construction, etc.). In this framework, the author states that in the design phase, there are two behaviours (expected and predicated), which must be considered. Stalker extended this framework, proposing the undated behaviour, which results from feedback using a product prototype (the physical use situation). It is often used in knowledge management, and has been enriched by Labrousse (Labrousse 2004), who proposed the integration of Process, Product and Resource (FBS-PPR). Unfortunately, this framework only covers product behaviour, while ignoring end-user behaviour; moreover, it does not go so far to propose how these behaviours could be characterised and what their influence on product performance might be. Hasan et al. (Hasan, Bernard et al. 2003) and Houssin et al. (Houssin, Bernard et al. 2006) proposed a "Working situation" model, in which they modelled the task of the end-user and of the product in identifying the risks that could be generated when the end-user performs his task in an identified dangerous zone. However, this model does not go as far as analysing the nature of different tasks or their types. As well, it does not identify their influence on product performance. The authors proposed this model simply to aid the designer in integrating end-user safety in the design process. Our work, detailed in the following chapters, adapts from this Function, Behaviour and Structure approach.

It must be stated that all these important contributions do not yet offer any formalized methods or tools to help a designer study and evaluate behaviour and to carry out a Behavioural Analysis complementary to a Functional Analysis during the design stage.

1.3 Problem statement

Mechanical engineering design is connected with human behaviours targeted at and eventually leading to the development of the product. These behaviours take place all over the product lifecycle. In order to improve product performance, our research carefully thinks out a piece of research linking the user cantered and functional engineering design approached into an integrated package. It aims to a better integration of product and user behaviour during the early design phase. Designers have been obliged to set aside their dreams of a 100% machine due to the vital requirement of the user to perform some definite tasks with machines. While machine productivity and use conditions are the main reasons for automating production systems, human intervention on such systems remains a critical need and the tasks performed by the user remain poorly defined at the early design stage.

In traditional engineering design, designers normally take into consideration product functions and structures, while users' behaviours in terms of using the system are generally not fully considered during the early design phase. A product's behaviour is studied only from a technical point of view in order to verify its reliability and potential problems in the detailed design phase. However, this behaviour is neither characterised nor studied from a use point of view. Nowadays, although designers do increasingly have some understanding of user behaviour, they rarely pay much attention to the behaviour which derives from the structure (how the structure will move to fulfil the function), and behaviour which is fulfilled by the user (how the user will react to the machine).

Here, we quote two examples observed in real companies in order to reveal some factors of user behaviour in terms of using the system.

- Example one: Amusement equipment such as rotary amusement and rail amusement equipment, among others, are generally operated by workers. Moreover, passengers are in a position of being controlled; it is thus hard for them to choose and control the motion, and their fate is totally tied to the machine. Users here refer to two groups of persons: workers (operators) and passengers (clients). Regarding equipment operators, the main mechanical danger comes from incorrectly transferring and converting from kinetic to the potential energy (Bagge and Pendrill 2002). Various safety protection devices can be adopted to distance the human from the machine in order to keep the operator far away from danger. Due to the danger posed by the design, it is crucial that various safety regulations be adopted and set up, and management should stipulate human behaviour. Personal protection equipment should also be provided to avoid or reduce injury. However, such personal protections could not only restrict user movement, visibility and accessibility, but also require time to be put on.

- Example two: At a Printer Manufacturer's, in order to respect the legislation, designers try to reduce the dangerous phenomenon. Usually the potentially dangerous phenomenon is bounded according to its nature in a zone. The concept of the dangerous zone defines any zone inside and/or around a system in which a person is exposed to a risk of hurt or health damage (Standardization 1994). In our research, the dangerous zone can be situated at three levels, as shown in Figure 1.3.

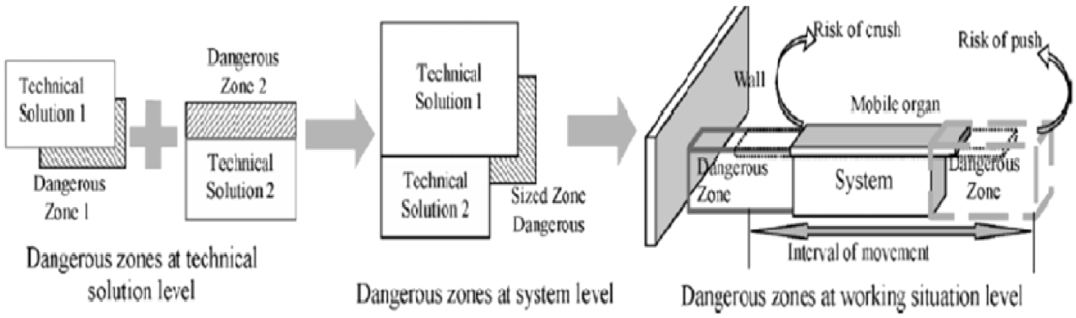


Fig 1.3 The dangerous zones sized at the various levels after (Houssin, Bernard et al. 2006)

- Technical solution level: in which could be engendered a dangerous phenomenon.
- System level: in which we represent the dimension 'assembly' of the technical solutions. In this level, the dangerous zones defined in the first level may be modified or even disappear.
- Working situation level: in which a dangerous zone does not exist before system installation but results from its integration on using site (e.g. the installation of a mobile organ near a wall or near another system would create a dangerous zone because of risk of crush between both systems or between the mobile organ and the wall).

Classically, the utilization of the product is studied in prototyping phase. If there is a problem, designer should add some new functions, such as some protection equipments which make product safe in use. If it is not possible, designer propose some coercive utilization procedures. These equipments and procedures could decrease product reliability and availability and by consequence its performance. In this stage, designer's choices are limited by the product itself and all modifications could be very expensive.

Additionally, early product development phases are considered to have the most influence on major changes of products in general. Thus the changing of products and product systems towards a sustainable development has its highest potential in early design phases. Furthermore, product development is becoming increasingly complex within industry. Taking into consideration the impact of utilization conditions of a new product is one more task to be added and integrated into the long list of things already under consideration.

Due to the complexity of the situation, there is a real need for efficient and easy-to-understand design methods applicable to product development and design. Adapting products to achieve a sustainable society, together with customer preferences and the complex situation facing designers constitutes the basis of this thesis. Moreover, this work develops a methodical approach for assisting designers in their endeavors to improve the product performance during the early design phase through taking into account using conditions and requirements, while taking into account other functional requirements at the same time.

1.4 Structure of the thesis

In the above sections, we explain the general introduction about our research, which includes the background, problem statement, as well as the research questions and aims.

The rest of the thesis is organized as follows:

Chapter 2 reviews the related work in order to shed light on the characterization and representation of design, engineering design and study of design theories and approaches. Three streams of related works are reviewed: a. the definitions of design; b. state of the art of the characterization and representation of the engineering design and engineering design process; and c. design theories and approaches from different kind of perceptive and criteria.

Chapter 3 proposes a global view of the behavioural design approach integrated with the task model and knowledge bases as a feasible solution to improve system performance starting from the early design phase. Two domains are explored, namely task domain and knowledge domain. A generic task model and knowledge based behavioural design model are developed respectively. A UML based representation scheme is also developed according to the characterization and representation of the behavioural design approach.

Chapter 4 develops a Behavioural Design Approach Software (BDAS). Two major parts are included in this chapter. One part concerns framework of the distributed Behavioural Design Approach (BDA) system for mechanical engineering design. The other part is the implementation of the system in C++ program language. Finally, a software application is in development to support and allow a systematic utilization of the "behavioural design" approach by integrating it into the daily work of the designer. The whole structure of the thesis is illustrated in figure 1.4.

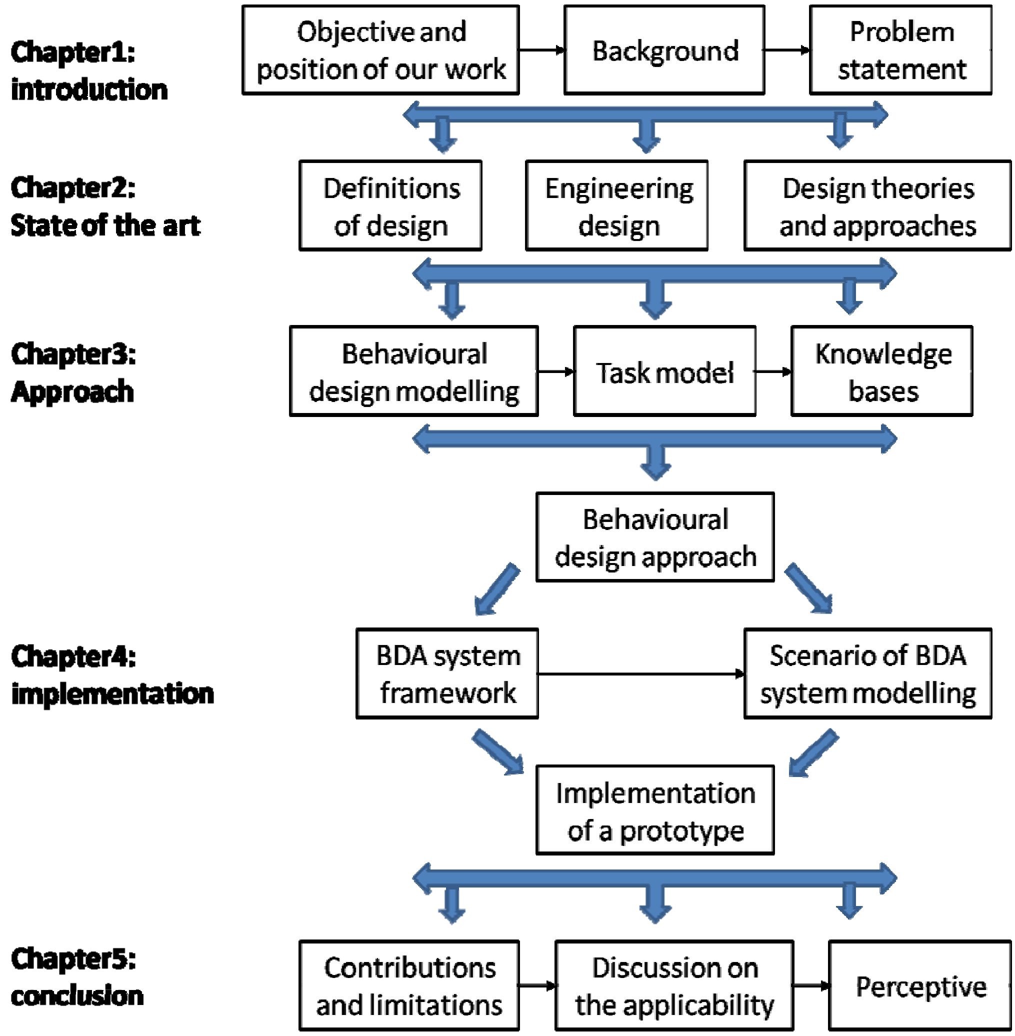


Fig 1.4 Structure of the thesis

Chapter 2 The state of the art

The previous chapter introduced the core research question of the thesis and the general motivation behind it. Here in the second chapter, we study the state of the art from multi-disciplines by adopting a divergent-convergent review strategy.

The product design process is the primary phase and also the creative part during the product life cycle. The implementation effect of the follow-up manufacture process is directly influenced by design results. The design results are also the major factor of controlling the product cost and use. Product design theories and methods have attracted widespread attention by academic and industrial research. And the design theories and methods have become important research areas and application fields. The reasons are as follows: 1) the increase of product complexity; 2) the urgency of market equipment; 3) the spread of the networked product development mode.

Design is one of the most crucial sectors of the economy. In recent years, more and more countries are realizing that design brings wealth to their societies. Design has been considered as both a technical and a socio-technical activity.

Here, we concentrate our study on the state of the art of design in the aspects of design theory, design process and knowledge.

2.1 Definitions of design

Design is a complex, multifaceted and broad concept with no universally agreed definition. There is agreement that design as a noun refers to a plan for the construction of an object while "to design" (verb) refers to making the plan. Here, we constrain our research into the engineering design field. So the main task of design is the application of their scientific and engineering knowledge to find and optimize solutions within the diverse requirements (Pahl and Beitz 1996).

Different aims of design bring on different design activities and definitions. Many researches and institutions define design from several standpoints. The UK Department of Trade and Industry (Department of Trade and Industry 2005) said the following about design: "*Design is a structured creative process. Design is readily associated with industrial product design for manufactured products -specifically the 'look' of a product. However, the application of design is much broader, for example designing for function; for aesthetic appeal; for ease of manufacture; for sustainability; and designing for reliability or quality and business processes themselves*." Design is a creative activity whose aim is to establish the multi-faceted qualities of objects, processes, services and their systems in the whole life cycle.

Therefore, design is the central factor of innovative humanization of technologies and the crucial factor of cultural and economic exchange (ICSID 2008). In nature, in design revolution today design is a tool for innovation and development.

Design is a process, an activity, and not only the results of that activity. Simon (Simon 1969) proposed that "*everyone designs who devise courses of action aimed at changing existing situations into preferred ones*". Design is an inseparable part of the overall technological system and provides the primary source database for all other activities in the system (Yoshikawa 1989). Design activity can be characterized as a goal-oriented, constrained, decision-making, exploration and learning activity which operates within a context which depends on the designer's perception of the context (Gero 1990). Hinrichs (Hinrichs 1992) defines design as "*the task of generating descriptions of artifacts or processes in some domain*". The definition provided in Engers et al (van Engers, Gerrits et al. 2001): "*design is the creative process of coming up with a well-structured model that optimizes technological constraints, given a specification*".

The government of New Zealand (NZIER 2003) defines design as follows: "*Design is an integrated process. It is a methodology (or a way of thinking) which guides the synthesis of creativity, technology, scientific and commercial disciplines to produce unique (and superior) products, services, and communications*". Some governments prefer talking about the potential of good design. The Danish government's 2007 white paper (Danish 2007) defines the good design as the following:"*Good design is an increasingly important means for businesses to hold their own in international competition. Design has the power to make products and services more attractive to customers and users, so they are able to sell at a higher price by being differentiated from the competition by virtue of new properties, values and characteristics".*

As highlighted by several of the definitions above, design is an activity that follows a certain methodology and a number of steps — such as research, conceptualizing, modelling, testing and re-design — and not only the results of that activity.

As Dieter et al. (Dieter, Schmidt et al. 2009) said in his book, the way to summarize the challenges presented by the design is to think of the four C's of design:

- Creativity: requires creation of something that not existed before or not existed in the designer's mind before;
- Complexity: requires decision on many variables and parameters;
- Choice: requires making choices among many possible solutions at all levels, from basic concepts to smallest detail of shape;
- Compromise: requires balancing multiple and sometimes conflicting requirements.

Design allows a broad range of considerations to be taken into account. Design as a purposeful and goal-oriented activity takes on various forms in practice. It is a holistic approach which allows a range of considerations beyond aesthetics to be taken into account, including functionality, ergonomics, usability, accessibility, product safety, sustainability, cost and intangibles such as brand and culture (European Communities 2009). The aim of design could be competitiveness and differentiation on international markets, or it could be sustainability and quality of life.

In short, design as an activity can and often does take place in any organisation (Community Design Regulation 2002). Design as a driver of user-centred innovation is a structured innovative process

2.2 Engineering design

Since design is detached from manufacturing because of the labour division and the specialization, two streams of design have been developed separately: design as art and design as engineering (Von Stamm 2003). The engineering design can be applied to several endings. One is the design of products, whether they are consumer goods and appliances or highly complex products. Another is a complex engineered system such as the production systems. The emphasis in this thesis is on complex product design because it is the area in which many engineers will apply design skills.

Since design stands at the core of both the craft and engineering traditions, its meaning and usage in technique is not always settled. Where craft design draws on aesthetics primarily, engineering design has both creative as well as rational dimensions (Cross and Knovel 2000).

In engineering design, the end goal is the creation of an artifact, product, system, or process performs a function or functions to fulfill customer needs. Conceptualizing, defining, or understanding an artifact, product, or system, in terms of function, is a fundamental aspect of engineering design (Hubka and Eder 1982; Ulrich and Eppinger 1995; Pahl and Beitz 1996; Ullman 1997; Otto and Wood 2001).

Dieter (Dieter, Schmidt et al. 2009) has written further that engineers work "*at the margin of solvable problems*," proceeding from the known to the unknown. They work under conditions of change, uncertainty, and resource constraints. Dieter explains that unlike scientists who proceed within the framework of scientific laws, engineers employ heuristic laws to arrive at design solutions.

Pahl and Beitz (Pahl and Beitz 1996) write that the main task of engineers is to "*apply their scientific and engineering knowledge to the solution of technical problems, and then optimize those solutions within the requirements and constraints set by the material, technological, economic, legal, environmental and human-related* *considerations*".

Beyond the technical, engineering design can also be situated in the domain of the philosophy. It is creative, requiring grounding in mathematics and science, as well as in domain specific knowledge and experience (Lewis 2005). These authors identify types of design, goals and methods.

Types include:

- innovative: new tasks and problems needing original design, which must proceed through all phases (Gero and Kannengiesser 2007);
- adaptive: established solution principles held continuous but the realization is adapted to alterative requirements;
- variational: sizes and arrangements of parts are varied within the original design parameters.

Goals include:

- optimization of function;
- minimization of cost;
- aesthetic considerations;
- ergonomic considerations (Das and Sengupta 1996).

Solution methods include:

- traditional (e.g. literature review);
- intuitive, inclusive of preconscious or subconscious ideas or insight or flash, brainstorming, or using analogy;
- discursive, use of design catalogs or systematic combinations.

Among these various explanations of engineering design, there are two different exemplifications, which are the problem solving (Simon 1969) and the reflective practice (Schön 1987). Problem solving is a mental process and is part of the larger problem process that includes problem finding and problem shaping. Considered the most complex of all intellectual functions, problem solving has been defined as a higher-order cognitive process that requires the modulation and control of more routine or fundamental skills (Simon 1969). Problem solving paradigm is set up on the epistemology of positivism, while the reflective practice paradigm is built on the epistemology of constructivism (Pahl and Beitz 1996). Reflective Practice is the capacity to reflect on the action so as to engage in a process of continuous learning, which, according to the originator of the term, is one of the defining characteristics of professional practice (Schön 1987).

2.3 Engineering design process

The engineering design process is a formulation of a plan or a scheme to assist an engineer in creating a product. The engineering design process is defined as: "*the process of devising a system, component, or process to meet desired needs. It is a decision making process (often iterative) in which the basic sciences, mathematics, and engineering sciences are applied to convert resources optimally to meet a stated objective. Among the fundamental elements of the design process are the establishment of objectives and criteria, synthesis, analysis, construction, testing and evaluation*" (Ertas and Jones).

It means that the engineering design process is the set of activities by which designers develop and/or select the means to achieve a set of objectives. The engineering design process may bring about the creation of a new solution, the selection of an existing solution, or a combination of the two.

2. 3.1 The importance of the engineering design process

Here, we introduce two quotes to emphasize the importance of design in the product realization process:

- *"Studies have shown that 50 to 80 percent of the life cycle cost of products are influenced in engineering design"* (PREVIEW 1995).
- *"After all, 70% of a product's total cost is determined by its design, and that cost includes materials, facilities, tooling, labour, and other support costs"* (Munroe 1995).

The two quotes not only indicate the large impact that the engineering design process has on product cost but also some of the other considerations that go into the product realization process, such as tooling, facilities, and labour. These other considerations dictate that certain members of the engineering design team must be knowledgeable in these other areas.

2.3.2 Detailed description of the engineering design process

Morris Asimow (Asimow 1962) is among the first to give a detailed description of the complete engineering design process in what he called the morphology of design. Pahl and Beitz (Pahl and Beitz 1996) provide one of the better known engineering design process. One of the useful parts of this process is the fact that it not only shows the steps, it shows what the output of each step should be. The list of the engineering design process described below is established by researching the literatures (Ertas and Jones ; Hubka and Eder 1982; Pugh 1991; Pahl and Beitz 1996; Cross 2000; Dieter, Schmidt et al. 2009).

2.3.2.1 Phase I: Conceptual design

Conceptual design is just like it sounds—the generation of a concept. It is the process by which the design is initiated, carried to the point of creating a number of possible solutions, and narrowed down to a single best concept (Dieter, Schmidt et al. 2009). The objective of this phase is to determine the principal solution. Some of the terms used by Pahl and Beitz to describe it are:

1. *Identification of customer needs:* The objective of this activity is to completely understand the customer's need and to communicate them to the design team. (but in our paper the customer needs and user needs are not the same things)
2. *Problem definition:* The objective of this activity is to make a statement that describes what must to be accomplished to satisfy the customer needs. Quality function deployment (DFD) (Akao 1990) is a valuable tool for linking customer needs with design requirements.
3. *Gathering information:* The objective of this activity is to ensure that you benefit from the work of others (i.e. don't reinvent the wheel!).
4. *Concept generation:* Concept generation is involved with creating a broad set of concepts that potentially satisfy the problem statement.
5. *Concept evaluation:* Evaluation of design concepts, modifying and evolving into a single concept.

2.3.2.2 Phase II: Embodiment design (Preliminary design)

In this phase a structured development of the design concept takes place. Embodiment design consists of preliminary layouts and configurations, selecting the most desirable preliminary layouts, refining and evaluating against technical and economic criterion (Cross 2008). Embodiment design is concerned with three major tasks:

1. *Product architecture:* It is concerned with dividing the overall design system into subsystems.
2. *Configuration design*: It means to determine what feature will be present and how these features are to be arranged in space relative to each other.
3. *Parametric design:* It starts with information on the configuration of the part and aims to establish its exact dimensions and tolerance.

2.3.2.3 PhaseIII: Detail design

In this phase the design is brought to the stage of a complete engineering description of a producible product. The detail design includes specifying the materials, the sizes, the type of motor, the size of the hydraulic pump and cylinders, where the attachment and assembly holes should be drilled, the size of the holes, etc. It requires a lot of skills to specify this myriad of items correctly if the design to "go together" in a satisfactory manner (Hubka and Eder 1982). Many alternatives and options should be considered during this phase of the engineering design phase.

2.3.2.3 PhaseIV: Iterations

All of the steps or phases of the engineering design processes indicate feed-back arrows which indicate re-doing or iterating the steps. It is important to note the iterative nature of the engineering design process. Various stages may be visited multiple times during the evolution of your design.

This re-doing is necessary because we seldom know enough at any stage of the design process to produce a complete answer, let alone the best one. For instance, we must define the problem to begin, but the beginning is precisely when we know the least about the system that we are designing. We learn about its characteristics, performance and limitations as we design.

The basic engineering design processes described above are displayed in figure 2.1.

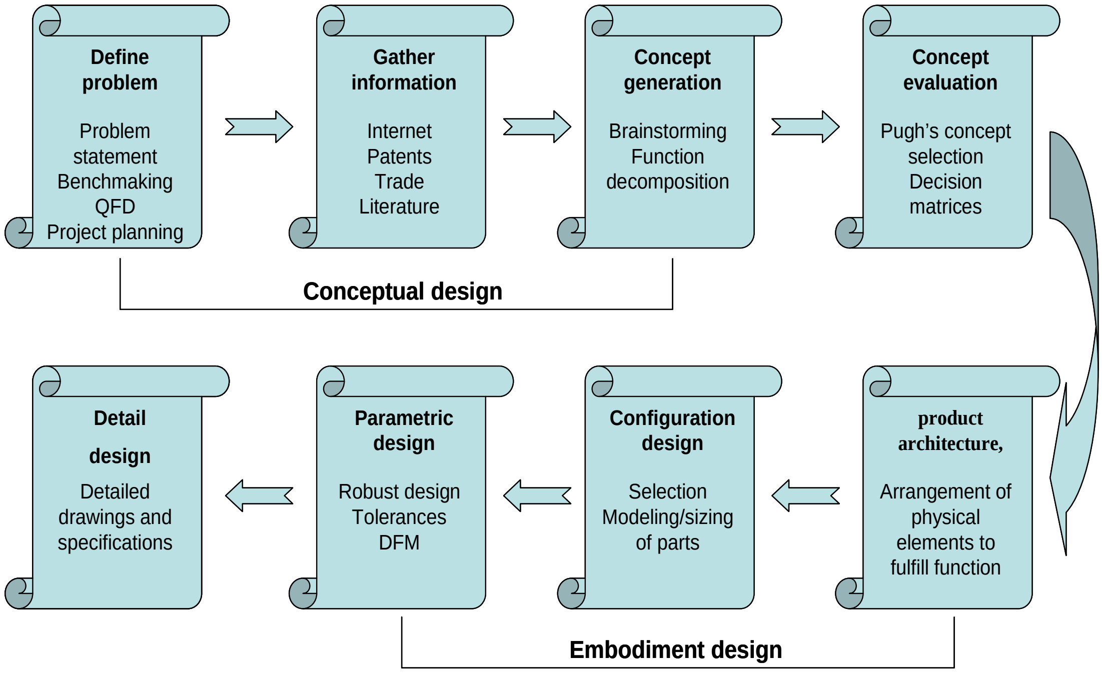


Fig 2.1 The basic engineering design process

This eight-step process is the representation of the basic engineering design process. The purpose of this graphic is to illustrate the logical sequence of activities that leads from the problem definition to the detail design. It constitutes the primary design process. However, the design does not normally proceed in a straight-line fashion. Much iteration will be necessary and can be expected for the final process.

Phases I, II and III take the design from the realm of possibility to the real word of practically. However, the design process is not finished with the delivery of a set of detailed engineering drawings and specifications. Many other technical and business decisions must be made that are really part of the design process.

In order to understand the realization of the engineering design process, we will review some known design theories and approaches in the following section.

2.4 Design theories and approaches

Over the last decades, most engineering design researchers have focused on developing prescriptive design methods such as the Systematic Approach (Pahl and Beitz 1996). Descriptive design theory was underestimate and sometimes ignored (Reich 1995). Few industrial design practices have a scientific or theory background. Engineering design is mainly performed on the basis of the designer’s experiences. On the other hand, no existing design theories have been really used in the industry because they mainly deal with the idealized design situations (the gap between theory and its uses). Nevertheless, it is generally agreed that the development of a theory of design will contribute to a better understanding of the design process and a better organization of design knowledge, and consequently allow better performing the design.

2.4.1 Design theories

A theory is an analytical tool for understanding, explaining, and making predictions about a given subject matter (Ayala 2008). Scientific theory is a deductive theory, in that, its content is based on some formal system of logic and that some of its elementary theorems are taken as axioms. In a deductive theory, any sentence which is a logical consequence of one or more of the axioms is also a sentence of that theory (Curry 1977).

A theory of design is scientific if it is developed using a scientific method. A scientific theory of design seeks to explain the design process and predict design results by repeatable or verifiable means. The method upon which the approach is developed is the experimental method consisted of three steps (Wiley 2000):

- The researcher observes facts (designs);
- He formulates hypothesis which can bring an explication to observed facts;
- He verifies by experimentation the pertinence of these hypotheses.

A theory of design is generic (or general) if its concepts and principles apply to various design areas. It is generally believed that there are no real differences between the design process that it is engineering products, architecture or civil engineering, chemical ,microelectronics and micro-mechanical products, etc. (Grabowski, Rude et al. 1998). Brown et al. (Brown, Waldron et al. 1998) indicate that although design problems in different domains require different domain knowledge such as knowledge

of equations, components, and the analysis techniques. There are underlying similarities in the form of that knowledge and in the way that is used.
The theory of design explains the phenomena of design by means of a set of concepts and operations between the concepts. It is generally believed that a theory of design is really useful if it is not only descriptive (to explain what is design), but also prescriptive (to show how to better perform design) and/or predictive (to forecast properties of designed objects) (Finger and Dixon 1989; Blessing, Chakrabarti et al. 1998). No theory can capture all of design perspectives; each theory provides one perspective, contributing to improve the understanding of design. Furthermore, developing a theory of design is itself a long and iterative process. The approach presented below is only an intermediate result of that process. To avoid to "reinvent the wheel", it must be based on the existing theoretical approaches (at least at its early stage of development).
Simon (Simon 1969) is the first to consider the design theory as a science of artificial. Early approaches view the design theory as a generalized problem-solving method (Asimow 1962; Simon 1969; Rittel and Webber 1984). Since the beginning of 1980's, the research on design theory has gained attention. The first general design theory was proposed by Yoshikawa (Yoshikawa 1989). Various known design theories developed today will be introduced in the following section.

2.4.1.1 General Design Theory (GDT)

General Design Theory (GDT) developed by Yoshikawa (Yoshikawa 1989), based on the philosophical and mathematical considerations, is the most general one. It uses set theory and topology to model design knowledge and design process. Although, it is limited to the study of idealized design process with perfect knowledge structure (topology). And it contributes to a better understanding of the process of designing and the structure of design knowledge from cognitive point of view.
In this theory, Yoshikawa picked up the notions of entity, entity concept, abstract concept, and attribute as basic items for an axiomatic theory of design, and proclaimed three axioms for them called the Axiom of Recognition, the Axiom of Correspondence, and the Axiom of Operation.
The axioms of GDT are basic conditions about the relationship and properties about entities, entity concepts, and abstract concepts.

1. Axiom 1 (Axiom of Recognition) Any entity can be recognized or described by the attributes.
2. Axiom 2 (Axiom of Correspondence) The entity set S' and the set of concepts of entity (ideal) S have the one-to-one correspondence.
3. Axiom 3 (Axiom of Operation) The set of abstract concepts is a topology of

the set of entity concepts.

2.4.1.2 Axiomatic Design Theory (ADT)

Axiomatic Design Theory (ADT) is a systems design methodology using matrix methods to systematically analyze the transformation of customer needs into functional requirements, design parameters, and process variables (Suh 1990). It aims at identifying generalizable principles which govern good design solutions. These principles are formalized in terms of axioms and theorems. The primary goal of axiomatic design is to establish a systematic foundation for design activity by two fundamental axioms and a set of implementation methods. The two axioms are:

- Axiom 1: The Independence Axiom: Maintain the independence of functional requirements.
- Axiom 2: The Information Axiom: Minimize the information content in design.

The design process is defined as the set of activities by which designers develop and/or select of a means (design parameters: DPs) to satisfy objectives (functional requirements: FRs), subject to constrains (Csikszentmihalyi). Axiomatic design breaks the design process into four domains, shown in Figure 2.2.

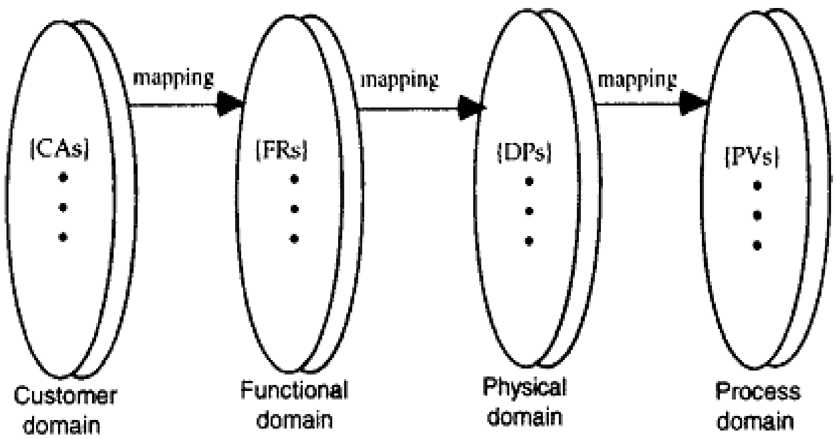


Fig 2.2 Axiomatic domains after (Suh 2001)

The customer domain can be thought of as the voice of the customer (VOC). The functional domain is initially populated by mapping the VOC into independent measurable functions. High-level functions are driven by the customer; lower-level functions are driven by design choices. Every function must be measurable. The physical domain is the domain of physics, chemistry, math and algorithms. The process domain is where the specifics of how the design parameters identified in the physical domain will be implemented.

Axiomatic design provides a framework for describing design objects, which is consistent for all types of design problems and at all levels of detail. Thus, different designers, as well as observers to the design process, can quickly understand the relationships between the desired functions of an object and the means by which the functions are achieved.

In summary, the main concepts of Axiomatic design are:

1. domains, which separate the functional and physical parts of the design;
2. hierarchies, which categorize the progress of a design in the functional and physical domains form a systemic level to more detailed levels;
3. zigzagging, which indicates that decisions made at one level of the hierarchy affect the problem statement at lower levels;
4. design axioms, which dictate that the independence of the functional requirements must be maintained and that the information content must be minimized as criteria for high-quality design.

In manufacturing, many disciplines and fields are involved, such as mechanical, electrical, hardware and software. However, all designs can be represented using the 4 design domains, enabling us to generalize the design process. The design objectives can be different from one problem to another, but all designers go through the same thought process. Table 2.1 shows how all these seemingly different design tasks can be described in terms of the 4 design domains.

Table 2.1 Characteristics of the 4 domains of the design world after (Suh 2001)

	Domains			
Character Vectors	Customer Domain **CA**	Functional Domain **FR**	Physical Domain **DP**	Process Domain **PV**
a. Manufacturing	Attributes which consumers desire	Functional requirements specified for the product	Physical variables which can satisfy the functional requirements	Process variables that can control design parameters (DP)
b. Materials	Desired performance	Required Properties	Micro-structure	Processes
c. Software	Attributes desired in the software	Output	Input Variables and Algorithms	Sub-routines
d. Organization	Customer satisfaction	Functions of the organization	Programs or Offices	People and other resources that can support the programs
(e) Systems	Attributes desired of the overall system	Functional requirements of the system	Machines or components, sub-components	Resources (human, financial, materials, etc.)

2.4.1.3 Universal Design Theory (UDT)

Universal Design Theory (UDT) is drafted as an on-going research project by Grabowski (Grabowski, Rude et al. 1998). It is based on the systematic design approach and views design process as a finite number of abstraction levels and a set of structured stages to follow. It serves as a scientific basis for rationalizing interdisciplinary product development. UDT takes all the common features of different scientific and engineering domains into account in order to find a system of statements of general validity with regard to the explanation and prediction of artifacts and the way of designing them.

At the current stage of development, it is more a prescriptive methodology rather than a descriptive design theory. It intends to model design process knowledge and focus on how design should be done as a procedure.

2.4.1.4 MathematicalDesign Theory (MDT)

Mathematical Design Theory (MDT) elaborated by Maimon and Braha (Braha and Maimon 1998) considers the real design process as an evolutionary process. The application of mathematics in design research may be perceived very broadly: from the use of statistics to analyze empirical data, to the development of formal or axiomatic theories of design knowledge, processes or objects.

This theory uses two different mathematical set-ups to study respectively the idealized and real designs. The idealized design process is modeled using set theory and topology notations which are similar to the general design theory developed by Yoshikawa (Yoshikawa 1989). The automation theory is used to represent the real design process. Production rules allow obtaining, after a finite number of transitions, a design solution.

2.4.1.5 Theory of inventive problem solving (TRIZ)

TRIZ is Russian acronym for The Theory of Inventive Problem Solving that originated from extensive studies of technical and patent information. Studies of patent collections by Altshuller (Altshuller and Rodman 1999),the founder of TRIZ, indicated that only one percent of solutions were truly pioneering inventions, the rest represented the use of previously known ideas and concepts but in a novel way. Thus, the conclusion was that an idea of a design solution to a new problem might be already known (as for example (Houssin and Coulibaly 2011)). However, where this idea could be found? TRIZ, based on the systematic view of the technological world, provides techniques and tools to help designers create a new design idea and avoid numerous trials and errors during a problem solving process.

Any problem solving process involves two components: the problem itself and the system in which the problem exists. A successful innovative experience shows that both problem analysis and system transformations are equally important to problem solving. Accordingly, TRIZ methodology includes the analytical tools for problem analysis and the knowledge base tools for system changing. The theoretical foundations of these tools are the patterns of evolution of technological systems.

Figure 2.3 illustrates the basic structure of TRIZ.

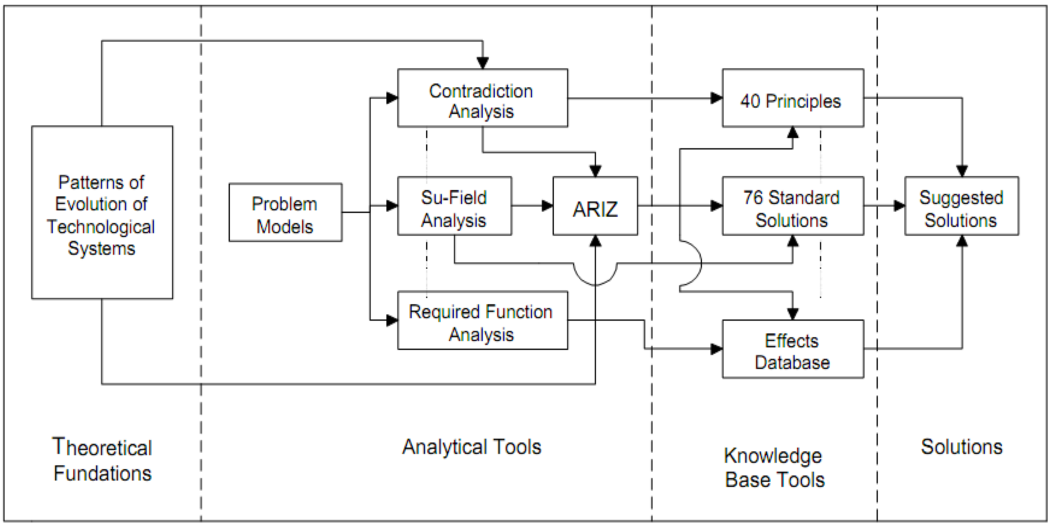


Fig 2.3 Structure of TRIZ theory after (Altshuller and Shulyak 2002)

Forty Inventive Principles are used to guide the TRIZ practitioner in developing useful "concepts of solution" for inventive situations. Seventy-six Standard Solutions were developed for solving standard problems based on the Patterns of Evolution of Technological Systems. The theory offers a framework and method to stimulate creative design solutions rather than to explain what is a creative design process. In this sense, it is more an approach to manage design innovation and to facilitate creative thinking than a theory of design properly speaking.

2.4.1.6 Concept-Knowledge theory (C-K design theory)

Concept-Knowledge theory(C-K design theory) is a design theory and a theory of reasoning in design. The C-K Design Theory was developed after a large number of empirical studies. It was originally drafted by Hatchuel (Hatchuel 1996), then consolidated by Hatchuel and Weil (Hatchuel and Weil 2003).

- It defines design reasoning as the logic of expansion processes, i.e. the logic that organizes the generation of unknown objects. The theory builds on several traditions of design theory, including systematic design, axiomatic design, creativity theories, general design theories, and artificial intelligence-based design models. Claims made for C-K design theory include that it is the first design theory that: Offers a comprehensive formalization of design that is independent of any design domain or object.
- Explains invention, creation, and discovery within the same framework and as design processes.

The name of the theory is based on its central premises: the distinction between two spaces:

- A space of concepts C, is a space containing concepts that are propositions, or groups of propositions that have no logical status (i.e. are undesirable propositions) in K;
- A space of knowledge K, is a space of propositions that have a logical status for a designer.

The process of design is defined as a double expansion of the C and K spaces through the application of four types of operators: C→C, C→K, K→C, K→K, as shown in figure 2.4.

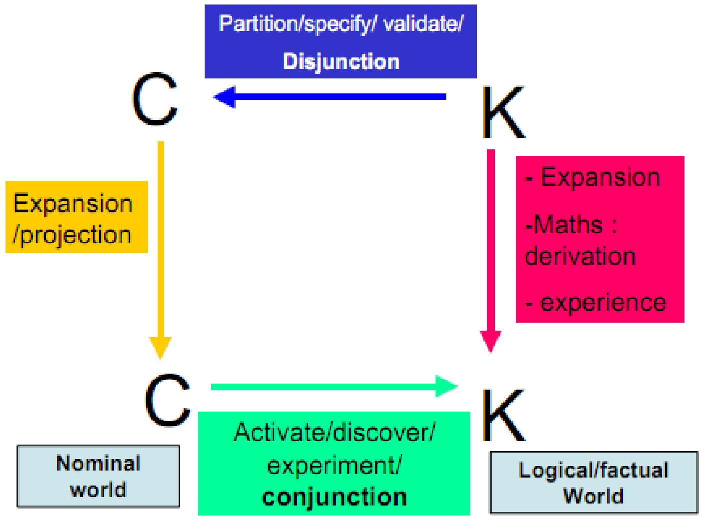


Fig 2.4 The design square modeled by C-K theory after (Hatchuel and Weil 2003)

A space of concepts is necessarily tree structured as the only operations allowed are partitions and inclusions and the tree has an initial set of disjunctions. In addition, we need to distinguish between two types of partitions: restrictive and expansive partitions.

- If the property added to a concept is already known in K as a property of one of the entities concerned, we have a restricting partition;
- If the property added is not known in K as a property of one of the entities involved in the concept definition, we have an expansive partition.
- In C-K theory, creativity is the result of two operations:
- Using addition of new and existing concepts to expand knowledge;
- Using knowledge to generate expansive partitions of concepts.

Besides the existing design theories, there is a considerable amount of approaches that can contribute to the development of a theory of design.

We will continue to introduce some known design approaches in the following section.

2.4.2 Design approaches

Design has been an important research subject for a long time, and therefore, many well-established engineering design approaches have been developed.

2.4.2.1 Systematic approach

The Systematic Approach which proposes to proceed in a structured way in design is mostly developed in Germany after the Second World War and is materialized by Hubka & Eder (Hubka and Eder 1982) and Pahl & Beltz (Pahl and Beitz 1996). The most representative result is also known as VDI design directives (Handbook 1987). This approach describes the engineering design process as a sequence of activities leading to intermediate results (performance specification, functional structure, principle solution, modular structure, preliminary layout, definitive layout and documentation). From a systematic approach, Pahl and Beitz proposed the famous four phases of the engineering design process, which is one of the most influential approaches as illustrated in the figure 2.5.

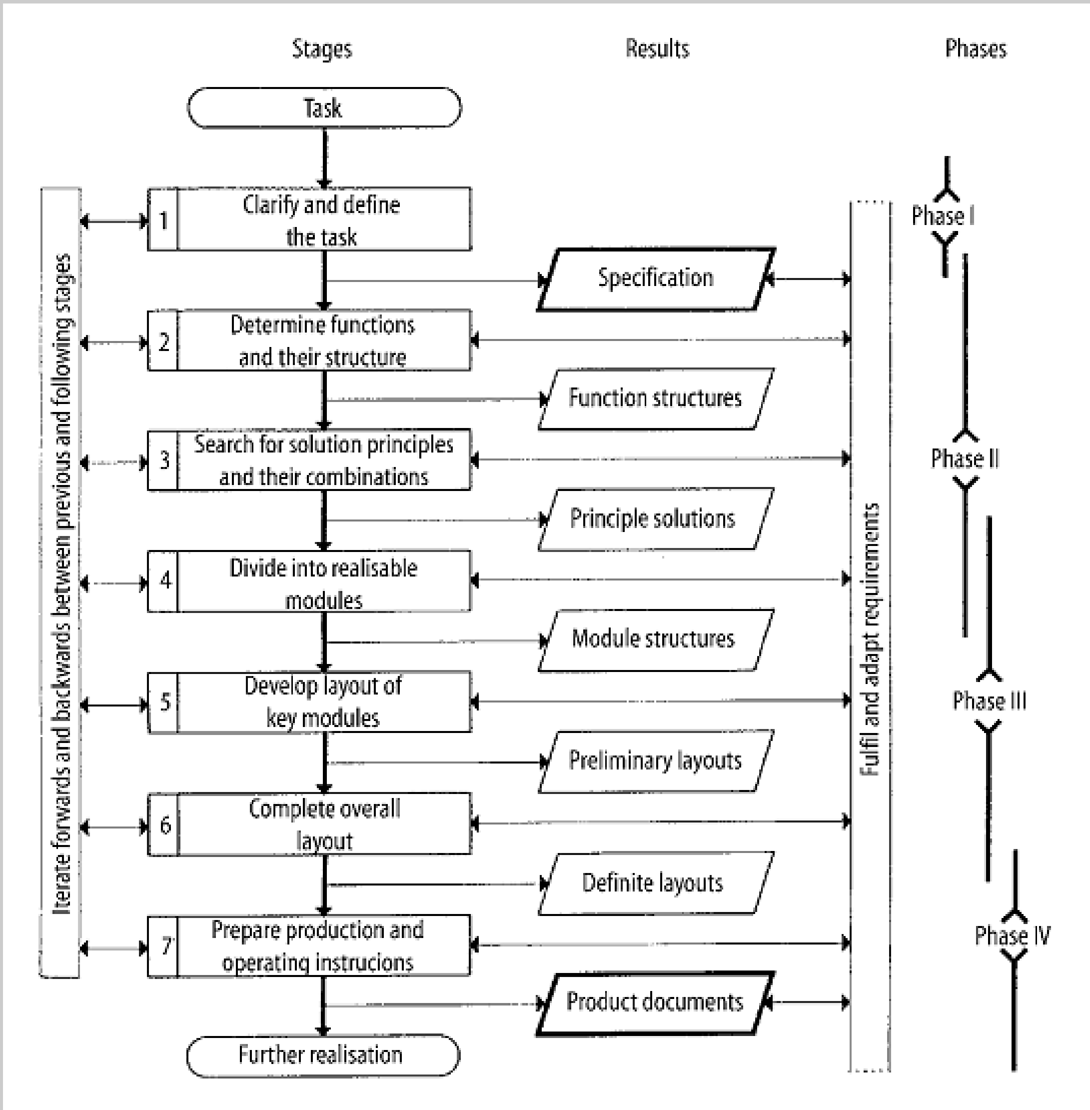


Fig 2.5 The engineering design process with four phases after (Pahl and Beitz 1996)

Four main phases of activities are defined: (1) Clarification of task(but it did not identify who related with the task) (VDI-2222) Conceptual design, (3) Embodiment layout, and (4) Detailed design. Each phase can be further detailed in sub-phases with associated working methods. The systematic approach is based on the believe that engineering design must be carefully planned and systematically executed.

2.4.2.2 Artificial Inteligence (AI) based Design approach

The AI based Design approach aims at creating computer software and hardware that imitates the designer's knowledge representation and reasoning. Early AI based design researches view the design as a problem-solving process of searching through a state space (Simon 1969). One of the most tangible results is the development of knowledge-based design systems in which design knowledge and requirements are modeled using logic so that they can be processed by computer (Sriram, Stephanopoulos et al. 1989; Gero and Rosenman 1990; Dym 1994; Brown, Waldron et al. 1998).

In this approach, the product information is the abstract of computer representation and expression. The computer expressions of the product design are as follows: Language, Geometric model, Graph tree, Objects, Knowledge model, Images, as shown in figure 2.6.

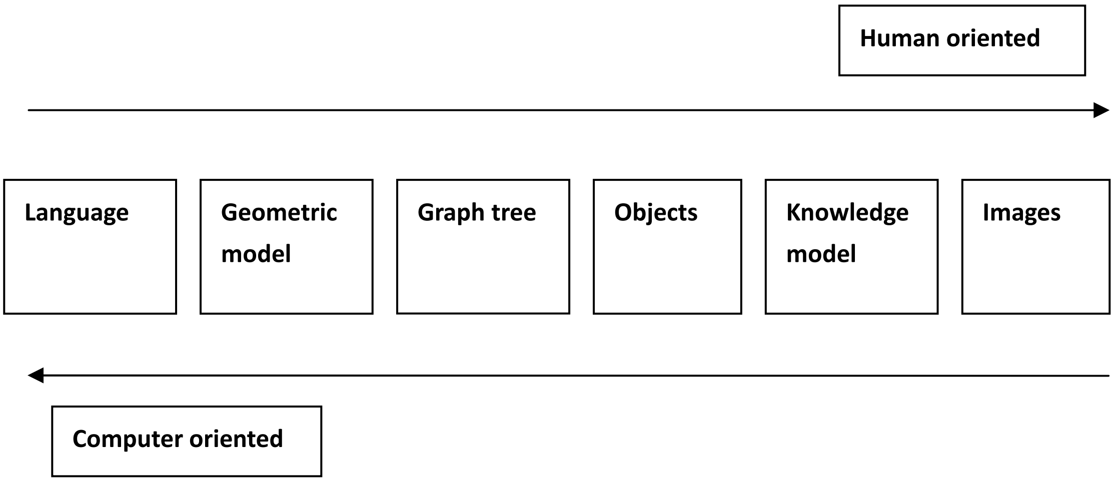


Fig 2.6 Computer expressions of the product design

And according to our analysis of these references, we make the comparison of different computer expressions of the product design, as shown in table 2.2.

Table 2.2 Comparison of different computer expression of the product design

Computer expressions	Researchers	Advantages	Disadvantages
Language	(Andersson, Makkonen et al. 1995)	Succinct and clear expression	It can't realise unify or share the different languages.
Geometric model	(Hsu and Woon 1998),(Lim, Duffy et al. 2001)	It's good for the description of product structure, in favour of follow-up phase of integrated design.	Not enough support for conceptual design
Graph tree	(Kusiak and Szczerbicki 1992),(Rudolph 2000)	It can describe all the characters of product and realize visualization.	Lack of class and inheritance
Objects	(Gorti, Gupta et al. 1998), (Martin and Roddis),(Tay and Gu 2002)	It contains abstractness and inheritance, and easy to implement reasoning.	Models don't have universality.
Knowledge model	(Sycara and Navinchandra 1992),(Zhang, Tor et al. 2001)	Reasoning process is easy to achieve.	The test of correct of the knowledge model, knowledge acquisition and knowledge base management need further improvement.
Images	(Kavakli, Scrivener et al. 1998), (Verstijnen, Van Leeuwen et al. 1998)	Close to human thinking. It can accelerate design process and easy to compare design results.	Difficult to achieve.

Takeda and Tomiyama (Takeda, Tomiyama et al. 1992; Tomiyama 1995) have developed a logical design model which views design as a reasoning process in which numerous logical deductions, abductions and circumscriptions are made about design requirements and solution. Figure 2.7 depicts the situation in which design is a process that converts requirements into a design solution under some constraints.

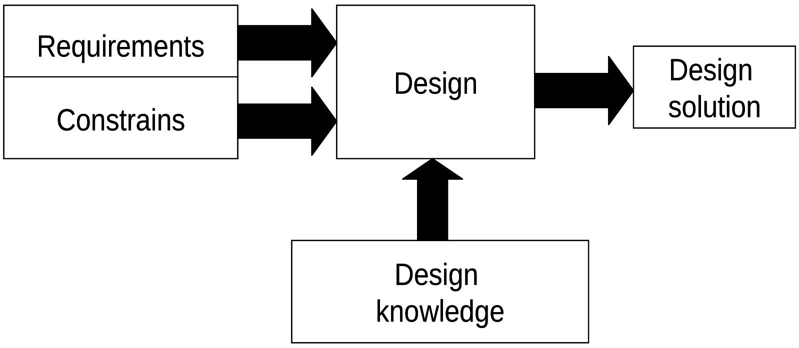


Fig 2.7 Knowledge-centered view of AI design

However, although the rule-based logic representation is well adapted to model knowledge of diagnosis problems, it is less easy for a designer to express his knowledge on design in terms of a list of rules. A survey shows that there is a shift of emphasis from logic based to autonomous, rational and interactive models of intelligent behaviour using agent technology (Vancza 1999).

2.4.2.3 Case-Based Design approach (CBD)

The Case-Based Design (CBD) approach proposes to re-use previous design solutions to solve similar design problems (Maher, Balachandran et al.). It is based on the fact that in most engineering design practices, new design solutions are adaptations of previous ones. Generally speaking, CBD paradigm is used within a same class/domain of artifacts. One of the main advantages is that it allows combining problem-solving with learning.

Gero et al. (Gero and Maher 1997) consider that Case-Based Reasoning (CBR) as a support environment for conceptual design is attractive for two reasons:

- The knowledge is represented as design cases that can be proprietary and/or familiar to the designer;
- The knowledge as case memory can be maintained and updated automatically with the use of the system.

However, one of the shortcomings is that the success of a design depends on largely the quantity and the diversity of cases stored in the cases base. Another common criticism of CBR is that being inherently conservative it is against creativity (Watson and Marir 1994).

2.4.2.4 Function-Based Design approach (FBD)

The type of knowledge, its level of granularity, and the operations on the knowledge needed in engineering design vary throughout the design process (Summers, Vargas-Hernández et al. 2001). However, some important information developed early in the design process (i.e., during conceptual design) needs to be maintained and accessible for the design engineer during the later stages of design. One of the more important types of information needed is the set of the required functions for the design.

In engineering design, the end goal is the creation of an artifact, product, system, or process that performs a function or functions to fulfill customer needs (Suh 1990; Dixon and Poli 1995; Ulrich and Eppinger 1995; Pahl and Beitz 1996; Ullman 1997). Modelling a design at the functional level and mapping these functions to embodied solution concepts aid the designer throughout the design process in validating the design against the requirements.

Umeda et al. (Umeda and Tomiyama 1997) pointed that the function is a critical aspect of a design, but has no clear, uniform, objective, and widely accepted definition. Pahl and Beitz (Pahl and Beitz 1996) defined function as the general input/output relation of a system whose purpose is to perform a task, typically stated in verb-object form. Cole (Cole Jr 2002) stated that functions are the actions, a system must perform in response to its environment in order to achieve the mission or goals given to it. Stone et al. (Stone and Wood 2000) defined function as a description of an operation to be performed by a device or artifact, expressed as the active verb of the sub-function.

The function definitions given in the design literature are diverse and even contradictory, but can be categorized according to the three main viewpoints (Deng, Britton et al. 1998) :

- System viewpoint: In this case, a function is viewed as a relationship between the input, the output, and the stated variables of a system. When a system transforms inputs to outputs, it exhibits a particular function.
- Performance viewpoint: In this case, a function is viewed as an abstraction of physical behavior. For example, consider a mechanical product that performs a specified behavior in a specified situation (working conditions), and these achieve the same results. The set of behaviors defines a functional class, and the results are its functions.
- Designer viewpoint: In this case, a function is viewed as a description of the design intention (i.e., the intended purpose of a product).

A good definition of function should take into account all these viewpoints. In this research, the focus is on the mechanical product functions that can be produced by the product or by some of its components.

A well-known model based on Function-Based Design is the Gero's Function–Behaviour-Structure (FBS) model (Gero 1990). The FBS model represents designing by a set of processes linking function, behaviour and structure together, which can now be seen as different states of the developing design, as shown in figure 2.8.

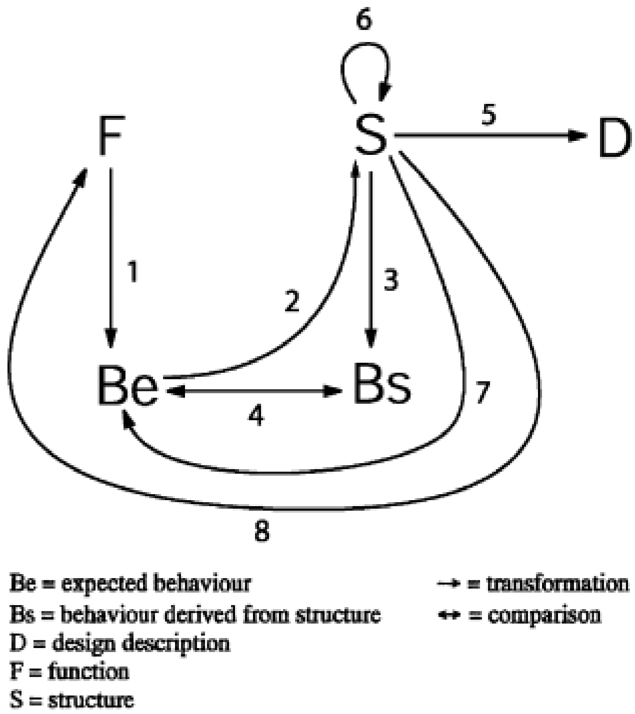


Fig 2.8 The FBS model after (Gero and Kannengiesser 2004)

According to the FBS model, designing an artefact involves a series of elementary steps which 'transform'. First, the desired function of the artefact (roughly, its purpose) into its expected behaviour (which will bring about the function); then the expected behaviour into a structure (intended to enable the artefact to exhibit the expected behaviour). After further steps of analyzing the structure for its actual behaviour (evaluating it against the expected behaviour, and possibly reformulating the expected behaviour) the structure is finally "transformed" into a design description from which an artefact may be produced (Gero and Maher 1997; Gero and Kannengiesser 2004). The sequential and loop-back steps are listed in the table 2.3.

Table 2.3 The FBS design steps

Step 1: formulation	$F \rightarrow B_e$	Transformation of the posited functions into behaviour that is expected to enable these functions.
Step 2: synthesis	$B_e \rightarrow S$	Transformation of the expected behaviour into a structure that is intended to exhibit this behaviour.
Step 3: analysis	$S \rightarrow B_s$	Derivation of the actual behaviour of the structure.
Step 4: evaluation	$B_s \leftrightarrow B_e$	Comparison of the actual and expected behaviour.
Step 5: documentation	$S \rightarrow D$	Production of the design description.
Step 6: reformulation 1	$S \rightarrow S'$	Choice of a new structure.
Step 7: reformulation 2	$S \rightarrow B_e'$	Choice of new expected behaviour.
Step 8: reformulation 3	$S \rightarrow F'$	Choice of new functions.

According to Gero, the experiential knowledge about function ,behaviour and structure that a designer needs to do all this, is brought together in design prototypes (Gero and Maher 1992; Gero and Kannengiesser 2007).

2.4.2.5 User-Centered Design approach (UCD)

The term ‘user-centered design’ is originated in Donald Norman’s research laboratory at the University of California San Diego (UCSD) in the 1980s and became widely used after the publication of a co-authored book entitled: User-Centered System Design: New Perspectives on Human-Computer Interaction (Norman, Lewis et al. 1986; Norman 1990).

In broad terms, UCD is a design philosophy and a process in which the needs, wants, and limitations of end users of an interface or document are given extensive attention at each stage of the design process (Greenbaum and Kyng 1991). User-centered design can be characterized as a multi-stage problem solving process that not only requires designers to analyze and foresee how users are likely to use an interface, but also to test the validity of their assumptions with regards to user behaviour in real world tests with actual users (Toffler, Toffler et al. 1995).

2.4.2.6 Design For X approach (DFX)

A wide collection of specific design guidelines are summarized under the label Design For X (Huang 1996), with each addressing a particular issue that is caused by, or affects, the characteristics of a product. DFX can be described as a critical success factor, and when properly implemented, will ensure that a product can be manufactured and tested. DFX methodologies address different issues that may occur in a phase of a product life cycle: Development phase, Production phase, Utilization phase, and Disposal phase.

The design guidelines usually propose an approach and corresponding methods that may help in generating and applying technical knowledge in order to control, improve, or even to invent particular characteristics of a product. DFX is a variable with many aspects, such as design for manufacture (O'Driscoll 2002) and design for assembly (Boothroyd and Dewhurst 1987) which make a product easier to produce with lower costs; design for disassembly (Crowther 1999), design for recyclability (Van Schaik 2001) and design for lifecycle (Ishii 1995) which make the designer plan ahead for product processing for after its useful life; design for environment (Rose 2000) which focuses on environment safety and health related issues and thus can help reduce the indirect cost of a product; design for quality, design for maintainability (Ivory, Thwaites et al. 2001) and design for reliability (Ireson, Coombs et al. 1996) which can also be assured by design and process control rather than by expensive testing, diagnostics and rework.

Starting and maintaining a DFX program is not easy, it takes time and dedication, but the results are worth the effort. Ideas and expectations should be clearly defined before the program is started. Strategic guidelines are highly recommended: (Buttars and Rowland 2006)

- DFX must be part of the corporate culture; management must support and encourage DFX.
- DFX should be driven by customer needs, i.e., understand the customer`s wants and desires.
- DFX requires teamwork and creative thinking; management must support teams and open thinking.
- DFX must have measurable and justifiable goals; define the key metrics: cost, yield, delivery, etc.
- DFX must be easy to use and apply; create and document methods and procedures. Use industry guidelines and standards.

DFX concepts are embraced by a commitment to design all human-technology elements and processes with full consideration of user performance capabilities. Design for the user via human factors engineering and implementation of systematic training programs are the principle means for developing and sustaining human performance effectiveness.

2.4.2.7 Eco-Design approach

The concept of Eco-design is an approach to design of a product with special consideration for the environmental impacts of the product during its whole lifecycle (Tischner, Schmincke et al. 2000). The goal of such an approach is to eliminate undesirable or potentially hazardous effects on the environment. One way in which manufacturing industry can reduce the impact it has on the environment is for it to adopt 'eco-efficiency' approaches. In particular, 'eco-design' is increasingly viewed as being key to sustainable and improved product development (Tischner, Schmincke et al. 2000).

O'Brien (O Brien 2002) addresses the sustainability of the design and manufacture of products, while presenting a closed-loop concept for industry, in which inputs of raw materials and return of waste to the environment must be minimized or eliminated. Nakashima et al. (Nakashima, Arimitsu et al. 2002) explicitly handle the product recovery system, in which parts and materials of the products are reused and recycled in order to minimize waste and environmental damage. To that end, some research has been conducted to address the disassembly process used for product recovery (see, e.g., (Dini, Failli et al. 2001)).

Shu-Yang et al. (Shu-Yang, Freedman et al. 2004) present seven principles of eco-design and conclude that any form of design that minimizes the environmental impact by emulating and integrating with natural ecosystems can be referred to as eco-design. Nissen (Nissen 1995) gives a list of traits which characterize an eco-product—a product that already incorporates environmental considerations within its design:

- the material used is a plentiful natural resource;
- the manufacturing process requires only a low consumption of natural resources;
- the emission of hazardous waste in the production processes minimal;
- when in use, the product is relatively environmentally sound;
- environmentally sound remanufacturing or recycling processes can be easily applied after use;
- and when finally discarded, the environmental impact of disposal/incineration is minimal.

2.5 Conclusion of the chapter

In the fierce competition of the global market, design is more and more regarded as a vital asset and the main source of the competitive advantage of a company. Design is a process of divergent thinking and creative design. It contains multi-trade approaches which allow taking into account different kinds of criteria. It is the solution of a function used to meet different kinds of techno-economic indicators, and is intended to establish the optimal plan from different possible proposals (Feng, Nederbragt et al. 1999). It is widely known that the bulk of the production cost is incurred at the end of the design process (Hsu, Chuang et al. 2000), and as a result it is therefore crucial to avoid errors in the early design stage.

From the review of the literature discussed above, it is possible to identify several attributes that are universal to the design theories and approaches. These are presented in Table 2.4.

Table 2.4 Universal attributes of design theories and approaches

Attribute	Explanation
Focus	Designers are required to adopt a change in design thinking, start asking different questions and to stop depending on “rule of thumb”.
Design thinking	Systemic thinking, considering the system as a whole, identifying relationships between the elements of a system.
Various perspectives	Include stakeholders within the design process, identify of real consumer requirements, and include the provider throughout the lifecycle.
Expandable network	The development of a network of partners to provide various knowledge.
Collaboration	Collaborative relationships across disciplines and companies.
Context extension	The design of solutions that can be realized across various contexts.
Customization globalization	The design of industrialized solutions based on a global platform which can be customized.
System-level innovation	Designers are encouraged to develop solutions at system level rather than introducing incremental changes.

Most current technical approaches stop at the functional level, without analyzing how the overall system (system-user) could behave in perform these functions. It is known that the user’s perception of a system is quite different from the designer’s (Stalker 2002). Additionally, involving a range of users in design by adopting an inclusive approach has been identified as an important way through which companies can manufacture more successful systems (Clarkson 2003; Gyi, Cain et al. 2010). To separate system technology from user-related features, it is necessary to split the notion of a system into two separate components: technical solutions and user-related features (Takala 2005). The strategy of knowledge management is not widely adopted for innovation in industries due to a lack of an effective approach of integration between user knowledge and technical knowledge (Xu, Houssin et al. 2011).

According to the review of the design theories and approaches discussed above, we find that product design is usually performed simply taking into consideration product functions and structures, while users’ behaviours in terms of using the product are generally not fully considered during the early design phase. So, in order to improve product performance, our research targets a better integration of product and user behaviour during the early design phase. In the next section, we will introduce the global view of the behavioural design approach.

Chapter 3 Behavioural Design Approach (BDA)

In the previous chapter, we take a multi-level and comprehensive strategy for researching the multidisciplinary state of the art in the domains of engineering design, design theories and design approaches. In this chapter, the essential propositions and hypotheses of our study are proposed. According to the focus of our research, the overall research objective of our study is to propose a global view of behavioural design approach as a feasible solution to improve system performance starting from the early design phase, and a model of the use task (performed by the user or by the system itself) required to realize the mapping of the behavioural design approach is also proposed. The chapter is planned in the following mode.

3.1 Research questions and general assumptions

A system's behaviour is studied only from a technical point of view in order to verify its reliability and potential problems in the detailed design phase. However, this behaviour is neither characterized nor studied from a using point of view. Nowadays, although designers do increasingly have some understanding of user behaviour, they rarely pay much attention to the behaviour which derives from the structure (how the structure will move to fulfill the function), and the behaviour which is fulfilled by the user (how the user will react to the product). To fulfill these functions today, the designer uses Functional Analysis (FA) to select a structure. In this step, the designer proposes the structure that could be adopted to fulfill the function and imagines its behaviour without any verification nor simulation. To validate this structure, the designer only considers the criteria proposed by the FA, which are often functional criteria. At the end of the system development cycle, the designer verifies whether the product respects established standards, and if it does not, then the designer modifies its structure (e.g., by adding safeguards, replacing a material with other more recyclable materials, etc.). The system could very well meet the designer's objectives, but not satisfy those of the end-user; where a machine could comply with current safety standards at a technical level, but still be perceived as unsafe by the user (Mondragon 2005).

3.2 The global view of the behavioural design approach

In order to design a complex product, it is necessary to define not only functions, and then the structures fulfilling these functions, but also their behaviour. Our approach concentrates on a system's (structure's) behaviour and on a user's behaviour. On various occasions, the user has to carry out one or more faulty works. It requires that the user intervene on a running machine, which may be in a dangerous zone when the system is operating or stops. For our behavioural design approach, designers on the one hand find out the technical solutions to fulfill some of the technical functions defined in the functional analysis, and on the other hand, when they do not find feasible technical solutions for the other functions, or due to cost reasons, they propose the functions to be performed by the user. We seek to minimize the differences between the conceived working situation imagined by the designer and the real ones at the end-use site (Houssin, Bernard et al. 2006; Pilemalm, Lindell et al. 2007).

3.2.1 The concepts of function, structure and behaviour

We saw in Chapter 2 that the mechanical engineering design process is normally considered to integrate the phases of analysis, synthesis and evaluation which advise one another through a series of feedback cycles.

Design process is a process of divergent thinking and creative design. It is the solution of a function used to meet different kinds of techno-economic indicators and intends to fix the optimal plan from different possible proposals (Gardoni 2005). It is common knowledge that the majority of the product cost is committed by the end of the design (Sieger and Salmi 1997; Al-Salka, Cartmell et al. 1998). To avoid errors in the early design stage is therefore important and necessary.

The first step to realize mechanical engineering design is to establish process modelling. Process modelling describes the work job of early mechanical engineering design which is usually supported by some known methods. The main implementation of the design is the mapping of function and structure. Most of process modellings are extended from two basic framework: Function-Structure (FS) (Suh 1990; Pahl and Beitz 1996; French 1999) and Function-Behaviour-Structure (FBS) (Gero 1990; Qian, Yu et al. 1990; Goel 1997; Umeda and Tomiyama 1997; Deng, Britton et al. 1998; Shimomura, Yoshioka et al. 1998; Zhang, Tor et al. 2001; Labrousse 2004).

In FS framework, a pre-consideration should be taken that the relationship between function and structure is direct. So, the main idea of this framework is to directly seek the structure which corresponds to the function. This type of framework usually includes:

- Function decomposition,
- Mapping of function and structure,
- Structure combination.

Moreover, the early mechanical engineering design is described by function and structure in two hierarchies. Normally, the method FAST (Fowler 1990; Norm 1996) is used to mobilize this approach.

A lot of discussions can be found on the design model in the context of engineering design. Various proposals on the design model contain to a certain degree notions such as function, behaviour, and structure. Since 1990 (Gero 1990), the model of Function-Behaviour-Structure (FBS) has become a popular design method for the early engineering design.

This type of framework considers that function and structure must be linked through behaviour,

- to depict the action that is executed for the completion of function,
- to indicate "how structure fulfils the function".

So, it is regarded as an ordinal process of mapping of function, behaviour and structure.

FBS framework is first brought forward by (Gero 1990), who points out that the structure expresses the internal and external states of a physical element. Based on Gero's proposition, there are a variety of the extended models. According to the expression of Function-Behaviour-Structure, Umeda (Umeda and Tomiyama 1997) develops a function-behaviour-state modeler that reasons about function by means of two ways: causal decomposition and task decomposition. They hold that this level represents the elements of an artifact and relationships b them. Deng (Deng, Tor et al. 2000) devises a dual-step Function-Environment-Behaviour-Structure (FEBS) modelling framework. In this framework the causal decomposition of function has been extended by incorporating the working environment of the system-being-designed so that the modelling hierarchy is more comprehensive. Labrousse (Labrousse 2004) develops a FBS-PPR model that has some major contributions for capturing product and its associated processes in a global and multi-representation model.

The FS framework and FBS framework and their extensions are aimed to integrate all product related data and knowledge over the lifecycle. These researchers conclude that behaviour serves as a platform of reasoning between the two: function and structure. They indicate that there is away from function to behaviour and from structure to behaviour; however they haven't offered any methods or tools to help the designer to study and evaluate behaviour from the design stage.

For the purpose of figuring out some crucial issues talked about above, in the following section, we try to propose an approach based on the concept of function, structure and behaviour to support behavioural design tools in the lifecycle stages.

3.2.1.1 The function concept

When a designer is assigned a mission to design a mechanical product, it is initially determinate by the desired functions of the final design output. So the designer's aim is to bring forward a mechanical product constituted of parts such that the assembled product offers the wanted functions. In general, designers have the same opinion that function is the most important concept in determining a mechanical product's fundamental characteristics (Chakrabarti and Bligh 1994; Pahl and Beitz 1996; Ullman 1997; Stone and Wood 2000; Houkes and Vermaas 2004; Vermaas and Houkes 2006; Erden, Komoto et al. 2008), because products with problems in their main functions will never sell, no matter how complicated their specifics.

There are many various, even contradictory definitions of function, with different researchers (Pahl and Beitz 1996; Shimomura, Yoshioka et al. 1998; Stalker 2002; Gero and Kannengiesser 2007) putting down on different significations either to signify the purpose or the action of a design. The lack of an exact definition for functions and different functional models of mechanical products brought by different designers cast doubt on the usefulness of descriptive and prescriptive design methodologies.

Collins et al. (Collins, Hagan et al. 1976) develop a list of 105 unique mechanical functions, here, the mechanical functions are listed to helicopter systems and do not use any classification scheme. Hundal (Hundal 1990) formulates six function classes complete with more specific functions in each class, though does not explain the real mechanical design functions. Koch et al. (Koch, Peplinski et al. 1994) use the 20 subsystem representations from living systems theory to represent mechanical design functions. Pahl and Beitz (Pahl and Beitz 1996) list five generally suitable functions and three types of flows, but they are at a very high level of abstraction.

Gero (Gero 1990) defines function as an intermediate between the goal of human and behaviour of a system. Designing in its original form is, according to Gero, an activity in which a set of posited functions are transformed into design description of artefacts that can perform these functions. These functions originate from clients, and the design descriptions, contain the information sufficient for manufacturing the artefacts. The functions are, according to Gero, not directly transformed into design descriptions but via a series of elementary design steps in which also the behaviour of artefacts and their structure are considered. Function is considered by Umeda and Tomiyama (Umeda, Tomiyama et al. 1995) as a bridge between human intention and physical behavior of artifacts. The authors state *"there is no clear and uniform definition of a function, and moreover, it seems impossible to describe the function objectively"*. The subjective character of function and its being a link between intentions and objects are recognized by many other function modelling researchers including Chandrasekaran and Josephson (Chandrasekaran and Josephson 2000), Deng et al. (Deng, Tor et al. 2000), Stone and Wood (Stone and Wood 2000), Keuneke (Keuneke 1991) and Gero (Gero 1990). Rodenacker et al. (Rodenacker and Sch fer 1978) define function as a relationship between input and output of energy, material, and information, and this definition is widely accepted in design research (Welch and Dixon 1992; Pahl and Beitz 1996). Bracewell and Sharpe (Bracewell and Sharpe 1996) represent functions based on extending the bond graph technique (Rosenberg and Karnopp 1983), which introduces the concepts of "flow" and "effort to cause a flow" in the system. Value engineering represents function in the form of "to do something" (Miles 1972). This representation as "verb + noun," which again shares subjectivity to some extent, is noted to be incapable of avoiding inappropriate modelling (Kitamura and Mizoguchi 2004). Vermaas and Dorst (Vermaas and Dorst 2007) claim that the definition of function creates a conceptual continuity with other domains of knowledge, i.e., renders the function concept compatible with similar concepts of function used in biology, psychology, and sociology.

According to the above analysis, the functions in the TRIZ (Altshuller and Rodman 1999) and Functional Analysis(FA) (Little, Wood et al. 1997; Stone and Wood 2000) are expanded and reclassified and the standard set of functions is presented, which is a set of functional descriptions to describe all mechanical design functions, sub-functions, as shown in Table 3.1.

Table 3.1 Abstracted list of functions and sub-functions

Functions	Sub-functions
Create	Synthesize, Produce
Change	Increase, Decrease, Convert, Form, Control
Combine	Mix, Embed, Assemble, Connect
Separate	Disassemble, Decompose, Extract, Clean
Accumulate	Absorb, Store, Concentrate
Move	Move, Transfer, Rotate, Vibrate, Lift, Orient
Measure	Determine, Detect, Measure
Preserve	Preserve, Prevent, Stabilize
Eliminate	Destroy, Remove

The object of moving between functions is a flow, which is divided into material, energy and parameter flow based upon the work by (Pahl and Beitz 1996), (Stone and Wood 2000)and (Kitamura and Mizoguchi 2004). Flow is expressed as noun. The standard set of flows is presented, which is a list of flows, sub-flows and complements, as shown in Table 3.2.

Table 3.2 Abstracted list of flows and sub-flows

Flow	Sub-flow
Material	Solid, Liquid, Gas, Geometric Objects, Loose Substances, Porous Substances, Particles, Molecular and Plasma, Chemical Compounds
Energy	Forces and Motion, Thermal Energy, Electric Field, Magnetic Fiels, Electromagnetic Wave of Light, Nuclear Energy and Activity
Parameter	Solids Parameters, Geometric Parameters, Fluids Parameters, thermal Parameters, Electromagnetic Waves of Light Parameters

In the early phase of mechanical engineering design, most of design decisions taken are concerned with the desired characteristics and the overall functions of the assembly. In this phase, the abstract functional specification of an artifact is transformed in to a physical description. In the later phases of design, the physical decisions that are made in the earlier phases are elaborated to ensure that they satisfy the specified functional requirements and life cycle evaluation criteria. To manipulate the function information, a functional data model (that describes the functional information through the design cycle) is needed so that appropriate reasoning modules can interrogate and extract functional information during the decision-making processes (as the geometric reasoning modules query data from the product data model (CAD model) during the shape design process) (Roy, Pramanik et al. 2001).

We can conclude that there are three roles for function in design:

- Function is used firstly as a modelling language by which designers can compose and develop their requirements.
- Function also serves as product representation which can connect requirements and product.
- Function is used to evaluate the product to know how much their intention is satisfied after construction and deliberation of product function representation.

So, according to the literatures, we give the function definition as follow:

Function (F): fulfill the customer requirements, and depict the purposes of the production system (product)

3.2.1.2 The structure concept

In mechanical engineering, structure is a body or assemblage of bodies in space to form a system capable of supporting loads (Pullan 2000).

Structure defines the different components of the product and specifies their geometry, dimensions, topology and other physical properties. The structure is derived from the functional specifications by satisfying functions with parts or sub-assemblies that realize each specific function. Note that several functions may be realized by the same sets of parts of by a same sub-structure. Thus each group of functions is associated to structures that consist of components and link involved in functions.

In mechanical engineering design, we can call structure as the physical structure. The physical structure of the product being designed consists of physical components that contribute to the required performance function, excluding those contributing to the other types of function, such as the assembly function, manufacturing function, market function and maintenance function (Deng, Britton et al. 1998; Deng, Tor et al. 2000).

The characteristics of a physical structure are defined by attributes. An attribute has a name, a value and unit, e.g. weight of 100N. Attributes may be static, such as weight, volume and temperature; or dynamic, such as velocity and acceleration. A particular type of material may have further attributes, such as specific weight, which is relevant to a specific design.

Structure is represented by entities, attributes of entities, and relations among entities. Entities are identifiers of products, and attributes of products and relations among the entities represent structures composed by the product.

So, according to the literatures, we give the structure definition as follow:

Structure (S): depict the elements of the production system (product) and their relationships to fulfill the function.

3.2.1.3 The behaviour concept

The term behaviour is used ubiquitously in mechanical engineering. It refers roughly to the way technical artefacts' behave in a given o hypothetical situation. It also plays a vital role in specific design methodologies since it allows connecting descriptions of the physical structure to descriptions of their technical functions. Although behaviour plays a pivotal role in mechanical engineering design, it is used in the engineering literature with various and possibly conflicting meanings. Different authors give different characterizations, and these are not consistent even in the works of a single author (Dorst and Vermaas 2005). We will not attempt to survey all meanings proposed, but introduce it into our behavioural design modelling.

Starting with the mentioned FBS model in chapter 2, Gero (Gero and Rosenman 1990; Gero and Kannengiesser 2004) defines the behaviour of a technical artefact (a designed object), its behaviour variables, as describing the attributes that are derived or expected to be derived from the structural variables of the object, i.e., what it does. And the structural variables describe the components of the object and their relationships, i.e., what it is. And they give as examples of behavioural variables of a window for instance thermal conduction and light transmission. Umeda (Umeda, Kondoh et al. 2005) defines behaviour as a transition of the states a long time where a state is depicted by entities, their attribute and their structure; a given example of behaviour is the electrical charging of a drum in a photocopier.

A product's function is achieved through certain behaviour or behaviours. Regarding the role of behaviour in achieving a function, the following points should be noted: Only under the working environment can behaviour produce its function. For instance, if we use a screwdriver to undo a screw with a badly slot, the behaviour of the screwdriver fulfils its function (twist out the screw), because the head of the screwdriver has nothing to act against. On the other hand, in an unintended environment, a product might achieve a certain unintended function. For example, apart from being used as a tool for enabling a person to drink water, a cup could be used for measuring (containing an approximate standard amount of liquid); or be used as a paperweight (Rosenman and Gero 1994). Unintended function is not considered in our work.

A physical structure has many properties and can demonstrate much behaviour beyond those intended by the designer. For example, when a bearing is supporting a shaft, its behaviour includes not only that of supporting the shaft, but also many others, such as dimensional distortion because of the force acting on the shaft, generation of heat because of the friction between the bearing and the shaft, and so on. The physical behaviour that can produce the required function is called structure behaviour (e.g. supporting the shaft). So, the structure behaviour which derived from the structure fulfils the functions according to functional-level design information.

At this point, the behavioural design approach influences the whole design process and associates the product representations to every project. Nevertheless, it must be completed to describe the product behaviours along with its lifecycle. Its specificity is related to the duality between the behaviour of the product (how it works) and the considered life-cycle phase (how it will be used). On the one hand, if major phases can be either forecasted or expected at the beginning of a project, the behaviours can be derived from some specific analysis. On the other hand, the behaviour depends on the life-cycle phase that we consider. The phase characterized product and user behaviour: a phase is viewed as a special kind of behaviour which is realized by user behaviour. Thus, the two behaviour paradigms, which must be distinguished and modelled separately, are:

- Behaviour issuing from an analysis structure, which will be later known as derived behaviour.
- Behaviour expected as a manual functional requirement for product design,

which will be referred to as the user behaviour paradigm

User behaviour is a subject of study that, from the perspective of safety, accessibility, usability, ergonomics, is to research the interactive relationship and function among user-machine-environment. It is employed by the thinking, methodology and theory of user-machine engineering. Functional allocation and decomposition between user and machine, user machine interface, working space, and information transmission are defined as its research object (Lewis and Rieman 1993; Huisg and Kohn 2009).

In the process of production, the startup, running and shutdown of the machines are done through the direct operation and adjustment by users. As controller of this process, user holds an active position in the relationship of user-machine. To some extent, the safety or ergonomics behaviour initiated by users may influence and safeguard the performance of machines.

So, according to the literatures, we give the behaviour definition as follow:

Behaviour (B): depict the property that the ideal origin of the structure which is to fulfill the automatic function (behaviour can be directly derived from the structure), i.e., structure behaviour; the behaviour derives from the user which is to fulfill the manual function, i.e., user behaviour.

3.2.2 Behavioural design modelling

We cite two examples observed in real companies:

- Example one: Designers want to fulfill the function, which is to transfer a movement. The structure needed to fulfill this function could be two rollers (as in a gearbox), and the behaviour of the structure is characterized by the two rollers turning in opposite directions. It shows that the user’s hand may be jammed between these two rollers when he operates the structure. The problem is as follows: when the user opens the door of this gearbox to be changed to intervene in the system, his hand might be close to the dangerous zone, as shown is Figure 3.1. Designers therefore build the cogs into the box. However, they do not research how the component (door) will guarantee the performance of the system (if the user opens the door, the system will be stopped, which decrease the system availability, and consequently its performance). Also, designers do not analyze the tasks performed by users when the users intervene on the system.

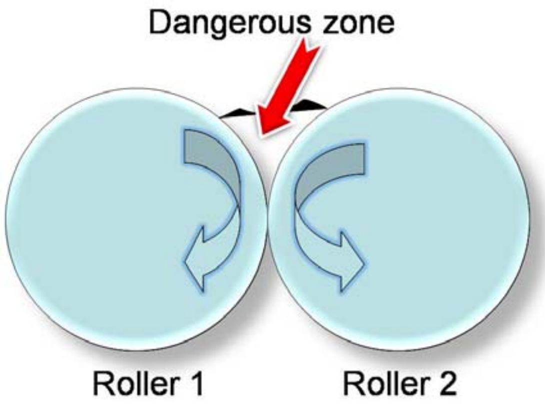


Fig 3.1 Dangerous zones between two rollers

- Example two: At a Printer Manufacturer's, in order to respect the legislation, designers added a door to forbid accessibility to an organ in the movement. They did that because at that stage, the system was already dimensioned and fulfilled the feature required in the functional specifications (Houssin et al., 2006). The problem was that when a user opened this door to act on the system, the door would stand in the way of the command console, where as the console had to be accessible during the operation. To respect legislation, the designer added some protection. However, they did not do any research to see how the component (door) would guarantee the performance of the system (if a user opened the door, the system would be stopped, which decreased the system availability, and consequently its performance). Furthermore, the designers did not analyze the tasks performed by users when they act on the system.

We herein propose a behavioural design approach that integrates user and structure behaviours from the early design phase. Behavioural design is a mechanical system design method based on multidisciplinary knowledge that takes into account, from its preliminary phases, the analysis and the specification of using tasks necessary for accomplishing the functions (Sun, Houssin et al. 2010). According to the two examples discussed here and other two examples in Chapter 1, we can identify two aspects of the concept of behaviour. The first involves behaviour carried out by the system according to the technical viewpoints. The second involves behaviour carried out by the users of the system or the correlative working team.

We propose below the global view of behavioural design modelling, as shown in Figure 3.2. It represents a set of steps linking together the three concepts (function, structure, and behaviour).

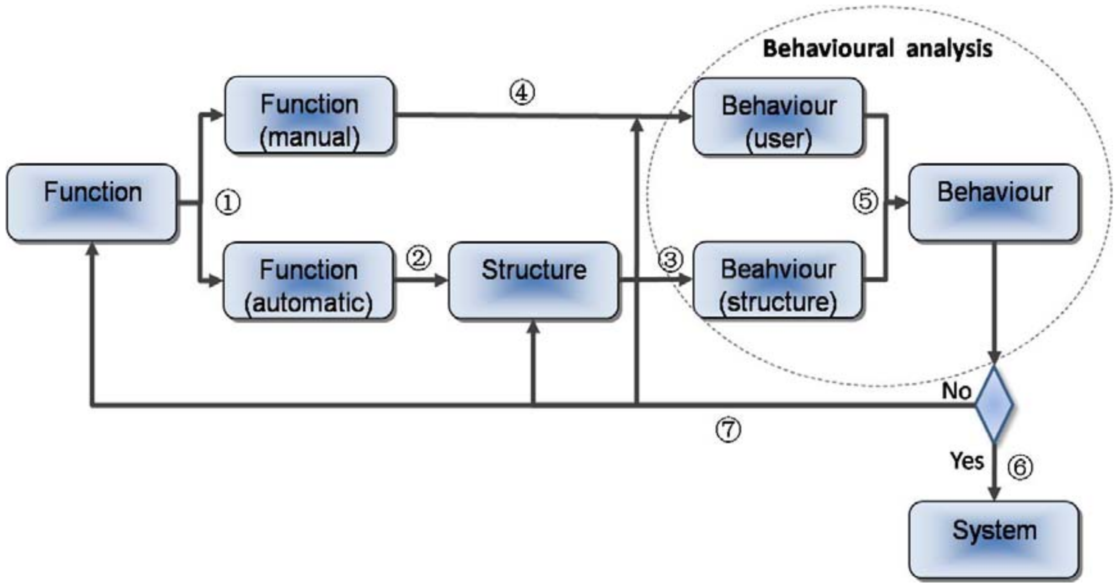


Fig 3.2 Global view of behavioural design modelling

Seven processes are listed here to describe the modelling procedure:

- Step ①: According to the functional analysis and requirements specification, we can divide the function into two parts. The first is the automatic functions realized by technical solutions; the second is the manual function fulfilled by the user, because of the cost or the difficulties related to automation.(not our research objective)
- Step ②: According to some methods, such as Functional Analysis (Conway 1990; Wixson 1999; Tan and Allada 2003), Axiomatic Design (Suh 2001), we could find the necessary structure to carry out the function. (not our research objective)
- Step ③: According to structure decomposition, we can obtain the behaviour of structure tasks (operation, motion, etc.) that the structure has to perform to achieve the function. (Task Model is the bridge between the structure and structure behaviour which will be introduced in the section 3.3. It is our study objective.)
- Step ④: Manual functions will be carried out by the user. Thus, in this step we propose identifying and studying the tasks performed by the user to fulfill manual functions. (Task Model is the bridge between manual function and user behaviour which will be introduced in the section 3.3. It is our study objective.)
- Step ⑤: To improve the performance of the system, we propose that the interaction between the structure's behaviour as well as the user's be analyzed. (The behaviour comparison is the key research in our work.)

- Step ⑥: If the behaviour of the structure meets the performance criteria (functionality, productivity, safety, cost, quality, etc.), designers can continue to develop the system.
- Step ⑦: Where the interaction between the user's behaviour and that of the structure does not ensure the needed performance, we have to change user's tasks, or go back to the structure level to modify the structure or go back to the function level to modify or change the function decomposition. We could also change the task performed by the user, which means changing the user's behaviour. (It gives the suggestions not the solutions which needs further research in the future work.)

We use UML Diagram (Booch, Rumbaugh et al. 2005) to represent the behavioural design modelling, as shown in figure 3.3.

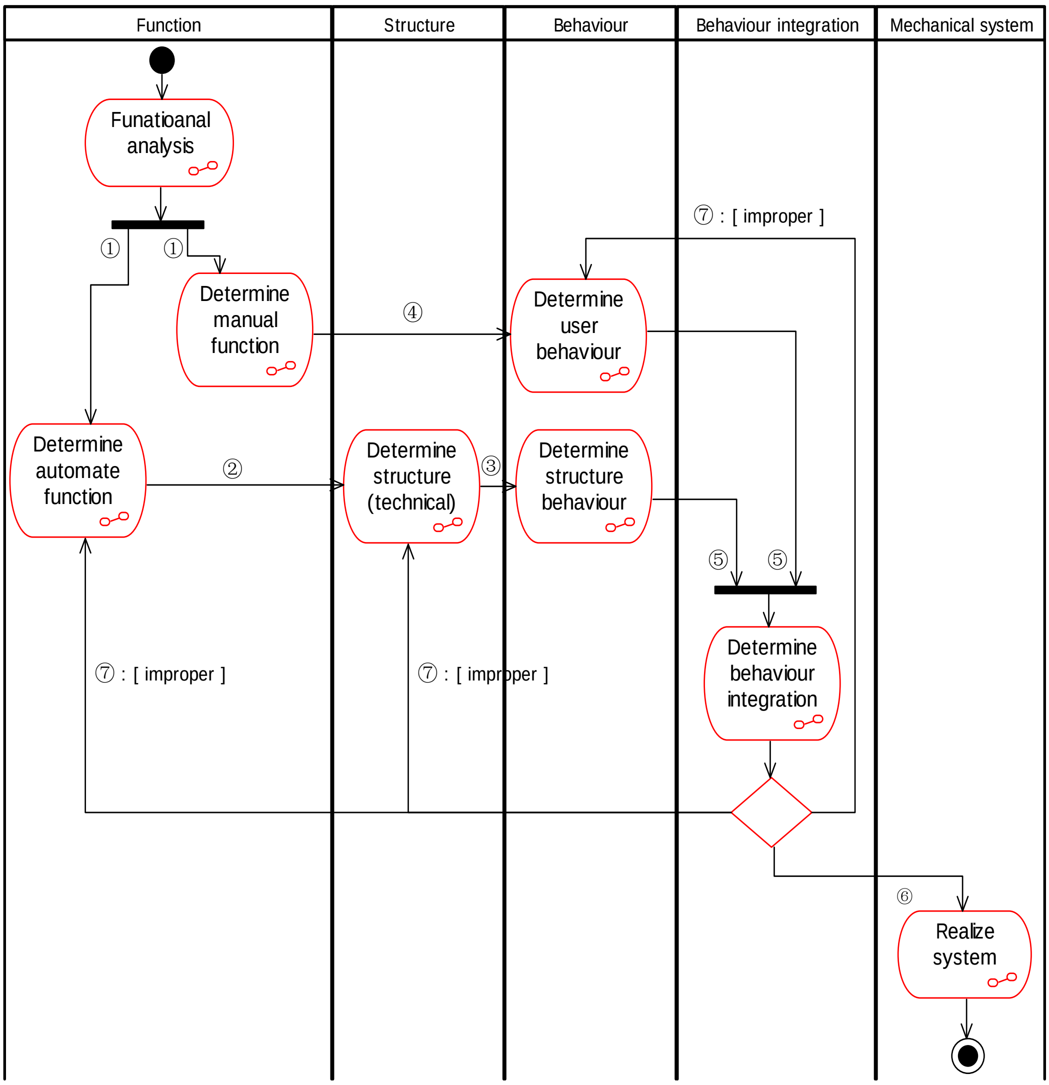


Fig 3.3 UML activity diagram of behavioural design approach

3.3 Task model

In order to implement our global view of the behavioural design approach at the early design stage, we introduce the task model (to be performed either by the product itself or the user). We adopt the definition proposed by Hernandez (Hernandez 1995): the task is a goal to achieve, which involves a determined change of an object's state. In other words, the behaviour of the product presents all the tasks to be performed by this product. Moreover, we take into account those tasks to be performed by the end-user of the product to assure the global performance demanded form this product. These tasks take into account the analysis and specification of the using conditions; that is to say, maintainability, user's safety (Coulibaly, Houssin et al. 2008), reliability and ways of system usage. Our approach is based on a "Task model" integrated into the FA. During the early phase of the design process, although system models are often primarily limited to geometrical aspects representing product-dimensioning and the associated functional surface qualities, they hardly or never take into account their behaviour and that of the future end-user and their interaction. In the following, we present our task model which will be integrated with our behavioural design approach.

Based on the previous model, the "behavioural design" approach is detailed, starting from the foundation of our conceptual model proposed in section 3.2. The task is not very well defined, and is used differently in different contexts. Here, it encompasses two concepts. Firstly, it is used to refer to purposeful activities performed by users; such activities may involve a general class of activities, or a specific case or type of activity. Secondly, it refers to activities performed by the structures or a series of structures. Tasks arise from the relevance between behaviours delivered by a design system and a principle used in the system.

Our behavioural design approach (Figure 3.2) concentrates on product (structure) and user behaviour. To achieve interactions between the user's behaviour and that of the structure (step ⑤), we should get data related to the structure (step ③) and user behaviour (step ④). Consequently, we introduce the task model into the mapping process. Before we introduce the task model, we must explain the relationship between the task and our behavioural design approach in the following section.

3.3.1 Task mode

Task can be characterized by its input, output relations. Generally speaking, a task has an input and output flow, which is called as the basic task, thus the task model has two poles, as shown in Figure3.4 (a). Most transitions from input to output with the task are controlled by the auxiliary flow, so the controllable task should be denoted with three poles, as shown in Figure3.4 (b). The control flow specifies the factors that can be manipulated to change the output intensity of an effect.

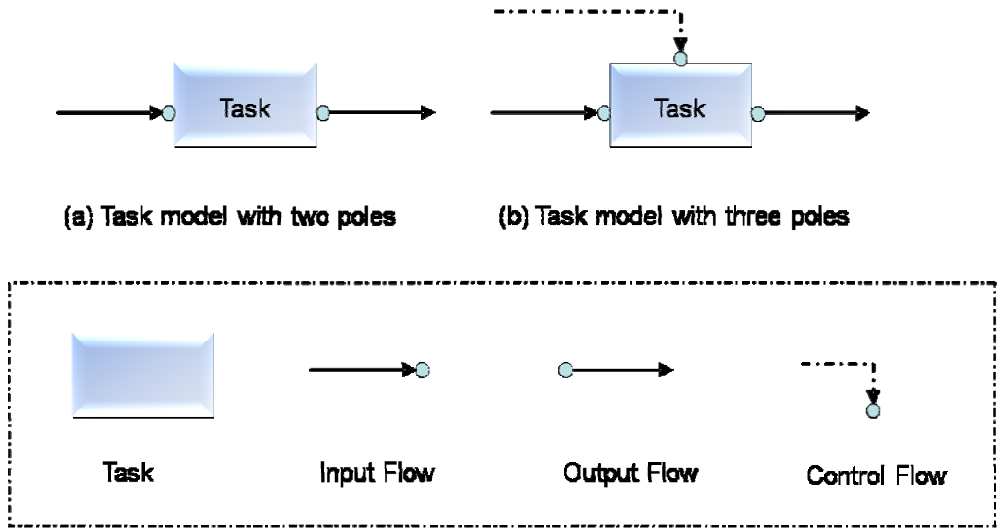


Fig 3.4 The concept of task mode

A task may have various input flows, output flows or control flows, so the task has various input poles, output poles or control poles (two, three or more).

Tasks can fulfill the transition from input to output, i.e., the occurrence of behaviours depends upon tasks. When a task is supposed to fulfill a specified sub-behaviour, it connotes that there exists at least a task, such that the sub-behaviour can be attained. Tasks can be linked to one another through its input or output ports and well-matched relationships among conterminous tasks, which affirm the causal relation and the structural relation of sub-behaviours. The behaviour can be realized by the following task modes, in which the directed connection typifies one or several flows. The task mode is shown in Figure 3.5.

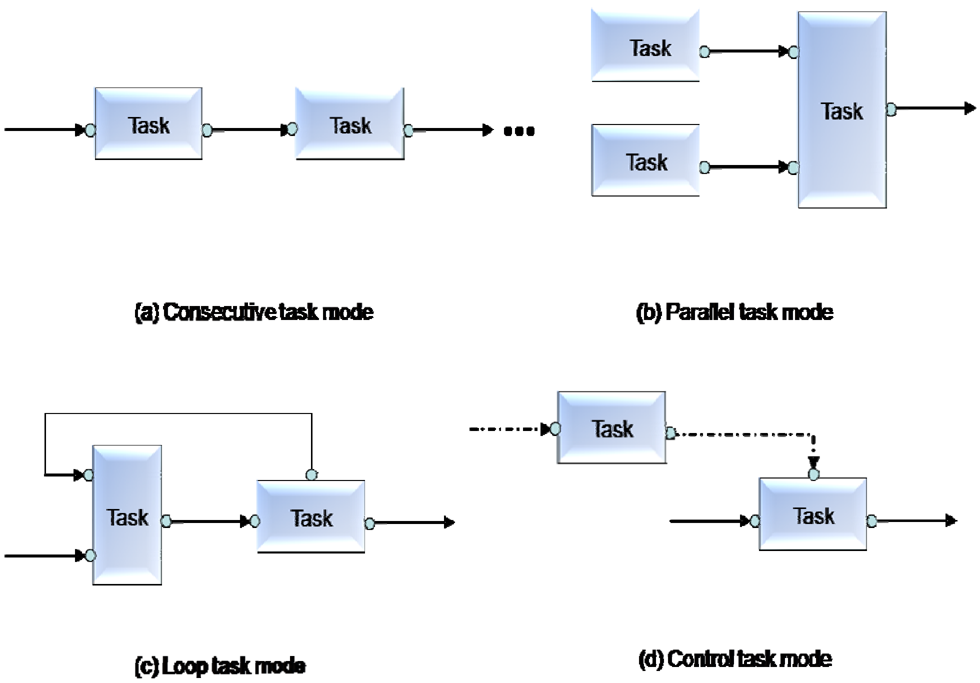


Fig 3.5 Task mode

- Single task mode: realize an internal behaviour by an effect, i.e., the behaviour only contains a single task, as shown in Figure 3.4(a). A task can fulfill behaviour. A behaviour can be fulfilled by several tasks respectively.
- Consecutive task mode: realize a behaviour by a series of tasks happening in sequence, as shown in Figure 3.5 (a).
- Parallel task mode: realize a behaviour by a series of tasks happening at the same time, as shown in Figure 3.5 (b).
- Loop task mode: realize a behaviour by a series of tasks, and the output of the later task is transmitted to the former task, as shown in Figure 3.5 (c).
- Control task mode: the characteristic of a task can be controlled by other tasks in order to control the accomplishment mode of behaviour, as shown in Figure 3.5 (d).
- Combined task mode: realize a behaviour by several above task modes.

3.3.2 "Task" Concept

As presented in previous works (Hasan, Bernard et al. 2003; Houssin, Bernard et al. 2006), the "Task" concept is characterized by some attributes and relationships. Here, we thoroughly detail this concept and consider that it is composed of Technical Tasks and Socio-technical Tasks:

3.3.2.1 "Technical Task" Concept

Automated tasks required from the system are known as technical tasks (Figure 3.6).

These tasks fulfill one or more system functions to be performed. This type of task is detailed, from a technical and automatic point of view in literature (Hernandez 1995), and is studied in the automatic research field to be better integrated into the design process. These tasks could be triggered automatically by another system or the user.

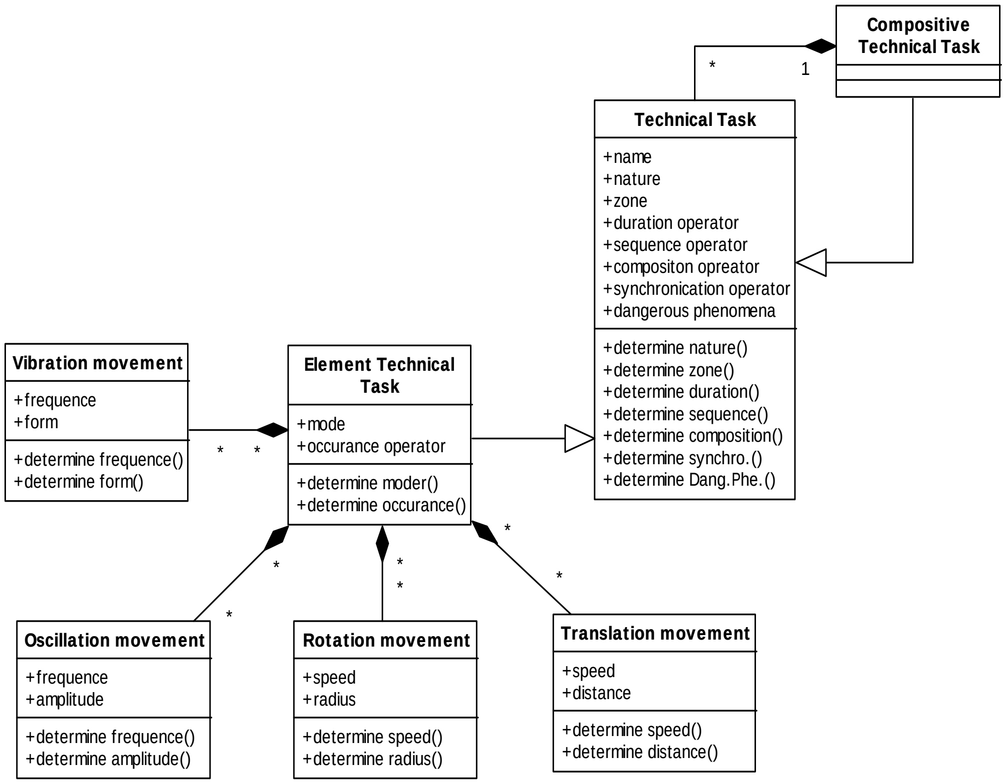


Fig 3.6 The "Technical Task" Concept

Here, we study these types of tasks from a automatic point of view. In other words, we study what the effects of these tasks are on the performance of a product of the use. Does it generate some dangerous phenomena, and for how long time, etc. The "Technical Task" could be characterized by:

- Name, duration, and Composition Operator, which represents the division of a task into many sub-tasks;
- Sequencing Operator, which determines the order and position of the task in a sequence of tasks;
- Synchronization Operator, which determines if the task must be in synchronization with other tasks; and
- Task Nature, which could be a Rotation, Translating, Oscillation or Vibration Task.

Figure 3.6 presents details of these concepts. This concept is also characterized by the dangerous phenomena that could influence the global performance of the system

(home-machine) (Hasan, Bernard et al. 2003; Houssin, Bernard et al. 2006).
This concept is decomposed into translation, rotation, oscillation, and vibration movements.

(1) "Translation movement" Concept

This concept represents the movement of translation that the product, or some of its components, makes to fulfill the function. It is characterized by its name, trajectory length, speed and occurrence. The occurrence allows us to know the repetition of this movement and, as a consequence, the repetition of its influence on system performance.

(2) "Rotation movement" Concept

This concept represents the movement of the rotation that the product, or some of its components, makes to fulfill the function. It is characterized by its name, rotation speed and occurrence. Moreover, the occurrence allows us to know the repetition of this movement and as a consequence of the repetition.

(3) "Oscillation movement" Concept

As with the other two concepts, the oscillation movement is characterized by its name, oscillation frequency, amplitude and occurrence.

(4) "Vibration movement" Concept

This movement is characterized by its name, vibration frequency, form (linear, nonlinear, sinusoidal, etc.) and occurrence.

3.3.2.2 "Socio-technical Task" Concept

This concept represents the tasks requested from the user's product to fulfill the functions, which could not be automated. These socio-technical tasks could be carried out by one or more users (Work Team). Tasks could be performed in an intervention mode (manual mode, maintenance mode, setting-up mode, repairing mode, etc.). As described by Hedrick et al. (Hedrick, Urbanic et al. 2004), each task could be simple or complex.
The "Socio-Technical Task" could be characterized by name, duration, composition operator (which represents the division of a task into many sub-tasks), sequencing operator (which determines the order and position of the task in a sequence of tasks), Synchronization Task Operator, and Task Nature (Physical, Mental and Sensory Tasks).So the tasks requested from the user could be split into three categories: mental, physical and sensory, as shown in figure 3.7.

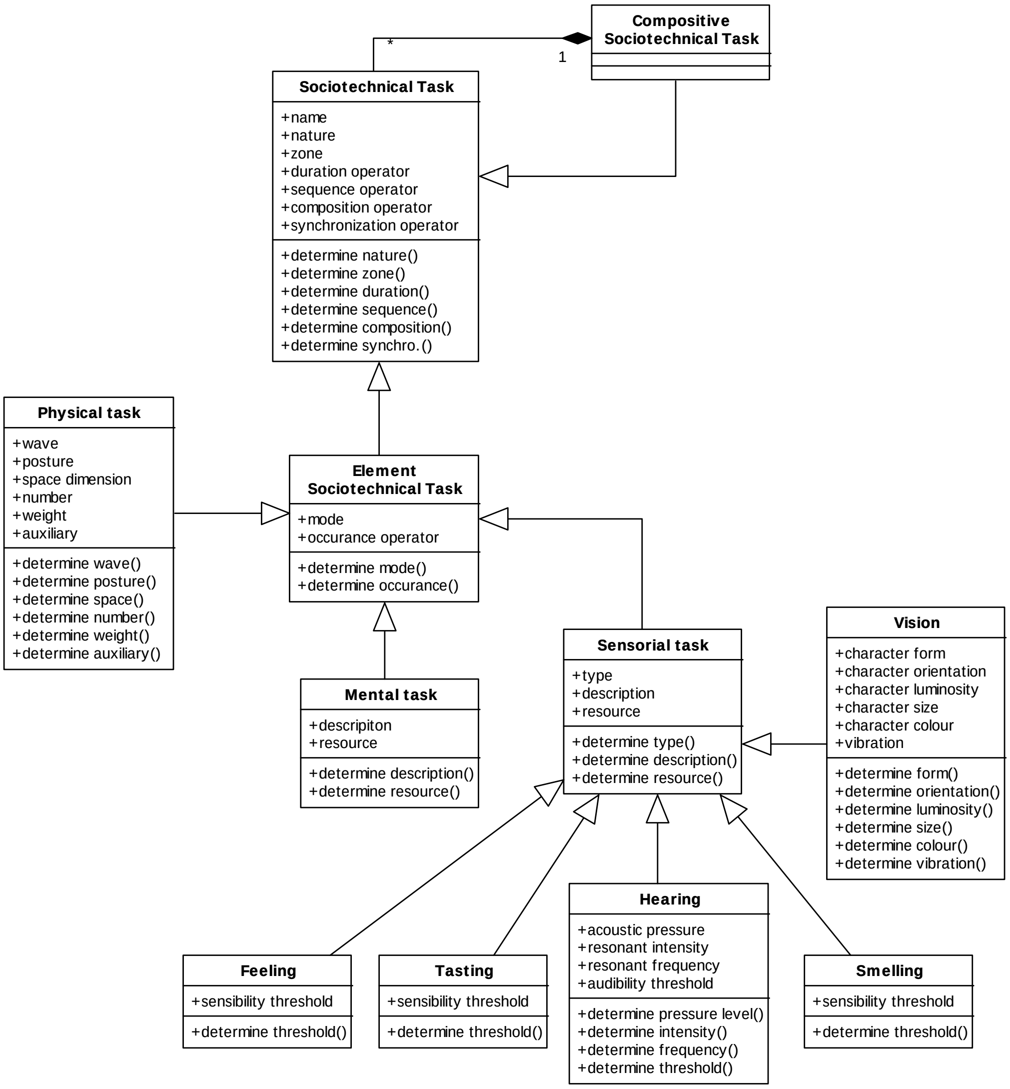


Fig 3.7 The "Socio-technical Task" Concept

(1) "Physical Task" Concept

This concept represents all gestures and postures required by the user to perform the task. Here, we assume that the mental activity required for physical tasks is negligible. This concept is represented by the "Physical task" class and its features and relationships. It is characterized by some attributes presented by Blanchard (Blanchard 1997):

- Name,
- Gestures to be carried out by the user to perform the task, and
- Postures needed to be carried out by the user to perform the task.

We complete these attributes by:

- Duration allotted by the designer to complete the physical task which should be identical to the real time spent to complete the task by user,

- Dimensions of the space needed to perform the task,
- Frequency: specified period between the performance of each task,
- Weight of handled objects: Its maximal value is limited by standards,
- Team work, which indicates the users who will perform the task.

(2) "Mental Task" Concept

Our contribution in this respect is limited to modelling mental tasks (described in detail in (Leplat 1990)) and integrating them into a formal approach within the design. Leplat (Leplat 1990) outlines that although it is nearly impossible to represent a mental task (because it is in the user's mind and thus unobservable), he demonstrated the importance of doing it anyway because of its crucial role in task planning.

In this context, the human user is defined as a plastic system that acts depending on the situation with which Rasmussen is confronted (Rasmussen 1983; Rasmussen 1997). This mental approach of the user determines work as a sequence of tasks to be performed. To do so, the user benefits from four information processing sequential stages:

- Perception: information research, detection, identification and acquisition,
- Assessment of the situation,
- Decision-Making, and
- Actions.

And three types of behaviour:

- Skill-based behaviour: semi-reflex action,
- Rule-based behaviour: the user is facing a situation he has already encountered and chooses a suitable procedure for recovery,
- Knowledge-based behaviour: the user is facing a new situation and has to follow all the stages of the decision-making procedure.

Moreover, each of the very complex tasks could be divided into many less complex sub-tasks, and be characterized by their abstraction levels, preconditions, post-conditions, sequencing and delegation (Chandrasekaran 1990). Based on these analyses, we could propose users' mental task parameters (knowledge, experience, competence, etc.). Here, we focus our modelling on the organizational point of view without focusing much on the human aspect.

The "Mental task" concept is represented by the "Mental task" class and its relationships. It is characterized by features shown in Figure 3.9.

(3) "Sensory Task" Concept

We could mention that sensory tasks do not require much mental activity (Stone and Sidel 2004). As with the "mental task" concept, we gather notions available in literature in order to adapt it to our approach. "Sensory Task" requires constant vigilance and attention, which means a high level of ability to continuously assess the situation in order to quickly detect the onset of an anomaly in the process.

A sensory task may involve one of the following five types:

- Vision: for example, seeing a light indicator to respond to. It is characterized by the following attributes (Tessier 1984): shapes, guidelines, brightness, size and color of characters, and workstation vibration.
- Hearing: for example, being able to listen to and understand a message such as beeps, or a variation of a machine noise.
- Smell: This represents the faculty needed to detect a particular smell (Olfac.univ-lyon1.fr).
- Touch: It represents the tasks performed by the user when using his ability to feel different kinds of sensations (heat, softness, viscosity, etc.).
- Taste: This type represents tasks performed by the user when using his ability to identify different types of substances.

3.3.3 The relationship between task, function, structure and behaviour

3.3.3.1 Task, Structure and Structure's Behaviour

From the technical viewpoints, the behaviour derives from the structure, which is called structure's behaviour (Figure 3.8). Structure may be broken down into sub-structure and sub-structure into sub-sub-structure, and finally into the elementary structure. Structure behaviour and structure are linked by task levels. In this level, we indicate that each elementary structure could perform one or more tasks, the set of which constitutes the structure's behaviour. Rather than a concrete structure, structure here refers to a technical solution. An indivisible element in the structure is called a primitive element. A primitive element can be either a physical or logical entity. Some elements group together and form a sub-structure or a structural element with well-defined characteristics.

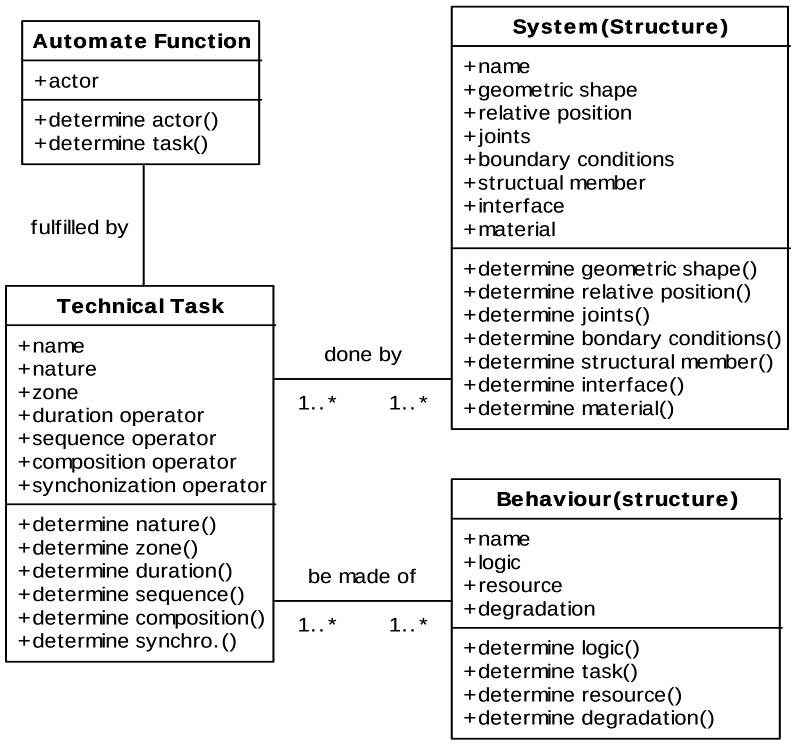


Fig 3.8 The relationships between task, structure and structure's behaviour

Structure is a physical task avatar, and the change of structure from one state to another must be caused directly or indirectly by the tasks. A structure contains structural features, such as what elements the structure is composed of, what the attributes of the elements are and how they are related. A task can be realized by various structures, and a structure can perform many varied tasks.

3.3.3.2 Task, Manual Function and User's Behaviour

According to socio-technical viewpoints, behaviour derived from the manual function is called user's behaviour (Figure 3.9). Based on the Functional Analysis, each elementary manual function could be carried out by one or more tasks; a set of these tasks constitutes the user's behaviours.

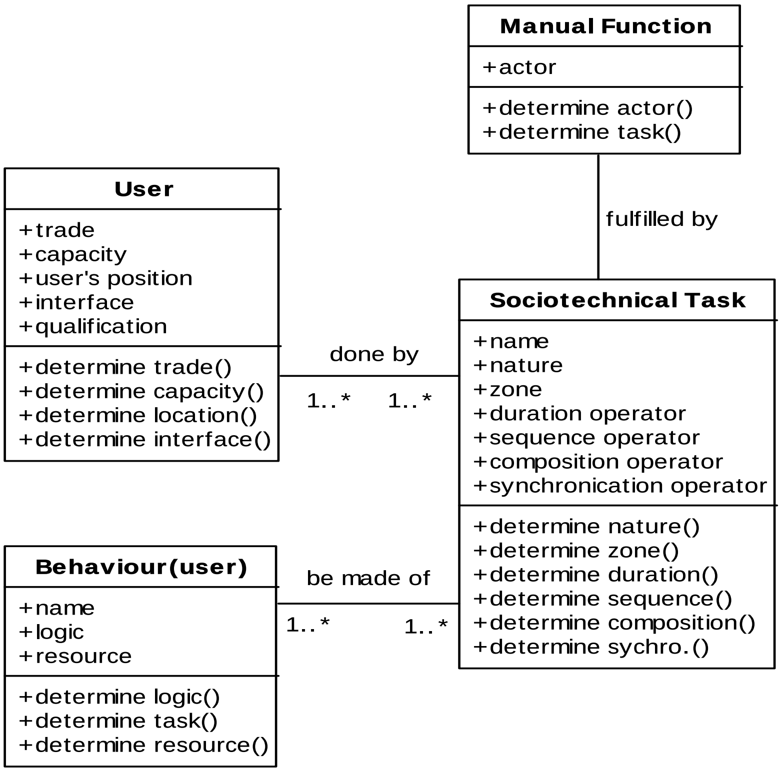


Fig 3.9 The relationships between task, manual function and user's behaviou

The user task represents the tasks requested from the user's viewpoint to fulfill the manual functions, which could not be automated. These tasks can be carried out by one or more users (the correlative working team). In the case of a working team, tasks are performed in an intervention mode (manual mode, maintenance mode, setting-up mode, repairing mode, etc.). The User Task in the design process has to be defined earlier. Five major principles have to be aware of as follows:

- ① the task can be carried out by people with diverse abilities;
- ② the task accommodates a wide range of individual preferences and abilities;
- ③ the task must be easy to understand regardless of user's knowledge and language skills;
- ④ the task must be communicated effectively to the user;
- ⑤ the task can minimize the hazards and the adverse of accidental or unintended actions.

3.3.4 The global view of Task Model

Task is one of the knowledge-based tools in the behavioural design. We use a tier of levels to explain the relationship between behaviour and task. The highest level is behaviour, which is composed of a concrete task, and the concrete task is composed of sub-task, and the sub-task is composed of the elementary task. Behaviour is the concrete, harmonious and aggregative task. For example, the mobile telephone integrates the notepad, daily record, calendar, recent call record, message and camera. And as a result of these functions, mobile telephones include lots of communicative behaviour. This product integrates some concrete tasks: finding a number, dialing a number, communication, selecting notepad, checking the daily record or calendar, and exchanging photos, texts, and e-mail.

We present our contribution and the conceptual foundations and structure of the task model exemplified in Figure 3.10. In this figure, we present details pertaining only to common concepts; other concepts will be detailed in the following. This model supports most of the parameters linked to the environment and use parameters. Identified concepts presented in this model are the results of analyzing real situations (Houssin, Bernard et al. 2006). In this model, we gather the parameters of use conditions from a socio-technical viewpoint (cognitive, social, organizational, etc.). To make the integration of our behavioural design approach easier, we used some concepts already used in FA.

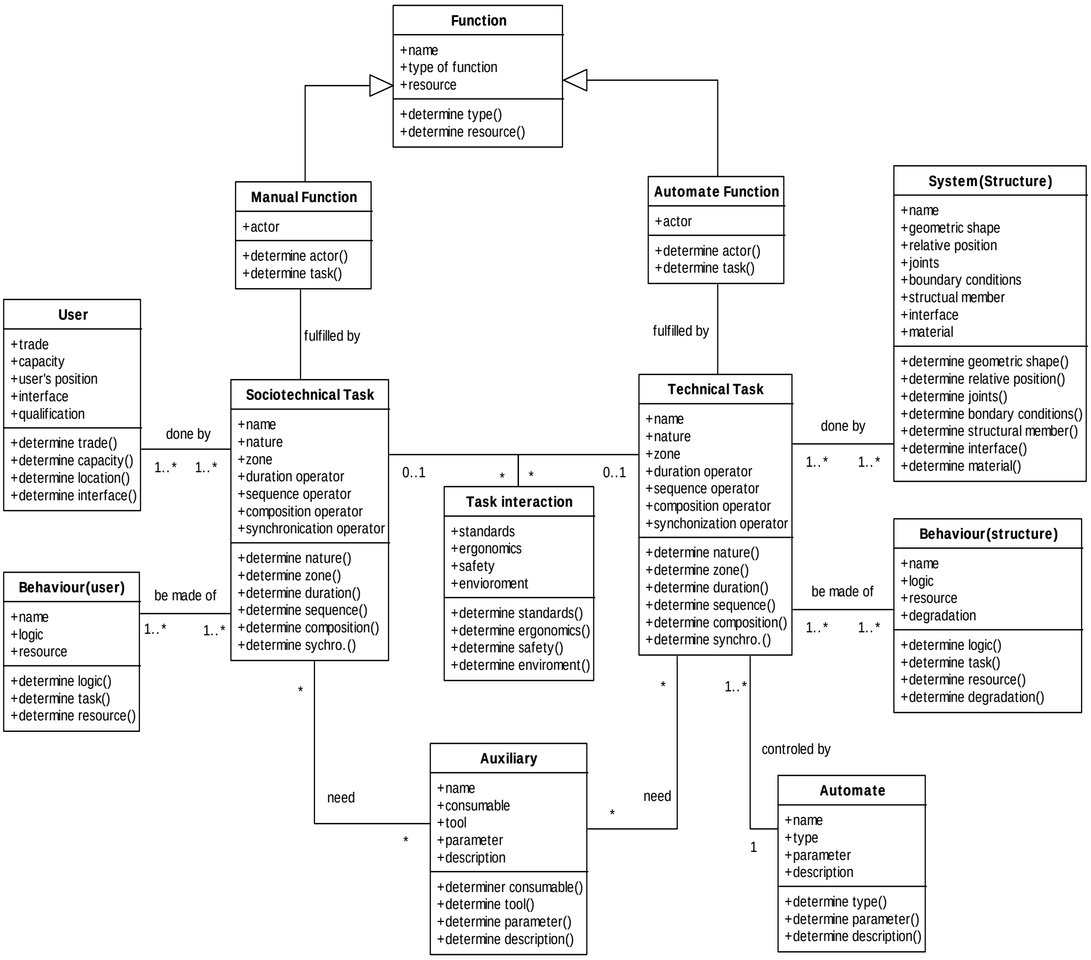


Fig 3.10 The global view of the Task model

(1) "System" Concept

The "System" concept represents the structure of the product. It is presented as a system to illustrate the possibility of its decomposition. It is characterized by its inputs (sensors, buttons, etc.) and outputs (actuators, cylinders, lights, motor, etc.). This characterization should be done in close contact with engineers, electricians, mechanics, etc. (Prouvost 2004).

We adopted the proposition made by (Hasan, Bernard et al. 2003), the characterized concept has the following attributes:

- Name, variation, version number, start date, end date and interface,
- Their relationships to service function, task, work team, etc.
- Inputs: this attribute determines the admission system (sensors, buttons, etc.).
- Outputs: this attribute determines system output (actuators, cylinders, lights, motors).

(2) "Function" Concept

This concept is different from the classical "function" defined in FA. It groups the technical and socio-technical functions, such as manual functions, that are very expensive to automate, and are fulfilled by manual tasks performed by users. Each function can be split into sub-functions depending on the operating mode (normal operating mode, stop mechanism, automatic control, degraded operating mode, etc.). It is characterized by its name, type, and resources.

(3) "User" Concept

The "user" concept characterizes the system user according to profession, experience, expertise, gender, age, etc. The "Work Team" concept is defined by the number of users, the relationships between them, and their relationships with the hierarchy and their cooperation. This last concept helps to allocate the required number of users to each task. These aspects are highly focused in this paper.

(4) "Auxiliary" Concepts

Other concepts required include the:

- "Consumable" concept, which represents all consumable materials used in the work situation, and that are necessary for completing a task or operating the system correctly (raw materials, cutting blade, cord, etc.).
- "Tool" concept: A tool is a hand-held instrument used to shape the material and/or carry out a task (screwdriver, ruler, stroboscope, pliers, etc.). It could be required to achieve a task.

3.3.5 Task Plans (TP)

From the perspective of safety, accessibility, usability, ergonomics, user's behaviour is a subject to study the interactive relationship and function among user-machine-environment. It is the synthesis of the thinking, methodology and theory of user-machine engineering (Henderson and Bhatti 2001). Functional allocation and decomposition between user and machine, user machine interface, working space, and information transmission are defined as its research object. For these reasons, Task Plans framework is proposed as a useful tool to help designers to determine the task.

It is adapted from the concept developed by Houkes and Vermaas as part of their function theory (Houkes, Vermaas et al. 2002), as shown in Figure 3.11.

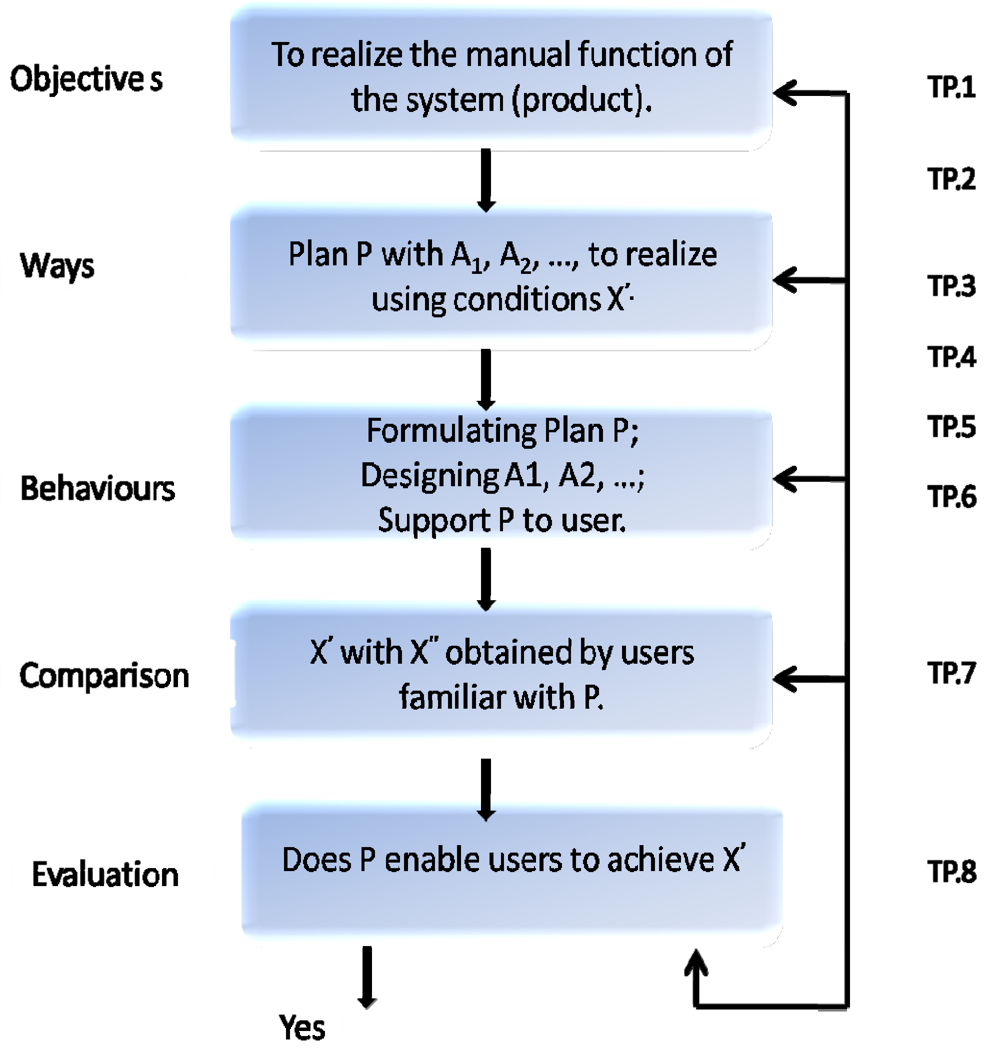


Fig 3.11 Task Plans

- TP.1 According to requirement specification, functional analysis and some known methods, the designer can arrive at the expected function of the product. The expected function contains manual function and technical function as discussed above. The Task Plans focus on manual function.
- TP.2 The designer wants to bring about his objective (manual function) of leading to using situation X. (X means the situation of the product)
- TP.3 The designer believes that X′, satisfying the manual function, is the closest consistent and feasible approximation of X. The designer intends to contribute to giving rise to X′ (from TP1 and TP2 by the characterization of the objective).
- TP.4 The designer believes that a intended user who is following an suitable task plans P that includes: 1) the artifact A_1 with functions f_{11}, f_{12}, …, satisfying manual function F_1; 2) the artifact A_2 with functions f_{21}, f_{22}, …,

satisfying manual function F_2, etc., will lead to X′ (the artifact means the tools used by the user or some sub-structure of the product to fulfill the manual function).

- TP.5 The designer plans to formulate task plans P and to suppose it to the intended users (from TP3 and TP4 by practical reasoning (Sandis and MyiLibrary 2009)).
- TP.6 The designer checks whether the resulting designs of A_1, A_2, etc. are consistent with P, and returns to TP4 or TP5 if this is not the case.
- TP.7 The designer believes that X′ can or cannot be given rise to by intended users to whom P is supposed. This viewpoint is based on the assumption that some of these users go through a series of P′ and give rise to X′′, and on a comparison of X′′ with X′.
- TP.8 The designer arbitrate that his objective (manual function) to bring out X′ has been achieved or not. In the following stage, he can decide to repeat the entire design cycle, settle on another plan (return to TP.4), or repeat at least one design cycle (return to TP. 6).

In this Task Plans, the designer's choice of an alternative using condition X′ may be equal to X. The using condition (X′) is to realize the manual function F which may not be contained in the client's objective; meanwhile other issues or situations must to be taken into account, such as government regulations and the designer's wish for safe and serviceable system (products). And the experience of designers is an important reason for his choice of artifacts A_1, A_2, etc. At the same time, the designer's past experience is clearly an important reason for his choice of artifactsA_1, A_2, etc. In some cases, the choice of X′ may even be dictated by the artifacts that the designer has experience with or easy access to.

A simple explanation of this Task Plans is that the designer has definite confidence in the function of realizing by certain structure S. For instance, if the artifact has the function to firm the utility cutter, the designer may choose the handle with the material of specific shapes, sizes and weights. Another explanation is that the designer considers that the description of some already existing artifact A′ with function f_1, f_2, etc., will be appropriate to designing A. For example, if A has the function of cutting, designers usually choose from the existing tools. But, in general, TP.4 will be much more difficult and complicated. Designers typically decompose into sub-functions the functions of A and try to design structures with those sub-functions. In this way, product design becomes recursive.

Let us assume that a behaviour transforms, in general, using condition into another. The mapping step can then be analyzed as consisting of two sub-steps. The first is that the designer develops the Use Plans aiming at the purposes by defining the behaviour b_i for which the following holds: the first behaviour b_1 transforms an initial using condition (x_0) into a using condition (x_1,) the second behaviour b_2 transforms this new condition (x_1) into using condition (x_2,) and so on, and the last behaviour b_n transforms using condition (x_{n-1}) into the using condition (x) associated with the purposes. These behaviours including in which ones the artifact to be designed is manipulated. Secondly, the designer determines the physical dispositions the artifact can have in order to let the behaviours with the artifact be effective. For instance if a_i is a behaviour in which the artifact is manipulated, then the designer determines which physical dispositions the artifact can have in order that a_i indeed transforms using condition (x_{i-1}) into (x_i).

Briefly summarized a plan, is an ordering of actions considered by an agent for achieving a purpose. Here, the Task Plan for an artifact is defined as a plan for achieving the purpose associated with the artifact. It contains at least one considered action that involves the manipulation of the artifact. We take the door design as an example, the major function is open the door. The Task plans for the door can consist of the following considered actions: tackle the handle; turn the handle until unlock the lock; push or pull the door until the enough passing space; loosen the handle. We exaggerated this example a bit in order to emphasize that this task plan-description of how users can achieve the purposes of artifacts in a rational reconstruction. It is not meant to be a description of what a user actually thinks or deliberates about when using an artifact such as the door; it rather is a description one ends up with if one starts with the more realistic and shorter phase “the user opens the door”. And then adds all those intentions the user should have had in order to turn the description into one that becomes logically and acceptable.

3.4 Behavioural design approach integration tools

The behavioural design approach allows designers to better understand how users will use the system, what the critical tasks are, which are most frequent, their importance, duration and degree of difficulty. Then the designer will have a more accurate representation of all work situations during the system’s use phase. This approach should prevent risk and performance-reducing situations. These functions are successively gathered, classified and prioritized according to criteria, standards and threshold sets of acceptability that define the expected behavioural performance.

Within our behavioural design approach, FA is completed by behavioural analysis. It is important to notice that while we do not intend to impose any method, we wish to propose an approach that could allow taking into account the behavioural analysis, as shown in Figure 3.12.

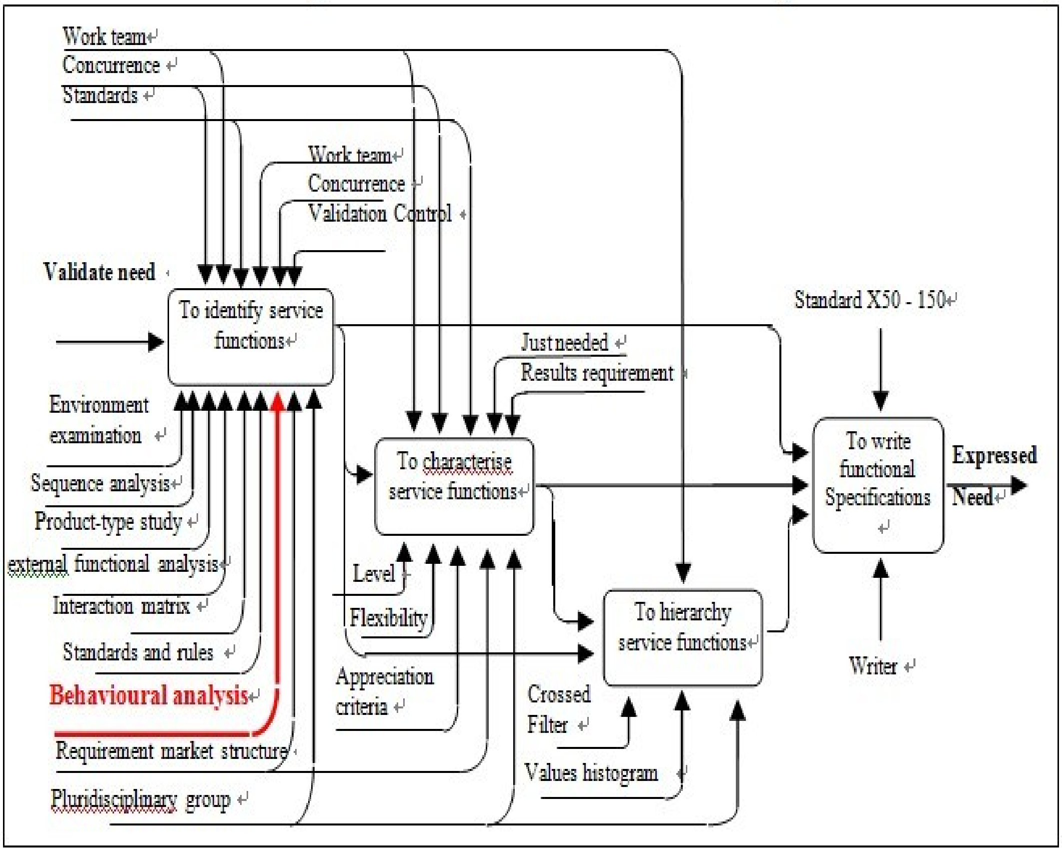


Fig 3.12 Behavioural analysis process

First, the designer defines functions from the specifications. Some of the functions are fulfilled by technical solutions which are well studied in literature. Then socio-technical tasks which may be physical, mental or sensorial (Figure 3.9) should be analyzed to determine the user's behaviour. In this context, a behavioural analysis has to be made. To achieve this objective, in this section we propose aggregating some existing methods that could supply all the data required to instantiate our proposed model.

From an ergonomic point of view, we retain the MAFERGO method (Reliability and Ergonomic Operational Analysis Method) (Neboit, Fadier et al. 1993), which aims to improve system reliability and safety by seeking to reduce the probability of a dysfunction occurrence in technical systems and/or man-system interaction for existing systems. We will use it in the early design phase to analyze the user's task. This method includes the following steps:

- Functional analysis (system, manufacturing processes) and ergonomic analysis (prescribed task, user's real activity),
- Availability analysis (qualitative evaluation of availability rate),
- Identification and classification of defects, which makes it possible to

establish a hierarchy of technical dysfunctions, and thereafter a series of task maintenance and repair operations,

- Dysfunctional Causal analysis is elaborated in the form of a causal graph of dysfunction scenarios. It incorporates all technical, human and organizational events.

To assess maintainability and safety, we choose an approach (Coulibaly, Houssin et al. 2008) which provides a support for designers to accommodate semantic behaviours via a CAD modelling enriched with semantic behavioural data.

To understand all system operating and failure modes, the FMEA method (Failure Modes and Effects Analysis) is proposed in many references, such as in Stamatis (Stamatis 2003). It allows the analyst to gain greater understanding of the system by forcing him to raise questions about the role of each component (system and user). It is used for existing systems and could also helpful to provide feedback data for future design. Presented in a tabular format, this method is divided into four phases:

- Definition of the system, its functions and components,
- Establishment of component failures and their causes,
- Study of the effects of failure modes,
- Proposal of corrective actions and preventive measures to eliminate and control detected risks.

In addition to the previous methods, our behavioural design approach results in formulating and materializing behavioural specifications during operations (normal use or otherwise, maintenance, setting-up, etc.) while respecting users' health and safety (user behaviour).

3.5 Knowledge based behavioural design modelling

Typically, design is the process of finding the proper design parameters to fulfill the design requirements. Therefore, the relationships between the design requirements and design parameters are essential to any design system. Such relationships can be broadly considered as a mapping route from the design requirements to design components. The process is achieved using the domain mapping method similar to the zig-zag decomposition in axiomatic design (Suh 2001).

In order to realize the global view of behavioural design modelling at the early design stage, the designers require much more knowledge from the knowledge base. Here, we introduce the concept of knowledge bases to help designers to make decisions. Knowledge base deals with design rationale and design history issues, which provide additional information (including inference networks, plans, goals, and justifications) about the engineering objects at the content level.

3.5.1 Knowledge and Knowledge Management (KM)

In the 1990s an in-depth understanding awakened that engineering knowledge has a value in itself for those who design and create new concepts of products (Ullman 1992; Pokojski 2004; Clarkson and Eckert 2005). Designers started to realize that their own knowledge resources and abilities have a direct impact on the creation of new products for the market. It became obvious that those who possess the wider knowledge are faster with bringing out new products.

Knowledge is a broad and abstract notion that has defined the epistemological debate in western philosophy since the classical Greek era (Ackoff 1989). Knowledge can refer to the theoretical or practical understanding of a subject. It can be implicit (as with practical skill or expertise) or explicit (as with the theoretical understanding of a subject); and it can be more or less formal or systematic (Dictionary 2005). In the past few years, however, there has been a growing interest in treating knowledge as a significant organizational resource. Consistent with the interest in organizational knowledge and Knowledge Management (KM), researchers have begun promoting a class of information systems, referred to as Knowledge Management Systems (KMS).

The three levels of refinement to knowledge items are data, information, and knowledge. Data consists of discrete, objective facts about events but nothing about its own importance or relevance; it is raw material for creating information (Ackoff 1989). Information is data that is organized to make it useful for end users who perform tasks and make decisions (Beckman 1999). Knowledge is broader than data and information and requires understanding of information. It is not only contained in information, but also in the relationships among information items, their classification, and metadata (information about information, such as who has created the information) (Zack 1999). Experience and skilled people are applied knowledge.

Polanyi and Sen (Polanyi and Sen 1966) originally categorize knowledge into two types: tacit and explicit. Tacit knowledge is difficult to formalize and communicate (Nonaka and Takeuchi 1995). It is transferred through personal interaction, mental models, technical skills, and experience. However, human strategies can be employed to sharpen explicit knowledge (Kidd 1998). For example, although breaking down a corporate vision into operational business or product goals results in explicit knowledge, human strategies such as the face-to-face meeting is usually adopted for this session in Japanese firms (Nonaka and Takeuchi 1995).

Explicit knowledge is easily formalized and expressed (Nonaka, Umemoto et al. 1996; Nonaka and von Krogh 2009). It can be facilitated by traditional information processing technologies (Liebowitz and Wilcox 1997; Liebowitz 2001)). Typically, system strategy is quite effective for sharing explicit knowledge. Knowledge based systems have been introduced for system strategy (Liao 2002).

Knowledge management does not carry its name accidentally because management normally means that 'something' has to be managed (Wiig, De Hoog et al. 1997). Since Polanyi's discussion of the distinction between explicit and tacit knowledge (Polanyi and Sen 1966), researchers were developed a set of management definitions, concepts, activities, stages, circulations, and procedures, all directed towards dealing with objects in order to describe the framework of knowledge management as the KM methodology. Many research programs on KM have been carried out from different points of view: economics, management, engineering and technology (Liebowitz 1999). Different KM working definitions, paradigms, frameworks, concepts, objects, propositions, perspectives, measurements, impacts, have been described for investigating the question of: What is knowledge management? What are its methods and techniques? What is its value? And what are its functions for supporting individual and organizations in managing their knowledge (Nonaka, Umemoto et al. 1996; Van Heijst, van der Spek et al. 1997; Wiig 1997; Wilkins Bert and de Hoog 1997; Johannessen, Olsen et al. 1999; Liebowitz 1999; Liao 2000; McElroy 2003; Wiig 2004; Rao 2005; Sousa and Hendriks 2006).

Managing knowledge is important because knowledge is one of the most strategic weapons that can lead to sustained increase in profits. It is no surprise that many researchers have investigated enablers for fostering knowledge (Nonaka and Takeuchi 1995; Nonaka and von Krogh 2009). A conceptual framework presents knowledge management as consisting of a repertoire of methods, techniques, and tools with four activities performed sequentially (Wiig, De Hoog et al. 1997).

Nonaka et al. (Nonaka, Umemoto et al. 1996) propose the concept of "the knowledge-creating company" which is a management paradigm for the emerging "knowledge society", and information technology can help implement this concept. Some researchers have investigated issues concerning the definition and measurement of knowledge assets and intellectual capital (Wilkins Bert and de Hoog 1997; Liebowitz and Beckman 1998; Liebowitz 2001). Wiig et al. (Wiig, De Hoog et al. 1997) state that a conceptual framework presents knowledge management as consisting of a repertoire of methods, techniques, and tools with four activities performed sequentially. For strategy, Drew explores how managers might build knowledge management into the strategy process of their firms with a knowledge perspective and established strategy tools (Drew 1999). Furthermore, a systems thinking framework for KM has been developed, providing suggestions for what a general KM framework should include (Rubenstein-Montano, Liebowitz et al. 2001). On the other hand, the organizational impact of KM and its limits on knowledge-based systems are discussed in order to address the issue of how knowledge engineering relates to a perspective of knowledge management (Hendriks and Vriens 1999). These methodologies offer technological frameworks with qualitative research methods and explore their content by broadening the research horizon with different perspectives on KM research issues.

The core competence of a global engineering and manufacturing enterprise increasingly depends on the quality of its knowledge resources and how these resources are used (Pokojski 2004; Davies, Studer et al. 2005). In a global engineering and manufacturing project involving many different systems, knowledge resources are highly redundant within each project stage, in that the same kind of information is required for each of the systems, and the same components may be used in many different systems. Also, knowledge resources are often reused as it flows from one stage of the project to the next, and knowledge developed for one project serves as the basis for later projects having similar requirements. Management of this redundancy through re-use is a major goal for the designer based knowledge management architecture. On the other hand, different kinds of contextual knowledge need to be captured at each stage.

The storing and delivering of knowledge were the basic functions of such systems. It didn't take long and the necessity to cover all engineering domains by those systems became evident. But the acquisition of engineering knowledge is necessary for the designer and company. In the following part, we will introduce the knowledge bases into our behavioural design modelling which will help designers to get the better design solution.

3.5.2 Knowledge bases

The development of the scientific method has made a significant contribution to our understanding of knowledge. The scientific method consists of the collection of data through observation and experimentation, and the formulation and testing of hypotheses (Godfrey-Smith 2003).

Knowledge base (Lehmann and Magidor 2002) is a special kind of database for knowledge management, providing the means for the computerized collection, organization, and retrieval of knowledge. Also collections of data representing related experiences, their results are related to their problems and solutions.

So, in order to help designers to realize the mapping among our behavioural design modelling, we introduce four knowledge bases into our modelling. They are ontology base, principle base, structure base and behaviour base, as shown in Figure 3.13.

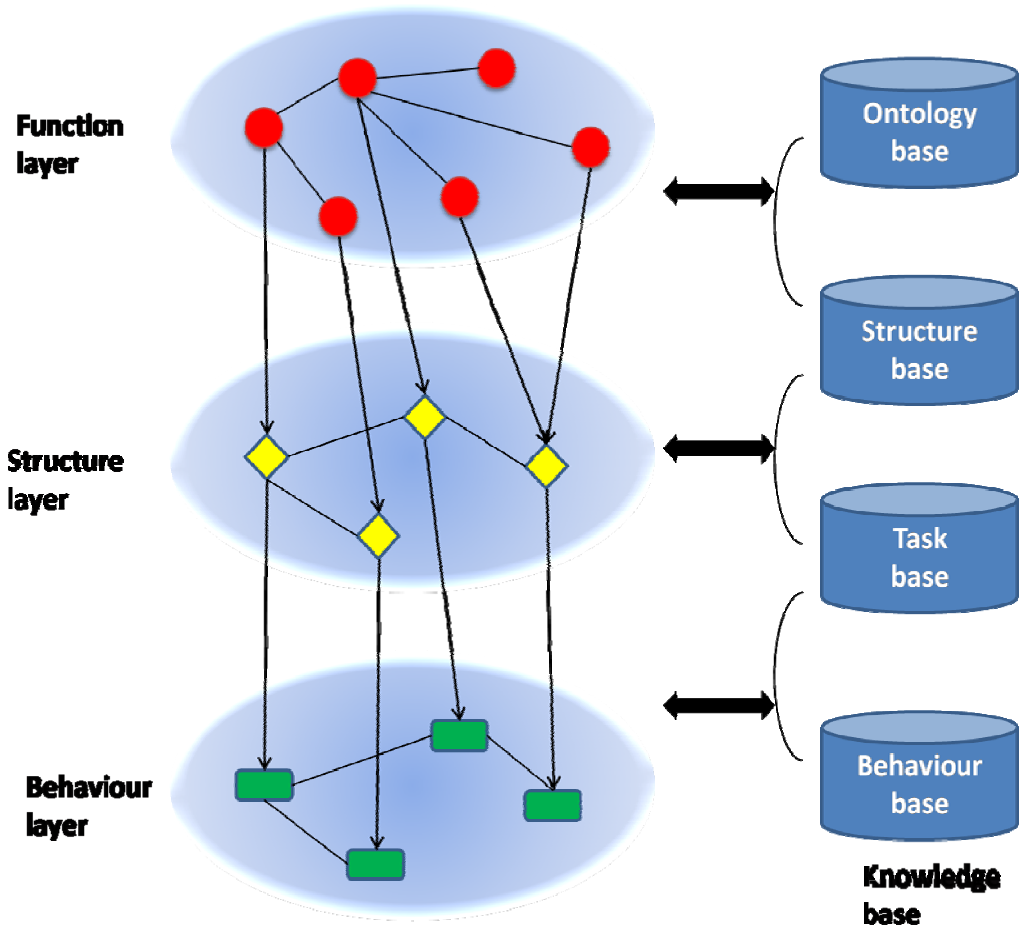


Fig 3.13 Knowledge based behavioural design modelling

(1) Ontology base

Ontology has become a popular research topic and have been investigated by several Artificial Intelligence research communities, including KE, Natural-language Processing and Knowledge Representation (Studer, Decker et al. 2000). Ontology meets a major demand in these fields: they establish a shared and common understanding of a domain that can be communicated across people and computers.

In science, ontology is a formal representation of knowledge as a set of concepts within a domain, and the relationships between those concepts. It is used to reason about the entities within that domain, and may be used to describe the domain (Mizoguchi 2004). In theory, an ontology is a "formal, explicit specification of a shared conceptualization" (Carnap 1956). An ontology provides a shared vocabulary, which can be used to model a domain — that is, the type of objects and/or concepts that exist, and their properties and relations (Gómez-Pérez, Fernández et al. 1996).

The ontological approach pursues the definition of the meaning of terms making use of some kind of logic, and the definition of axioms to enable automatic deduction and reasoning (Sowa 2000). The ontological approach has got a higher relevance since the representation of knowledge is considered the key factor in whatever engineering process, and it has been recognized as a way to facilitate the integration of engineering applications, to describe functional design knowledge, and to define requirements (Ciocoiu, Nau et al. 2001; Breitman and do Prado Leite 2003).

This base is acquired by giving an abstract of knowledge of the sample, consisting of the essence and connotation. It shows the essential contents of design and is a relatively stable side of knowledge. Ontology knowledge corresponds to the design reference that is from the function to the structure, and is a high abstraction degree of design. As with the door design, it contains two major functions: open and close. To fulfill the two functions, it needs their relevant structures and different motions of their corresponding behaviours. It is the stable ontology knowledge of the door design. Another example, the requirement of water pump design can be abstracted to increase energy in the water. The major function of washing machine is the separation of substances.

(2) Structure base

In the early phase of structural design a designer develops and investigates many potential alternatives for safety and economic factors that are to be carried by the structure. The conceptual design of structures wherein the designer investigates many potential alternatives and makes fundamental choices that have the major impact on the downstream decisions is one of the important areas for investigation from the standpoint of automation (Sriram, Maher et al. 1985; Luth, Jain et al. 1991).

Structure is a fundamental, tangible or intangible notion referring to the recognition, observation, nature, and permanence of patterns and relationships of entities. This notion may itself be an object, such as a built structure, or an attribute, such as the structure of society (Pullan 2000). In engineering, a structure is a body or assemblage of bodies in space to form a system capable of supporting loads. Physical structures include man-made and natural arrangements. Buildings, aircraft, soap films, skeletons, anthills, beaver dams and salt domes are all examples of physical structures.

Structure means of dealing with the analysis and design of structures that support or resist loads (Pullan 2000). Structure base is based upon physical laws and empirical knowledge of the structural performance of different landscapes and materials. In this base, we can use a relatively small number of basic structural elements to build up structural systems that can be very complex. This base stores structures of the machinery movement that is used in the mechanical movement system. From the structure base, we can find many types of the door structure which are chosen by the designer. We know there are many types of doors, such as hinged doors, sliding doors, folding doors, rotating doors and so on. We have already had many samples.

⑶ Task base

Dym and Levitt (Dym and Levitt 1991; Dym and Levitt 1991) define task as the actual activity to be carried out, the inputs required and outputs generated and a description of any additional knowledge required to apply the method. Tasks may be defined at an appropriate level to realize a specific process. For example, the tasks required to outline the design a car may include "design motor". And the description of the detail design of the motor bearing may include a lot of analysis tasks specific to a particular bearing technology.

A series of tasks specific to a particular design process is an inherently complex process which is driven by the need to find the most economic and efficient way to the product definition. Design Structure Matrices is such a progress to model task dependencies as a means to order design process (Eppinger, Whitney et al. 1994). This approach allows serial, parallel and iterative design flows to be identified.

The parameters of the task are regarded as a collection of design tasks characterized by their required input parameters. These in turn provide an indication of the context of the use of the task. Parameters may be quantifiable, such as geometry, weight and cost; or non-quantifiable, such as customer satisfaction or ergonomic performance.

To assist such knowledge capture a graphical model is proposed to represent both knowledge about the task, i.e. task knowledge and a part of its associated parameter knowledge. The view just as Debenham states, the goal of the knowledge acquisition stage is to construct a complete, consistent and correct model of the application which is comprehensible to the domain expert and which is in sufficiently precise form to enable a trained person to translate it unambiguously into some implementable formalism (Debenham 1989). In this context, design parameters are those that define the product's physical structure, such as its geometry and the materials used. The task definition itself takes the form of a translation or a test. A translation involves the creation and/or modification of design or performance parameters, while a test is the evaluation of the results of such a translation (McMahon, Meng et al. 1995).

The task knowledge is the analysis of how a task is accomplished, including a detailed description of both manual and mental activities, task and element durations, task frequency, task allocation, task complexity, environmental conditions, necessary clothing and equipment, and any other unique factors involved in or required for one or more people to perform a given task (Kirwan and Ainsworth 1992). We take into account those tasks to be performed by the end-user of the product to assure the global performance demanded for this product.

⑷ Behaviour base

Behaviour refers to the actions and mannerisms made by organisms, systems (e.g. mechanism), or artificial entities in conjunction with its environment, which includes the other systems or organisms around as well as the physical environment(Bateson, Bateson et al. 1999). It is the response of the system or organism to various stimuli or inputs.

Basically, the behaviour can be distinguished into four types (Kim 2010): physiological reactions and responses (e.g. salivation, increase in the blood pressure); bodily motions (e.g. walking, the motion of a robot); actions involving bodily motions (e.g., going shopping, telephoning your friend); and actions not involving overt bodily motions (e.g. reasoning, calculating).

In mechanical engineering, the term behaviour is used with less diverse meanings. There is a consensus that behaviour plays an important role in a number of design methodologies by allowing designers to connect descriptions of the physical structures of technical artifacts to description of their technical functions (Gero 1990; Umeda, Tomiyama et al. 1995; Deng, Tor et al. 2000; Labrousse 2004). The structural descriptions should furnish a ground for behavioural descriptions, which, in turn, provide the grounds for functional descriptions.

In the domain of mechanical engineering design, the description of functions of a technical artifact seizes how the artifact is related to the uses for which it is designed. However, the important and innovative phase of designing in which designers study solutions is usually considered as a phase in which designers purely pay attention to a functional term. In order to realize the functions, the concept behaviour of technical artifact enters this vital stage. Generally speaking, two approaches to functional analysis can be distinguished in engineering (Chandrasekaran and Josephson 2000; Chandrasekaran 2005): first approach, designers relate function of artifacts to behaviours of artifacts, and then relates these behaviours to structural descriptions of the artifacts; second approach, designers model functions of artifacts in terms of inputs and outputs, and then relate these functions directly to structural descriptions of the artifacts. In the first approach, it grants a pivotal conceptual role to the term behaviour, and suggests a clear ontological relationship: a technical artifact has its physical structure; this structure, in interaction with a physical environment, gives rise to the structure's behaviour; and these behaviours then determine in some way the functions.

Chandrasekaran and Josephson (Chandrasekaran and Josephson 2000) explore the different meanings that designers attach to the term of behaviour and distinguish five of them, which are characterized with the help of the primitive of state variables. These five meanings are listed as follows:

- ①Behaviour as the values of all depicted state variables of the artifacts at a special moment or over an interval.
- ②Behaviour as the values of some output state variables of the artifact at a special moment or over an interval (e.g. the amplifier is behaving well—the output voltage is constant).
- ③Behaviour as the value of some state variable of the artifact over an interval of time.
- ④Behaviour as the value of a property of the artifact or a relation between such values (e.g. a lintel distributes the load to the two sides).
- ⑤behaviour as the value of some state variables of the artifact or a relation between such values at a special moment (e.g. the car rattled when the driver hit the curve)

In our behavioural design modelling, the concept of behaviour contains two aspects: technical aspects and human aspects. On the one hand, the behaviour concept represents the behaviour which is carried out by the system according to the technical viewpoints. On the other hand, it also represents the behaviour which is carried out by the users of the system or the correlative working team according to human factors.

3.6 Conclusion of the chapter

The early stage of the design process is often based on designers' experience; assumptions and irreversible restricting the solution space are taken. It is difficult to take into account simultaneously every requirement (technical and socio-technical) imposed by the different phases of the product life-cycle.

More to the point, at a stage where the knowledge is uncertain, most of existing computer-based tools are based on models requiring the complete geometrical definition of the product. Our method covers multi-trade mechanical system design, and aims to a better integration of system-use conditions into system behaviour, starting from the early design phase. To that end, a global view of the behavioural design approach is proposed as a feasible solution to improve system performance starting from the early design phase, and seven steps are given to describe the approach procedure.

In order to implement our global view of the behavioural design approach at the early design stage, we introduce the task model (to be performed either by the product itself or the user).Moreover, we take into account those tasks to be performed by the end-user of the product to assure the global performance demanded for this product. These tasks take into account the analysis and specification of the using conditions; that is to say, maintainability, user's safety, reliability and ways of system usage. Our approach is based on a “Task model” integrated into the FA. And a model of the task plan (performed by the user) required realizing the mapping of the behavioural design approach is also proposed.

Finally, in order to realize our global view of behavioural design modelling at the early design stage, because of the designers’ requirements of much more knowledge, we introduce the four concepts of knowledge bases (ontology base, structure base, task base and behaviour base) to help designers to make decisions.

In the next section, a software application using UML and Visual Studio is currently in development to support and show the usability of the “behavioural design” approach by integrating it into the daily work of the designer.

Chapter 4 Implementation of Behavioural Design Approach

In the previous chapter 3, we have proposed the global view of Behavioural Design Approach (BDA) integrated with the task model and knowledge bases to realize the mapping of the model. At this step of our research, a combination of the accurate industrial context allows us to define all the factors which are necessary to show and confirm the applicability of our approach. This means that a computer based system for supporting the engineering design based on the proposed approach is indispensable, which is more than a simple database. It is necessary to involve a communicated system that provides information about the dynamics of the design process.

There are two objectives in this chapter, firstly, we show the applicability of the behavioural design model and their information models; secondly, to implant the UML language into a software prototype, we use program Visual Studio and the QT designer to realize the Behavioural Design Approach Software (BDAS). Finally, this chapter illustrates the applicability of the BDAS with some industrial data from a part of the unsophisticated industrial example.

4.1 General introduction of the software prototype

Over the last two decades, many computer-aided design methodologies have been proposed to increase the efficiency and attain better control over different phases of the design and development processes.

Traditionally, a computer-aided design methodology covers technical and organizational aspects of the design process, proving with systems, methods, and procedures to support design routine activities, such as documentation, storage and translation of the design results. The attention of a contemporary of design computer applications is on the later phases of the design process, while the initial phases are still poorly automated and receive little information support (Dietz and Yerazunis 2001; Li, Lu et al. 2005). There are reasons to think that the absence of a design theory, which would coherently explicate the entire design process in a scientific and unambiguous way, is the main predicament for the development of more sophisticated computer tools capable of assisting designers in their nor-routine activities (Cavallucci, Lutz et al. 2000; Schumann, Wendel et al. 2010).

Moreover, there is no computer-aided software that permits combining all these aspects which we discussed above into the design. We have to develop the software based on behavioural design approach which includes the task model and knowledge bases. The software aims to illustrate the practicality, applicability and validity of the behavioural design approach. And it is supported by four distinct knowledge bases which stores all the previously data capitalized by the designer and provides communication between designers and users in order to improve product performance. As we discussed in the preceding sections, a distributed Behavioural Design System for engineering design has been developed and implemented in the operating system of Microsoft Windows by our research group, which is short for "BDAS" prototype.

4.1.1 Directions of the BDAS prototype

We have studied the current problem of existing computer-aided software and engineering design system. We conclude that the existing systems embody a part of the requirements of engineering design through the point of technical view and socio-technical view. The integrated behavioural design approach is applied in our software in order to make up for the limitations of current computer-aided systems. As far as the functional requirements and the system framework are connected, the directions of the BDAS system are listed as follows.

- Directions from technical point of view:
  - The concept architecture (Youngs, Redmond-Pyle et al. 1999) depicts the BDAS system in terms of its key design elements and the relationships among them;
  - The module of interdependent architecture includes two major structures: functional decomposition and layers;
  - The execution architecture depicts the dynamic structure of the BDAS system;
  - The code architecture depicts how the source code and libraries are planned in the development environment.
- Directions from designer's (software user) point of view:
  - A user-machine interface is built for the designer-machine interaction.
  - Private knowledge base is dispersed for storing the privacy of personal knowledge; public knowledge base is dispersed for the common access to all stakeholders;
  - The interface statement is reserved for a communication interface and other software.

For the sake of an examination of the pragmatic and concrete issues associated with the role of architecture in the design and development of systems, we carried out a review of a variety of engineering design software systems to understand architecture. The systems include several images and signal processing systems, operating systems, communication systems, and instrumentation and control systems (Soni, Nord et al. 1995; Kazman, Klein et al. 1999; Pentti and Atte 2002; Gomaa 2006). These systems are listed in table 4.1.

Table 4.1 Summary of various software systems

System	Application Domain	Size	Important System Characteristics
A	user interface	small	window management
B	signal processing	medium	monitoring, real-time
C	image and signal processing	medium	high throughput
D	signal processing	medium	monitoring, real-time, safety-critical
E	image and signal processing	very large	high throughput
F	computing environment	medium	management of distributed information
G	instrumentation and control	large	fault tolerance, multi-processing, safety-critical
H	instrumentation and control	large	multi-processing
I	operating system	large	real-time
J	Communication	very large	multi-processing
K	Communication	very large	distributed, heterogeneous, multi-processing

small: fewer than 100 KLOC; medium: 100-500 KLOC; large: 500-1 MLOC; very large: more than 1MLOC (Geer, Bace et al. 2003)

4.1.2 BDAS software development environment and tools

In this section, the languages, methods and environments used for BDAS software development are listed here.

1 Visual C++

Microsoft Visual C++ (often abbreviated *VC++*) is a free, integrated development environment (IDE) product from Microsoft for the *C, C++* programming languages. As the preferred tool under Windows platform, its stability and usability are undisputed. In order to adapt the requirement of the development library depending on complier, we choose *Visual C++ 2008* as complier and source editor which is used for editing and compiling the source program developed by the *Qt* interface library.

2 Qt-4.7.4 and associated tools

Qt (Qt 2011) is a cross-platform application framework that is widely used for developing application software with a graphical user interface (*GUI*). *Qt* is fully object –oriented, easy to extend, and allows true component programming. Based on the future possibility of running software on a Linux platform, we choose *Qt* as the *GUI* development library.

qmake is a tool that helps simplify the build process for a development project across different platforms. *qmake* automates the generation of *Makefiles* so that only a few lines of information are needed to create each *Makefiles*. *qmake* can be used for any software project, especially for Visual Studio.

3 COM technology

Microsoft COM (Component Object Model) technology in the Microsoft Windows-family of Operating Systems enables software components to communicate(COM 2011).It is used to enable designers to communication and dynamic object creation in a large range of programming languages.

At first, we choose *COM* interface of *QAxObject* of *Qt*, but when *dumpcpp* tool of *Qt* decompiles "*type library: sldworks.tlb*" of *Qt*, it causes many problems of a *COM* interface. Finally, we choose a specific tool of *Visual C++* decompile *type library* to generate interface classes.

4 Databases

There are huge amounts of data to be processed in our program, and most of them are fixed existing. So we use a database as the basis for data storage and retrieval. In view of small scale of our software, we choose embedded database system *SQLite*. *SQLite* is an *ACID*-compliant embedded relational database management system contained in a relatively small C programming library (Owens 2006). In contrast to the other two famous database management system (*Mysql*、*PostgreSQL*) in the free and open source software world, its processing speed is much more faster. Otherwise, *Qt* provides convenient interface class for the development of *SQLite*. Considering the increscent amount of data, our system develops the proxy class for the operation of the database. *Database* is the uniform interface of our software for the operation of the database. In this way, when we add other types of support of the data system, we simply need to modify the *Database* class, no need for modify core logic code of the software. *Database* is equal to the core function of the interlayer between databases, which effectively cut off the effect of the change of the data system on the core function.

5 CAD system platforms

In our project, we choose *SolideWorks* as default *CAD* software platform, which is used to process and browse the 3D model data. *SolidWorks* is a 3D mechanical *CAD* program that runs on *Microsoft Windows(Planchard and Planchard 2010)*, and leads the global 3D *CAD* industry with easy-to-use 3D software that trains and supports the world's engineering and design. *SolidWorks* provides complete 3D solutions so that we can translate our ideas into reality, push the limits of design, and achieve our goals. In our software, it is enough to process and browse 3D model data.

6 Extensible Markup Language (XML)

Extensible Markup Language (XML) is a set of rules for encoding documents in machine-readable form (Bray, Paoli et al. 2000).It is a flexible way to create common information formats and share both the format and the data on the *World Wide Web*, intranets, and elsewhere (Walsh 1998).

Why we choose *XML*, two major reasons are listed as follow:

- In our project, there are lots of data types of 3D model data. Characteristic data corresponding to different types of the model contain various types of data. In order not to interfere too greatly with core functions of our project, and avoiding the effects of a possible new increased model on the existing program function, we need to find one data structure which is flexible. Because of the low flexibility of extendibility of *C/C++*, we choose *XML*.
- We use a *database* in our project. The data structure of the database is relatively fixed. And the processing capacity of the database on diversity structure data is comparatively low. So we need a tool to indirectly improve the adaptability of the database for processing diversity of datum types.

4.2 BDA system framework integrated with the task model and knowledge bases

We introduce the concept of combining both structures (CAD models) and behaviour (user's and structure's) of mechanical engineering design system into our computer system. In order to achieve the composition of the behavioural design approach, we introduce the task model and knowledge bases where systems consist of component objects and interactions between the user and structure. This section first defines the target of the BDA system based on our approach. And secondly, the outline of the BDA system framework is introduced.

4.2.1 Target of the BDA system

Developing the BDA system for mechanical engineering design is a complicated task that includes not only the technical solutions but also user's behaviour related to the system. It helps designers to answer the questions (Who? What? Why? When? Where? How?) (Harris, Davis et al. 1996; Hickey and Davis 2003) to determine all the parameters of the structure task and the user task (zone, duration, sequence) which are defined in Chapter 3. These questions are classified at the task level (user task and structure task) and not simply at the function level as it is the case in the functional analysis.

In order to answer these questions, designers can modify certain decisions concerning the choice of technical solutions and socio-technical solutions (as discussed in Chapter 3). In mechanical engineering design, the end goal is the creation of a product, system or process performs a function or functions to fulfill customer needs. After the definition of the specification of the user requirement, once the Functional Analysis (automatic function) is determined, the designers research for the appropriate technical solutions to fulfill these functions. However, designers neglect the manual functions which must be realized by the user and impacts for the future design. The BDAS enters into the early design process to help designers to classify the manual function and the automatic function, as shown in figure 1.1. And then our software help designer to complete the structure behaviour which is derived from embodiment design (structure analysis). Also BDAS helps designers to analyze the user behaviour which is derived from the manual function. And therefore in the detail design, it helps to specify the materials, the sizes, and the type of motor and so on. All these factors are determined by both the technical solution and the socio-technical solutions which are influenced by the integration of structure behaviour and user behaviour. Finally BDAS allows analyzing the integration of these two types of behaviour to determine the global behaviour in using our working situation. The BDAS can help designers to find the potentially dangerous phenomena and zones before the manufacturing phase which reduce the cost of the redesign and approve the performance.

4.2.2 Outline of the BDA system framework

In Chapter 3, we have discussed that the behavioural design approach can be regarded as the extension and exploitation of user and machine, user and designer, which are characterized by the technical task and socio-technical task from the early design stage. The distributed BDA system for aiding mechanical engineering design should contain the following major functional elements (as shown in figure 4.1): design assignment, system feedback, knowledge selection, agent integration and information dissemination.

Comparing with the conventional engineering design system, the BDA system first carefully thinks out a piece of research linking the user centered and functional engineering design approaches into an integrated package. Designers can input the new user's task through fulfilling various manual tasks. In order to appropriately use the knowledge for supporting the BDA system, it is critical to identify the context of user's task and structure's task and to verify the adaptive usage of reliable knowledge. After identify the context of the tasks, the designers are encouraged to evaluate and comment on the values of the task interaction according to the results of their interaction. In the meanwhile, the task and knowledge usage is automatically recorded in the knowledge base for improving the traceability and trustworthiness of the knowledge elements of the BDA system.

In addition to the above functional requirements of the BDA system, other aided systems are also considered such as the PLM system and CAD system which are compatible with and collaborate with existing information infrastructure.

The BDA system framework is designed to aid the engineering design process according to the behavioural design model. As we have argued above, the information of a task integration model provides a consistent knowledge as a working framework in the BDA system. Because of the lack of the sufficient knowledge and capability of the individual designer to fulfill a whole engineering design project, it is necessary to associate various designers and experienced users in the project. They are organized into a collaborative working team for solving the design targets. Based on the behavioural design model, the BDA system framework is designed by adopting the intelligent agent (Franklin and Graesser 1997; Russell and Norvig 2010) as shown in figure 4.1.

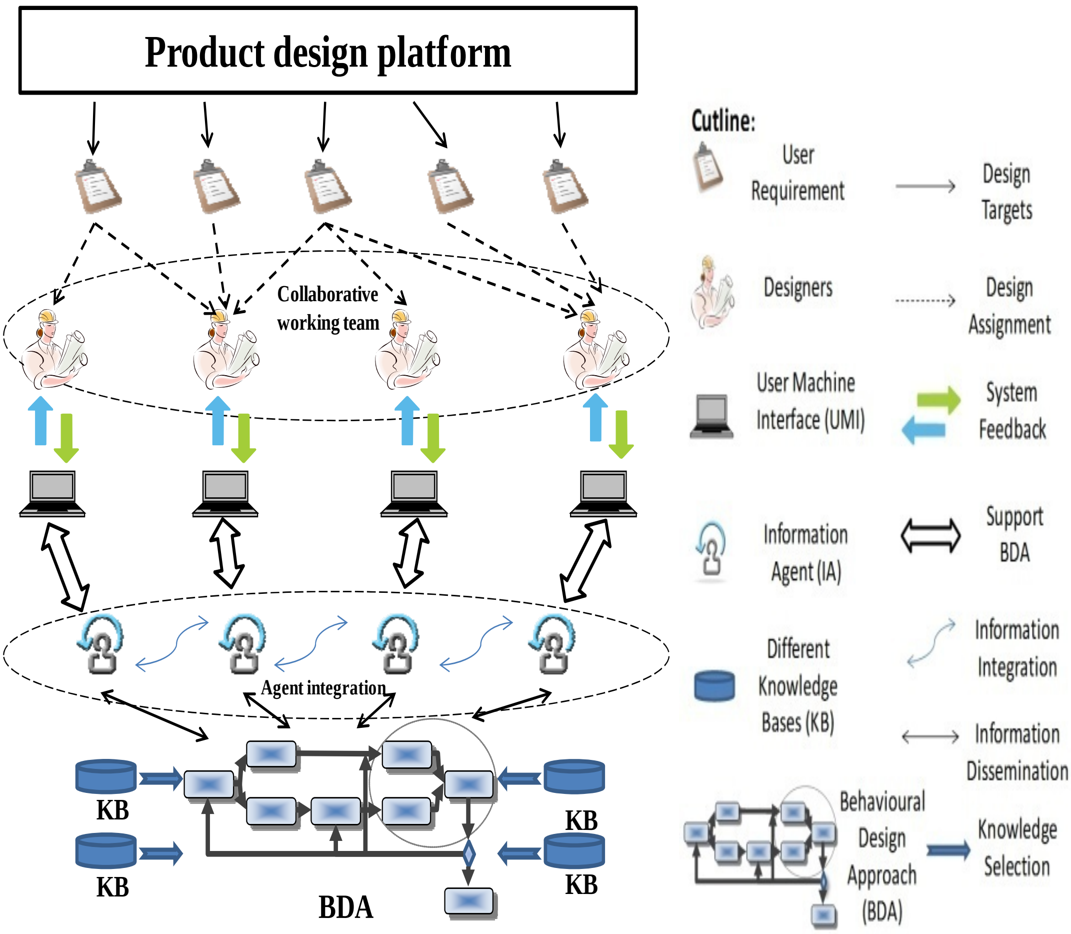


Fig 4.1 Framework of distributed BDA system for engineering design

On the product design platform, the user requirements are analyzed into the design targets that are dispatched to the collaborative working team. The designers collaborate with each other to fulfill the dispatched design targets. Each designer uses knowledge and information of task integration through a user machine interface with the support of the CAD and BDA system. The information agent provides BDA supporting programs such as accessing, searching, compiling, task comparisons and visualizing structures derived from the CAD system. It follows the designer's operations and communications with other agents in the agent integration with respect to the agent requirements. The knowledge base accesses the agent information for storing and iterating relevant knowledge.

The BDA system framework is a comprehensive and collaborative implementation of the behavioural design model, which demonstrates how a collaborative working team can be supported by the behavioural design approach for mechanical engineering design. The collaborative working team and user machine interface construct the user centered layer where using conditions and knowledge creation are performed by the user. The information agent integration embodies in the computer aided layer where supporting the behavioural design approach. The knowledge base stores the iterating information accessed by the agent integration.

In this section, we define the targets of the BDA system framework and its relationships with the existing technical knowledge. After introducing the functional requirement of the system, the BDA system framework is designed based on the integrated approach of BDA for mechanical engineering. In the following, the scenario of the BDA system will be implemented in detail.

4.3 Scenario of the BDA system modelling

For the BDA system realization, system modelling targets to define in detail the basic computational models under the BDA system framework. In order to translate the behavioural design modelling and their information models into computer language, the UML is used to model the overall structure of the BDA system. System modelling consists of use case modelling, class diagram modelling, activity diagram modelling and sequence diagram modelling. Each modelling is presented with a circumstance as follows.

4.3.1 Use case of BDA

To describe how to achieve the different functions of BDA, we use the use case modelling. A use case modelling describes a specific usage of the system by one or more actors. Every sub-system interacts with human or automated actors that use that system for some purposes, and those actors expect that system to behave unpredictable ways (Booch, Rumbaugh et al. 2005). Use case diagrams are fundamental to modelling the behavior of a system, a subsystem, or a class. Each one shows a set of use cases and actors and their relationships(Medvidovic, Rosenblum et al. 2002). Use cases are a software modelling technique that helps developers determine which features to implement and how to gracefully resolve errors (Adolph, Cockburn et al. 2002). A common use of a use case in business models is to use them as the basis for identifying functional and non-functional requirements on one or more information systems (Eriksson and Penker 2000).

A common question is when modelling use cases on a software system is "How do I know I have defined the right use cases in terms of the business?" The BDA process is one of the core business processes of our system. Based on the integrated behavioural design approach, designers perform two major activities to improve product performance, which is the behaviour's integration (behaviour's comparison) and identification of a dangerous phenomenon. The use case of behaviour's integration depends on several sub-cases such as function classification, structure capture, behaviour analysis, behaviour comparison, modification of existed knowledge and creation of new knowledge. The use case of identification of the dangerous phenomenon includes the sub-use cases such as identifying, comparing and evaluating information. In the BDA system framework, a product design project according to the analysis of user requirements is decomposed into several design targets. And they are assigned to different designers in the collaborative design teams. As design members are involved in the product design process, the management of designers and their rights in the BDA system is another business support. Also there are many other detailed business processes which do not explain furthermore.

Use case diagram is to show what system functions are performed for which actor. An actor is a person, organization, or an external system that plays a role in one or more interactions with the system (Santander and Castro 2002). Actors may be described at the abstract level (business actors), or at the system level (system actors) (Heumann 2003). In the BDA system, business actors include information agents, knowledge bases and the CAD tools integrated with the system. Designers belong to the system actors who manipulate the system functions to realize the design targets. A designer can act both as a task or behaviour analyst and the knowledge user or actor according to their manipulated activities. The administrator of the project is also a system actor whose duty is to manage the whole project, assign the design targets, and authorize designer's right in the system.

In order to better comprehend the relationships between the system and its actors, we propose several use case diagrams of the BDA system and identified actors. Figure 4.2 shows a use case diagram of a designer who manipulates the system as both a behaviour analyst and a knowledge user. Using the BDA system, this diagram explains how a designer can realize the behavioural analysis and use knowledge available in the data bases for design targets.

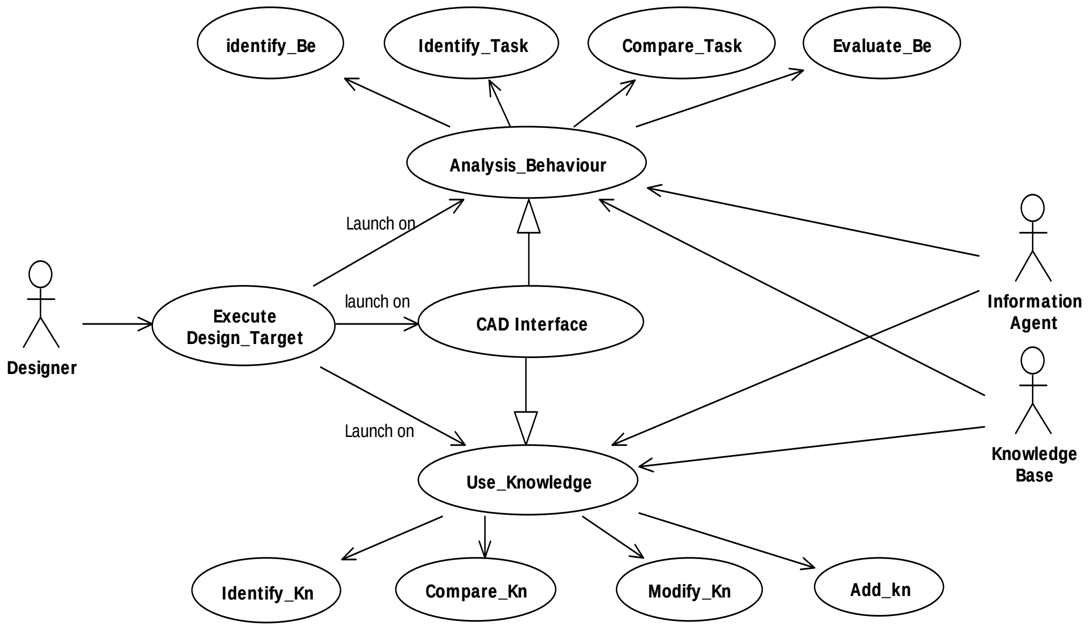


Fig 4.2 Use case diagram for the designer as the behavioural analyst and the knowledge user

In the above figure, three major actors involved in the use cases, which a designer, knowledge user (knowledge base) and the agent (information agent). The relationship between an actor and a use case illustrates the actor's operations on the system described by the use case. In this use case diagram, a designer is related to the use case of *Execute_Design_Target* that will launch on three use cases of *CAD_Interface*, *Analysis_Behaviour* and *Use_Knowledge* according to the specific circumstances of the designer. The *CAD_interface* links the *Use_Knowledge* and *Analysis_Behaviour* to realize the behavioural comparison. The use cases of *Use_Knowledge*, *CAD_Interface* and *Analysis_Behaviour* are made up of more use cases in depth will not be detailed here.

4.3.2 Class diagram of BDA

The internal of our behavioural design model and the BDA system are detailed by the class diagram modelling which illustrates the objects and their relationships. The class diagram describes the structure of the system by showing the system's classes, their attributes, operations (or methods), and the relationships between the classes (Booch, Rumbaugh et al. 2005). It helps us to fully understand the real structure of the BDA system and thus provide the main building block of object oriented modelling for the realization of the BDA system.

BDA system is generally classified into three major classes:

- Economic class: Technical solutions and socio-technical solutions, such as project management, ergonomics and the design process. It involves the product, human and technology used in the mechanical engineering design.
- Organizational class: Framework of the BDA system which contains the CAD system, PLM system, knowledge system and so on.
- Using class: Use case of the BDA system. It illustrates the relationships between the system functions and the actor of the system.

The three classes are constructed by four major packages according to their functions of the BDA system. They are *User_Machine_Interface_Package*, *Behaviour_analysis_Package*, *Agent_Package* and *Database_Package*. The entire class diagram of the BDA system is illustrated in figure 4.3 by including all four packages.

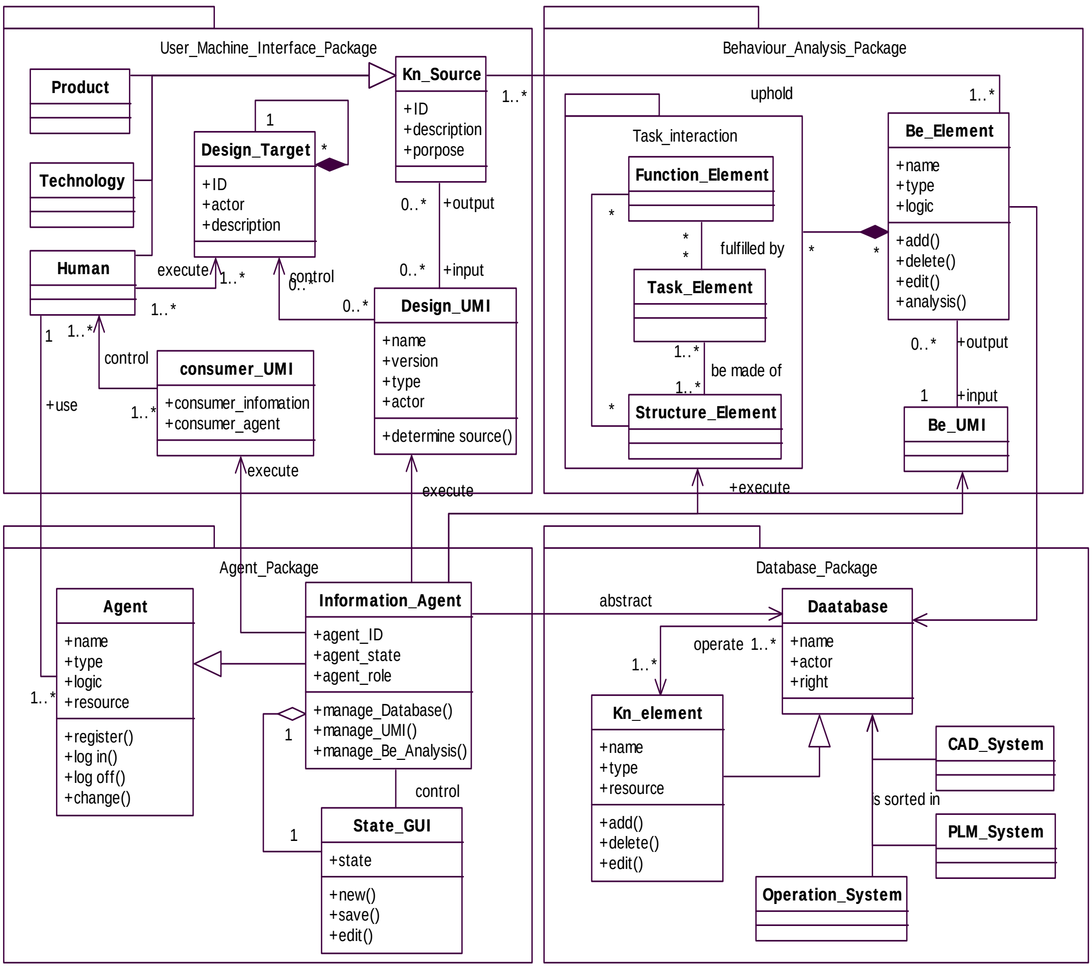


Fig 4.3 The entire class diagram of the BDA system

In Chapter 3, the *Behaviour_analysis_Package* and *Database_Package*have been discussed; the *User_Machine_Interface_Package* and *Agent_Package* will be further explained in the implementation of the BDA system.

4.3.3 Object interactions and process operations of BDA

In order to show object interactions and how processes operate with one another , we use a sequence diagram which shows object interactions arranged in time sequence (Booch, Rumbaugh et al. 2005). Sequence diagram models flow of logic within our BDA system in a visual version, enabling us both to document and validate our logic. It helps us for both analysis and design targets. Figure 4.4 depicts the sequence diagram of *Behaviour_Analysis_Package*.

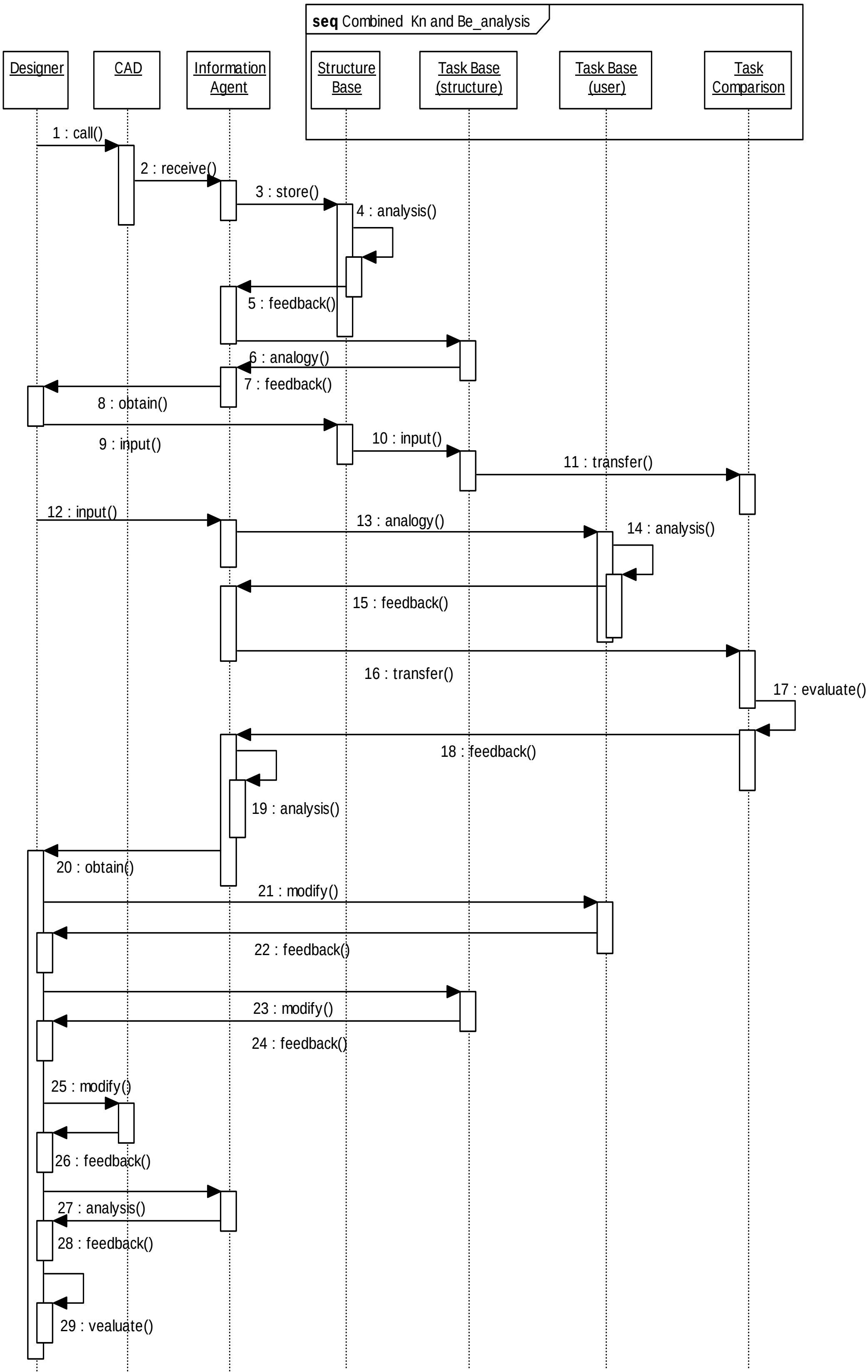


Fig 4.4 Sequence diagram of Behaviour_Analysis_Package

In the figure 4.4, there are seven instances of classes involved in the behavioural analysis process of the BDA system. We list the simple explanation of the sequence diagram of *Behaviour_Analysis_Package* as follows:

1. Call: Designer calls the CAD files.
2. Receive: Information Agent receives the CAD files.
3. Store: structure base stores the product structure derived from the CAD files.
4. Analysis: System analyzes the imported structure and classifies the structures that are already existed in the Structure Base and those who are new ones.
5. Feedback: Information Agent receives the feedback of structure's analysis.
6. Analogy: Make an analogy between Structure Base and technical Task Base (structure).
7. Feedback: Information Agent receives the feedback of results of the analogy.
8. Obtain: Designer receives the results.
9. Input: Designer inputs the new structure into the Structure Base.
10. Input: Designer inputs the new structure task corresponding to the new structure.
11. Transfer: Information Agent transfers the information into the Task Comparison.
12. Input: designer inputs manual functions.
13. Analogy: Make an analogy between Function Base (manual) and socio-technical Task base (user).
14. Analysis: System analyzes the obtained manual task and divides them into new and existed tasks.
15. Feedback: Information Agent receives the feedback of result of task's analysis.
16. Transfer: Information Agent receives the feedback of results of the analogy.
17. Evaluate: The evaluation occurs in the step of the Task Comparison.
18. Feedback: Information Agent receives the feedback of results of the Task Comparison.
19. Analysis: System analyzes the results.
20. Obtain: Designer receives the results.
21. Modify: If the result of 17 is not good, the designer changes the parameters of the Task Base (user). (evaluate again)
22. Feedback: Designer receives the results.
23. Modify: If the result of 19 is not good, the designer changes the parameters of Task Base (structure). (evaluate again)
24. Feedback: Designer receives the result.
25. Modify: If the result of 21 is not good, the designer changes the parameters of

CAD files. (evaluate again)

26. —29. Designer repeats the evaluation step.

Behaviour_Analysis_Package holds core importance for behavioural engineering design, which is fulfilled by designers with the help of Knowledge bases and Information Agent. In this figure, different targets and activities are involved in this sequence diagram, and the flows of the event among them are presented in it. Further sequence diagrams will introduce in detail in the execution of the BDA system.

4.3.4 Dynamic behaviours modelling of BDA

In order to describe the dynamic behaviours of specific objects through the activities, states and transitions, we use the activity diagram and the statechart diagram.

Activity diagram and statechart diagram model show the dynamic aspects of the system. Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency (Booch, Rumbaugh et al. 2005). Activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control(Eriksson and Penker 2000). In Chapter 3, we used the activity diagram to explain how to realize the behavioural design approach in *Behaviour_Analysis_Package*, as shown in figure 3.3. Here, we take the consumer login as an example to illustrate the *User_Machine_Interface_Package*, as shown in figure 4.5.

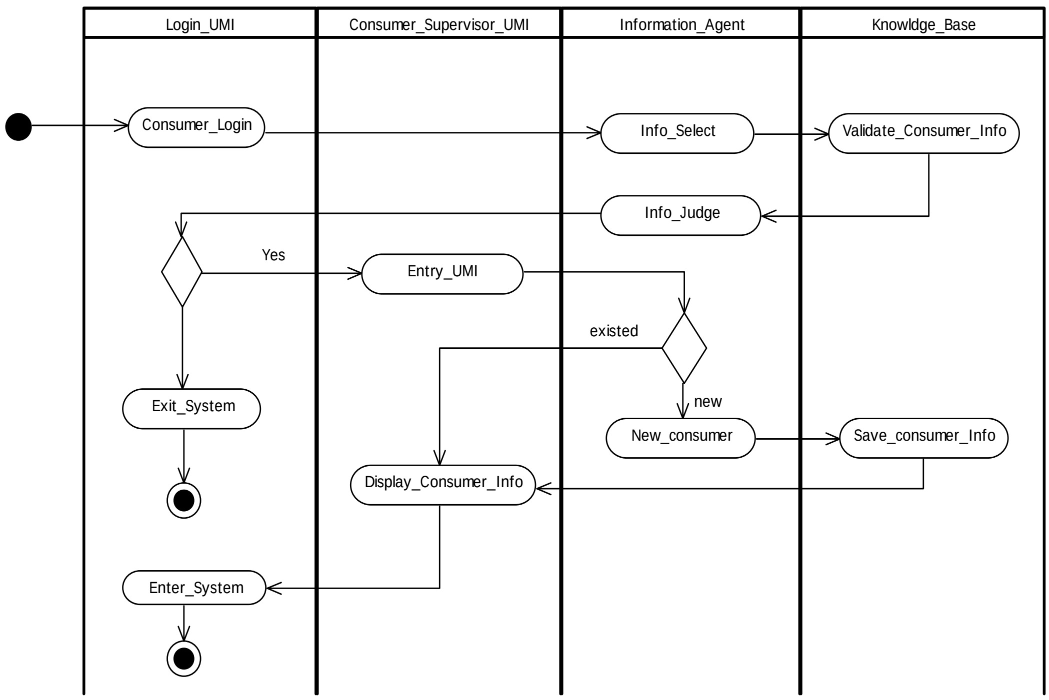


Fig 4.5 Activity diagram of consumer login into the BDA system

In figure 4.5, four main objects are participated in the activities such as *Login_UMI, Consumer_Superviser_UMI, Information_Agent* and *Knowledge_Base*. Each of them plays a different job and responsibility in the BDA system. *Login_UMI* is used for entering and exiting the BDA system. *Consumer_Supervisor_UMI* is used for transferring the information to the *Information_Agent* and displaying the *Consumer information*. *Information_Agent* is in charge of the selection and judgement of information. *Knowledge_Base* stores the new consumer information and provides existed consumer information.

From the figure 4.6, we can also use the statechart diagram to each individual object. It can specify the sequences of states an object goes through during its lifetime in response to events, together with its responses to those events. Here, we use an example to describe the statechart diagram of *Login_UMI*, as shown in figure 4.6.

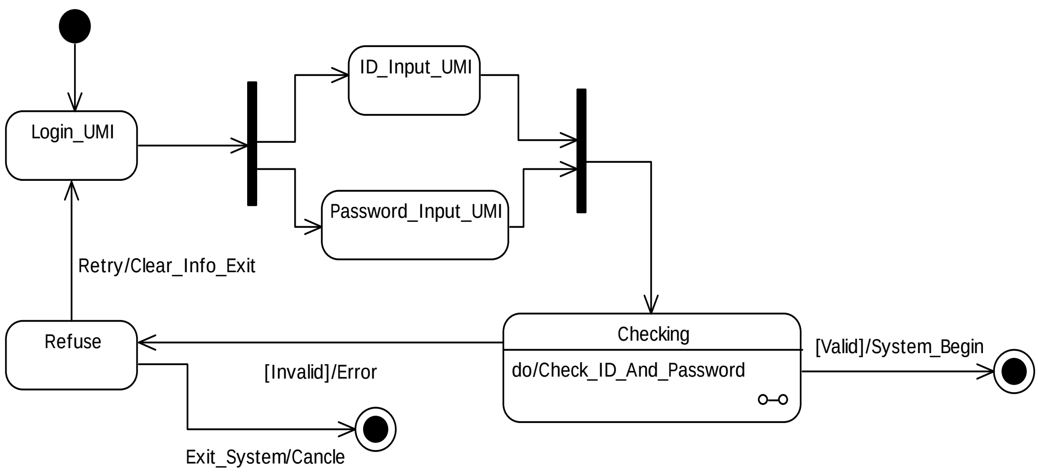
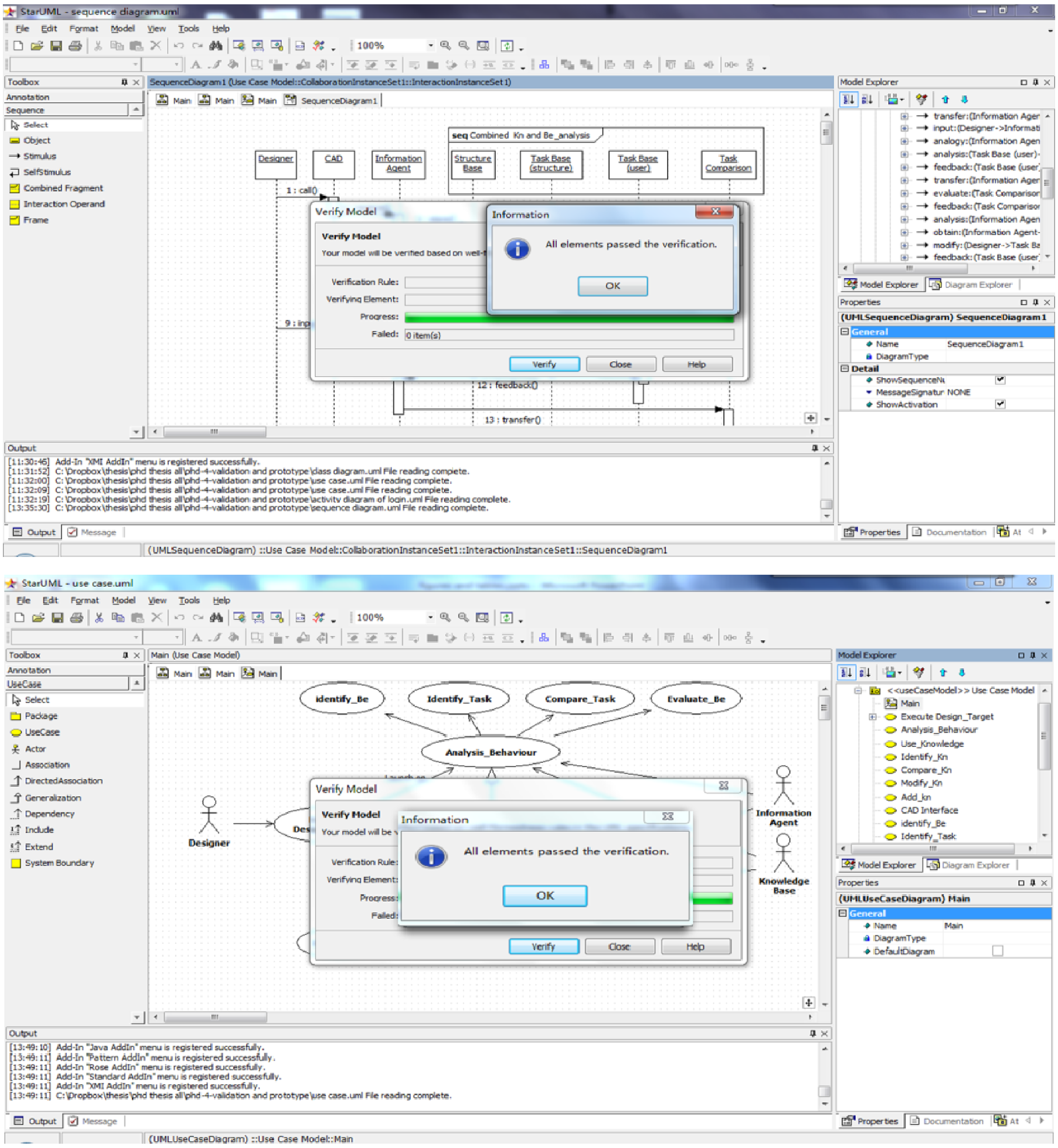


Fig 4.6 Statechart diagram of Login_UMI

In figure 4.6, when the object of *Login_UMI* is initiated, it is in the state of waiting for the ID and Password. And then, it is the state of checking, if ID and Password are right, the system will begin; if ID and Password are wrong, it will go to the state of refuse. After the state of reuse, it can go back to the object of *Login_UMI* or exit system. Because of the limits of time and space, other dynamic aspects of the system will not be introduced here.

4.3.5 Corroboration of dynamic aspects of the BDA system

Use case diagrams, sequence diagrams, activity diagrams and statechart diagrams are four kinds of diagrams used in the UML for modelling the dynamic aspects of systems. These dynamic aspects show collaborations between objects and changes to the internal states of objects. We have modelled the dynamic aspects of the BDA system with different types of UML diagrams. Before the implementation of our software, it is vital to corroborate these dynamic models and static models. In order to realize the corroborations, *StarUML™* (StarUML 2005) which is a Software Modelling Platform is used. All the UML diagrams in this paper are constructed by this software. The corroborations of four major diagrams which are explained in this paper are illustrated in figure 4.7.



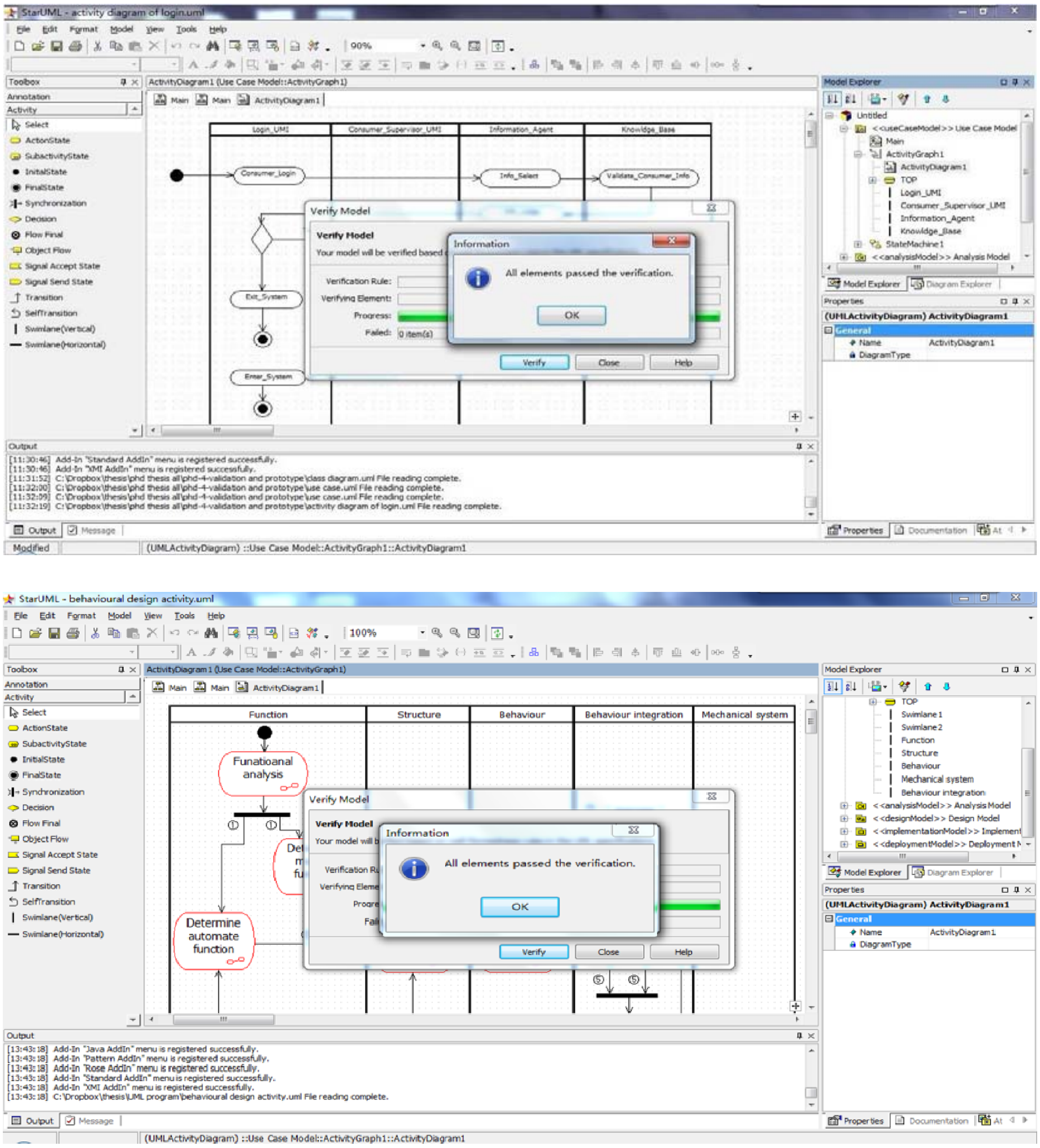


Fig 4.7 Corroboration of the dynamic aspect of the BDA system

Figure 4.7 shows the corroboration of Use case diagrams (figure 4.2), sequence diagrams (figure 4.4) and activity diagrams (figure 3.3 and figure 4.5) that are relied on the function in the *StarUML*™. According to the corroboration of UML diagrams, it verifies the coherence of our modelling for later programming stage. It provides a variety of customizing variables to meet the requirements of the user environment can ensure high productivity and quality .It is efficient to integrate with other tools in the future. It also must provide a high level of extensibility, and allow integration with existing tools or user's legacy tools. Based on the successful corroboration of dynamic aspects of the BDA system, the BDA software is implemented in the following section.

4.4 Software prototype implementation (BDAS)

According to the analysis of directions of the system and its development environment and tools, BDAS (Behavioural Design Approach Software) is initially developed in a straightforward way, where the client and administrator are installed at the same machine. Interfaces of BDAS are presented in the following.

4.4.1 The detailed manipulation steps of BDAS

This software assists the designer to take into account and to respect standards, safety and ergonomics legislations. At the first step, designer calls the structure (structural prototype) from the *CAD* (for example: *Solidworks*) system. The design cycle can be shortened a lot through the structural prototype. A structural prototype enables the designers to check whether the design specifications are matched by performing simulations rather than physical experiments; a physical prototype is only needed for the final examination. Not only does the structural prototype make design certification faster and less expensive, it provides the designer with direct feedback on design decisions. Structural prototypes need to model the behaviour of the equivalent structure correctly; otherwise, the predicted behaviour does not fulfil the actual behaviour resulting in poor design decisions. But creating accurate models is a challenging problem.

Our approach is based on the description of a design prototype for its function, structure, and behaviour (Sun, Houssin et al. 2010).The structure is a description of the physical embodiment of an artefact, while the function is the purpose of the artefact, and the behaviour that the designer intended to achieve. As illustrated in figure 4.9, the structure behaviour does not depend on the function, but simply on the structure.

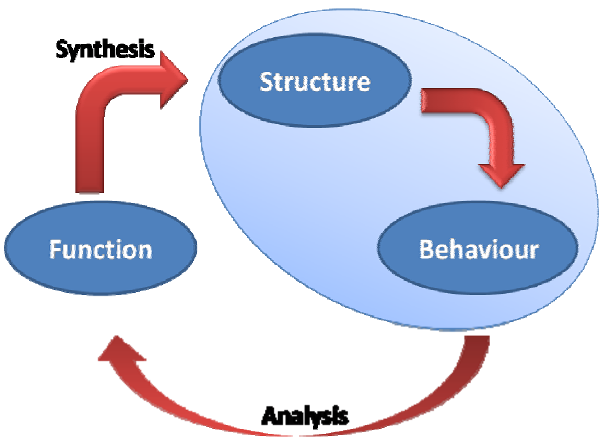


Fig 4.9 Encapsulation of structure and behaviour in a composition

During design or synthesis, we transform the function into the structure, while, during design certification, we obtain the behaviour from the structure and verify whether this behaviour matches the function.

As shown in figure 4.10, we open a *SLDASM* from the *SolidWorks* library and then use BDAS calls it.

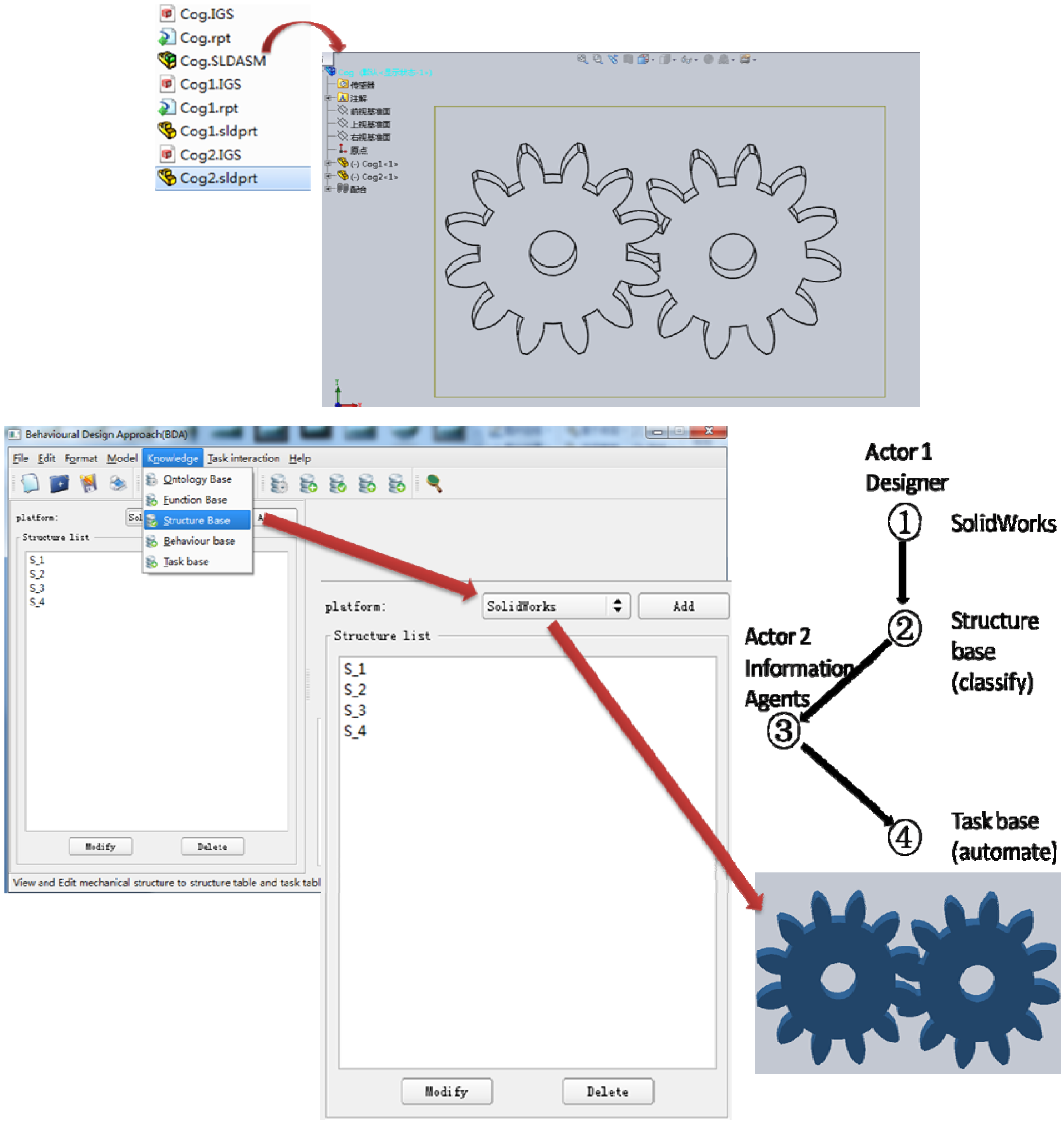


Fig 4.10 Analysis of the structure derived from the Solidworks (structure task)

And then, the Information Agents receive the CAD files and then transfer the information into structure base; BDA system analyzes obtained structure and classifies them into existed structures and new structures; Information Agents receive the feedback of structure's analysis and make an analogy between *Structure Base* and *Technical Task Base*(structure); if there is no tasks match the new structures, designers input the new tasks with all needed attributes which are corresponding to the new structures, as shown in figure 4.11.

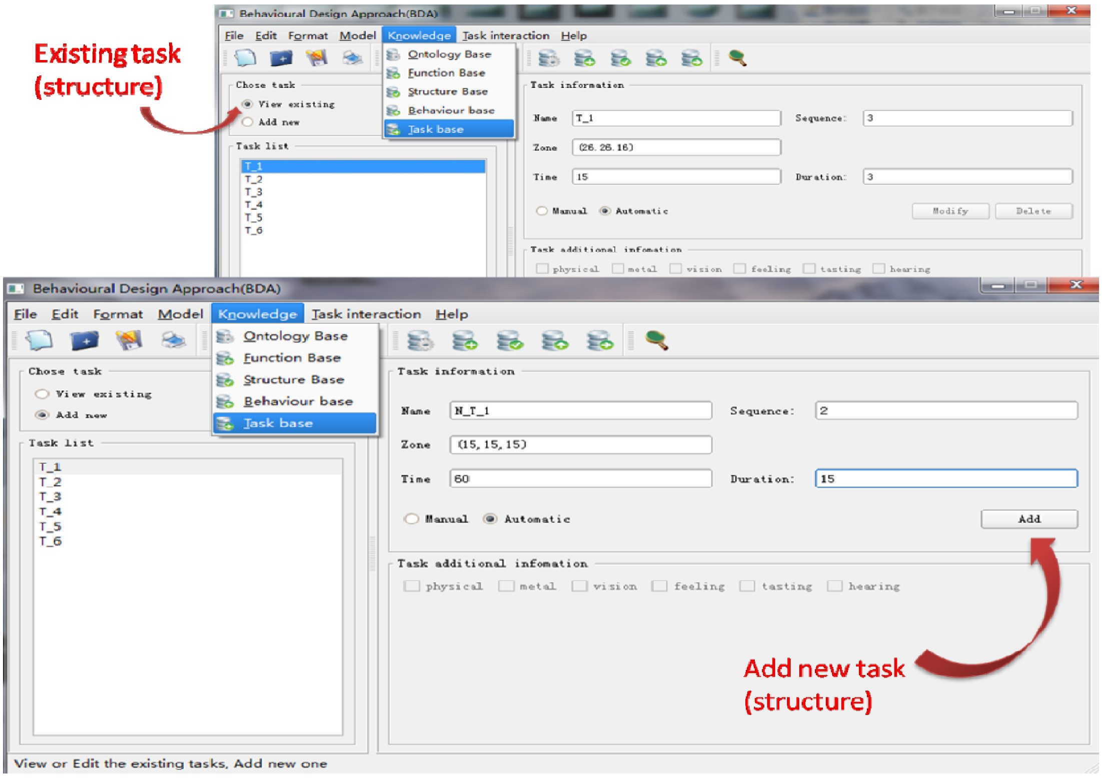


Fig 4.11 Classification and analysis of the structure task

According to our behavioural design approach, engineering design is intended to influence or result in certain user behaviour. It is an effort to explain plenty of types of systems (products, services, interfaces, environments) that have been strategically designed to influence how people use them. Designers try to search for the sources of risks and potentially dangerous phenomena. In order to realize a function, it is necessary to choose a technical solution to carry out a technical task and a socio-technical solution to carry out the user task. So, in this step, designer inputs manual functions which are derived from the FA (the manual function fulfilled by the user, because of the cost or the difficulties related to automation); Information Agents receive the information and then transfer them into Function Base; BDA system make an analogy between Function Base (manual) and Social Task Base (user); the task are divided into new and existed tasks; as shown in Figure 4.12.

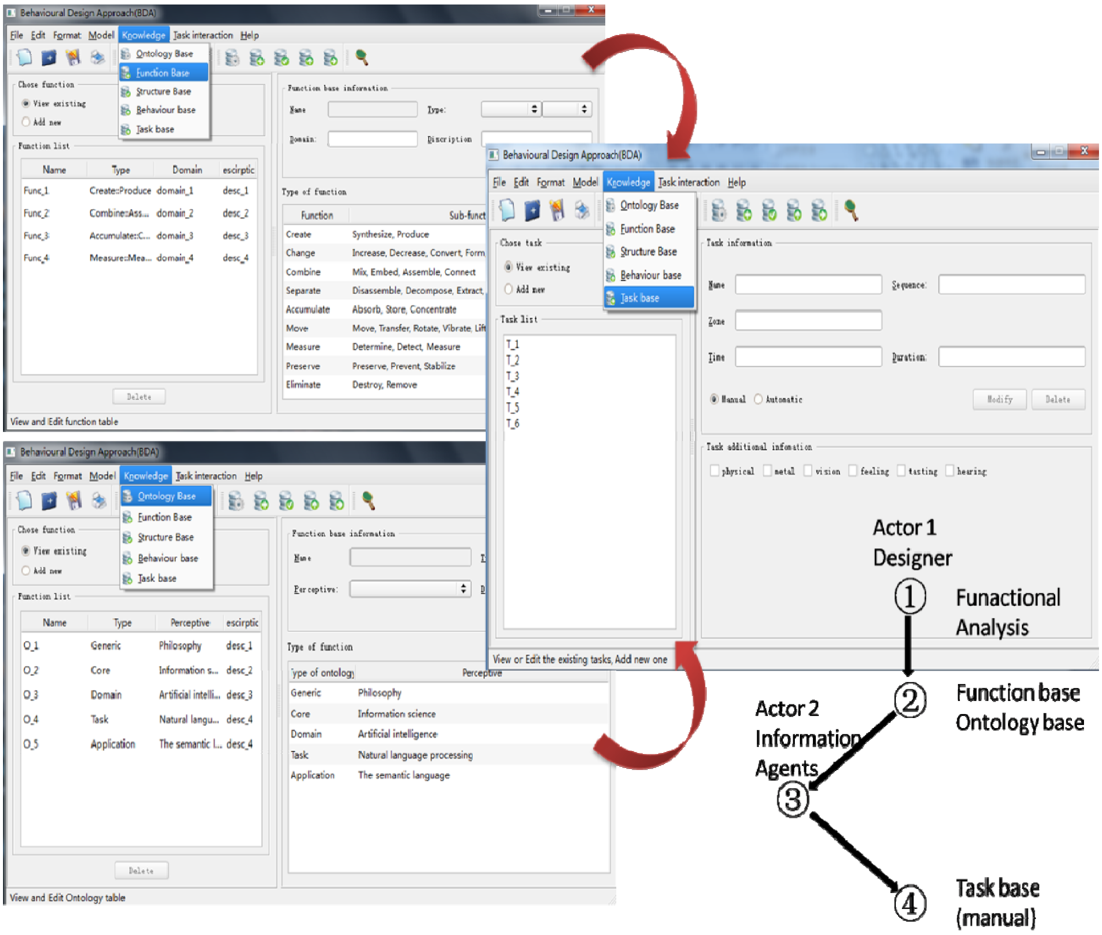


Fig 4.12 Analysis of manual function (user task)

If there is no task matches the new functions, designers input the new tasks with all needed attributes which are corresponding to the new functions, as shown in figure 4.13.

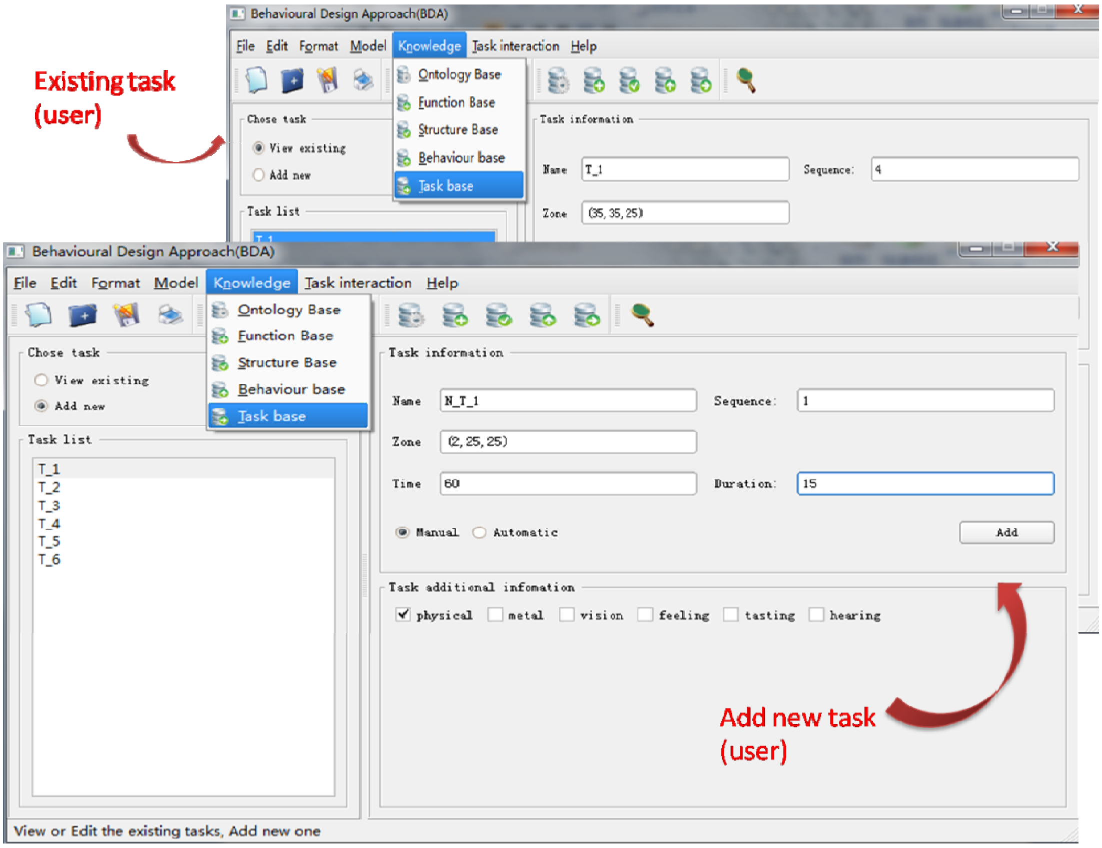


Fig 4.13 Classify and analysis of user tasks

After that, the information agents receive two results: analysis of Structure Base (structure task) and analysis of Function Base (user task); and then the evaluation step occurs in the Task comparison; as shown in figure 4.14.

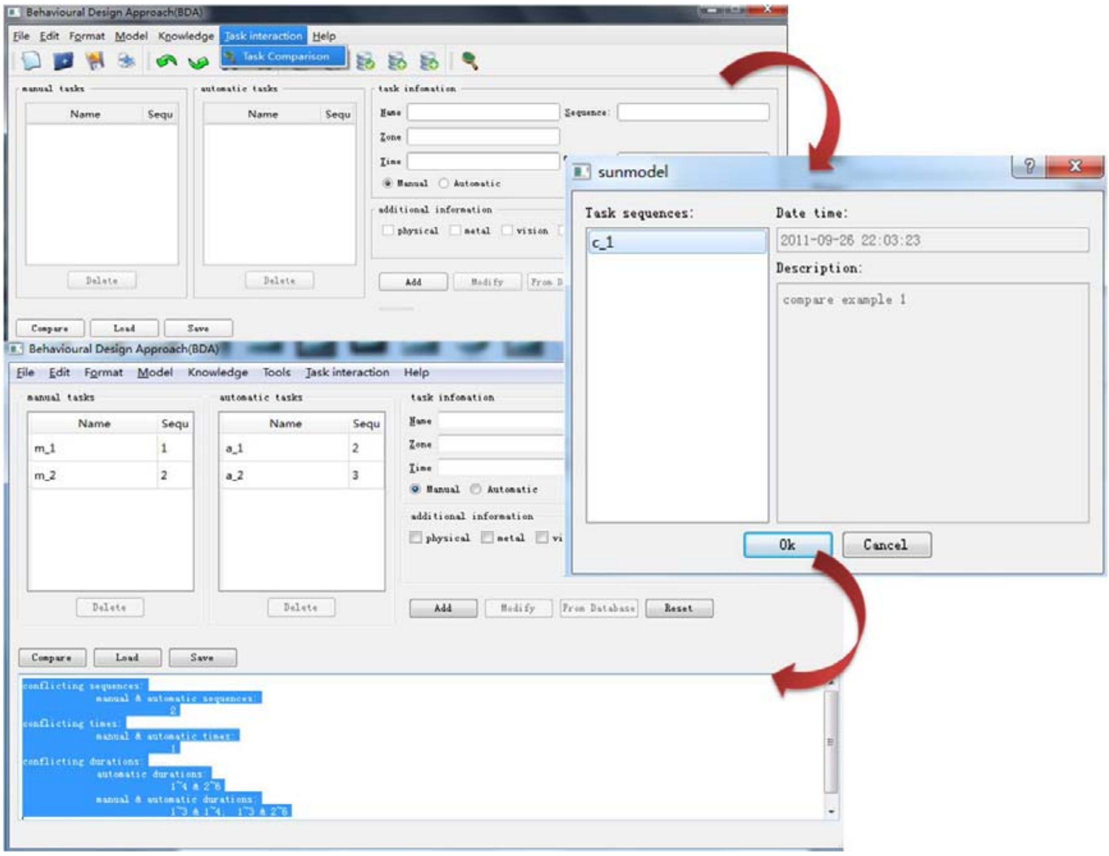


Fig 4.14 The evaluation of Task Comparison

Finally, the Information Agents receive the feedback of result of Task Comparison; if the result is not acceptable, the software will propose designer to modify the solution, task, structure, etc. to cancel all causes that influence product performance (decrease the dangerous phenomenon and engendered hazard; long and difficult tasks; etc.); if all these are not possible, the software proposes the new behavioural analysis; a step of new behavioural analysis to assure the modifications cannot lead any other accessibility or performance problem.

In fact, in the final step of the Task comparison, the result indicates that the solutions contain of generate dangerous phenomena and risks (the movement, electrical energy, radiation, etc.) (Houssin, Bernard et al. 2006; Coulibaly, Houssin et al. 2008; Houssin and Coulibaly 2011). But, their attributes (zone, sequence, time, duration and structure) in the system sometimes allow designers to make some modifications. These modifications must be the most economically possible and could not influence the system function.

This risk estimation is the determination of quantitative or qualitative value of risk related to a concrete situation and a recognized dangerous concerning BDAS. It also allows comparing the various possible solutions. However, part of the difficulty in risk estimation is that estimation of both of the quantities in which risk assessment is concerned - potential loss and the probability of occurrence - can be very difficult to measure (Diggle 1995). The chance of error in measuring these two concepts is large (Edwards, Wiholm et al. 1996). Risk with a huge potential loss and a low probability of occurring is often treated differently from one with a low potential loss and a high likelihood of occurring (Lerche and Glaesser 2006).

Thus, every company defines its proper scale of evaluation for the representative risk of its own domain of industry. It is easy to have a good estimation and assessment of risk for the engineering design. BDAS analyze all the risk parameters which are simulated by the CAD for the engineering design.

4.4.2 Integration of the interfaces into BDAS

BDA organizes the major interfaces and enables to realize the collaborations and communications, which are integrated into a single application. The main frame of BDA is illustrated in figure 4.15, which contains different interfaces for the main functional requirements.

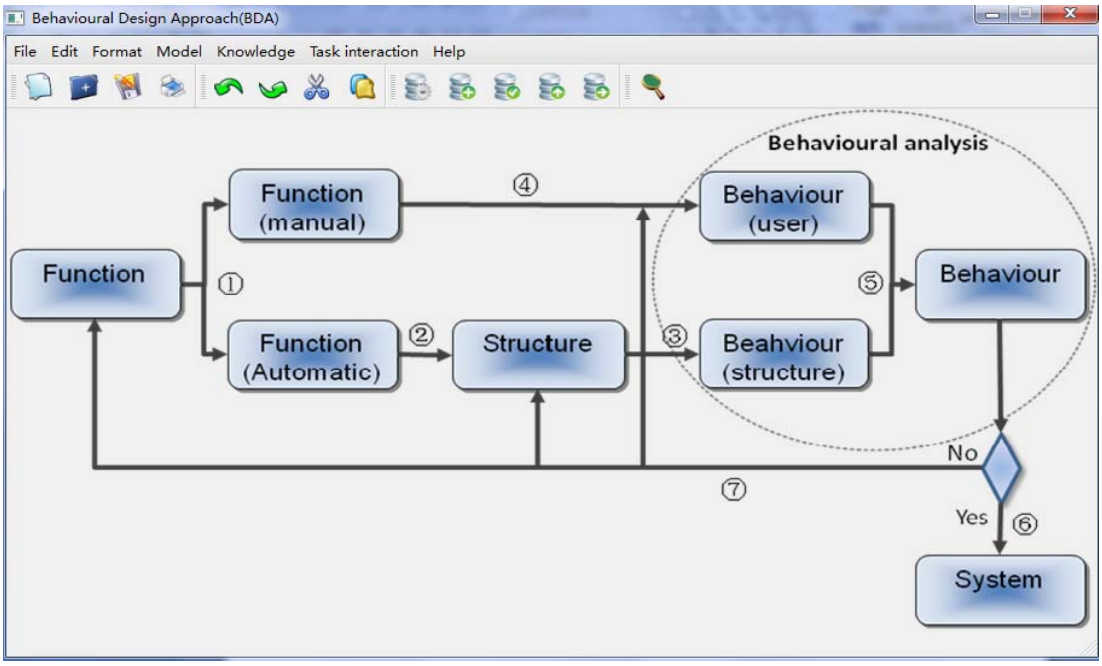
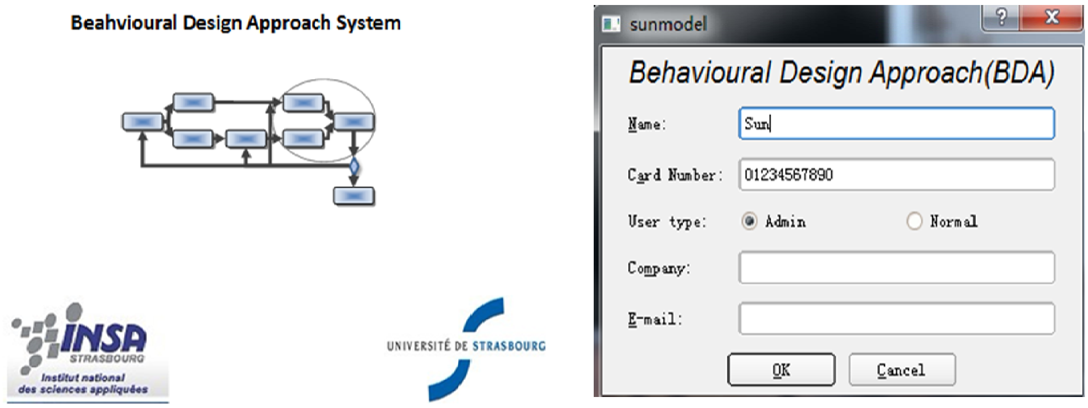


Fig 4.15 Main frame of BDAS

The main frame of BDA is composed by four core interfaces which are menu interface, fast button interface, soft panel and desktop area. Menu interface contains seven main menus such as *File, Edit, Format, Model, Knowledge, Task interaction* and *Help*. The items in the menu of *Model, Knowledge* and *Task interaction* are illustrated previously. Several buttons are displayed in the fast button interface, which provides a direct access to the different functions. Each button with an icon can be activated after the correct *user login* operation. In the center of the soft panel is the global view of the behavioural design approach.

Login interface is also designed for the designers to login in the system. Login interface gives the authorization to persons to be able to use the software. Four text fields of the *Name, Card Number, Company* and *E-mail* and a permission choice of *User type* are designed in the interface for inserting the using information. Two buttons are designed for entering and exiting the software. Besides the *Start cover* of the software is designed, as shown in figure 4.16.



(a) (b)

Fig 4.16 (a) Start cover, (b) Login interface

As we explained above, the core functional requirements and their corresponding interfaces in BDA are displayed. Because of the limit of the space, the detailed program code will not be shown here. And some additional auxiliary interfaces for supporting the software will not be explained either. In the following table 4.2, we list the representations of the icons which we used in our software in order to clarify their meanings.

Table 4.2 The explanations of the icons in BDAS

Icon	Corresponding function	Icon	Corresponding function
	New file		Open file
	Save file		print
	Undo		Redo
	Cut		Copy
	Ontology base		Function base
	Structure base		Behaviour base
	Task base		Task comparison

Based on the behavioural design approach, the interfaces of Knowledge Bases and Task Interaction correspond to the two major behavioural design activities for engineering design and also the major functions of BDAS software. The interfaces in the software provide a convenient way for designers to take into account and to respect standards, safety and ergonomics legislations in the early design phase to improve product performance.

4.5 Conclusion of the chapter

In this chapter, our research aim is to build a computer aided system for supporting mechanical engineering design. We have implemented and realized our behavioural design approach integrated with the task model and knowledge bases. According to the behavioural design model, task model and knowledge bases we discussed in Chapter3, we follow the framework of the distributed BDA system for engineering design to build the prototype system, which contains three steps such as the target of the system framework, scenario of system modelling and implementation of the software prototype.

In the section 4.1, because of the no computer-aided software that permits combining all these aspects which we discussed above into the design, we introduce the direction of the prototype from the technical point of view and designer's (software user) point of view. We also select the software development environment and tools, and implant the software by using Visual C++ program language.

In the section 4.2, we have defined the targets of the BDA system framework and its relationships with the existing technical knowledge to help designers to improve product performance. After introducing the functional requirement of the system, the BDA system framework is designed based on the integrated approach of BDA for mechanical engineering design.

In the section 4.3, in order to translate the behavioural design modelling and their information models into computer language, the UML is used to model the overall structure of the BDA system. System modelling consists of use case modelling, class diagram modelling, activity diagram modelling and sequence diagram modelling. In order to reduce the mistakes in our modelling for advanced programming stage, the dynamic models have been verified through the function of verification in the software *StarUML*.

In section 4.4, after introducing the BDA system framework integrated with the task models and knowledge bases which are explained by UML modelling, we have implanted the software by using Visual C++ program language. The systemic knowledge model has been implemented into five distinct knowledge bases, such as Ontology Base, Function Base, Structure Base, Behaviour Base and Task Base. The *SolidWorks* interface has been implemented to help designers to transform the CAD information into BDAS software. The Behavioural Analysis functions such as task input, task comparison and information agent management have been built into the corresponding interfaces. At last, start cover, login interface and software interface have been integrated into a single application that is the BDAS prototype.

Chapter 5 Application, conclusions and future work

In this chapter, a case study will be discussed based on the behavioural design approach, the conclusions of the thesis will be made by reviewing the research questions presented in Chapter 1 and making recommendations for future studies in relation to further development of Behavioural Design Approach and correlative software.

5.1 Discussion on the applicability

This section is about how BDAS actually works in practice. This section will therefore not discuss hard research findings but will instead report on our experiences with our design method. When we used the method in education we had the opportunity to supervise the correct application of the method and to correct when necessary.

In this case study, we research with a group of undergraduate student to finish the course project of utility cutter design.

A utility cutter is an inexpensive tool made by enclosing a razor blade in a handle which provides a hand-grip. Utility cutters are used in a variety of commercial and domestic applications, including construction, household repairs, arts and crafts and many others. Utility cutters provide the extremely sharp blade which is scored into segments, so that as the tip of the blade become dull and end segment can be broken off. And then, the user can continue using the cutter by exposing the next successive segment, which is still sharp.

According to the typical utility cutter design, students design the utility cutter design as shown in figure 5.1.

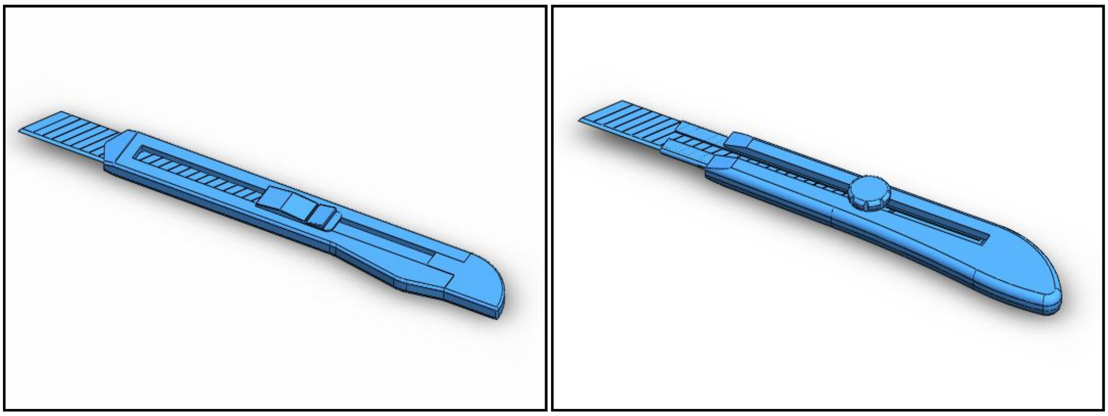


Fig 5.1 Typical model of utility cutter

And then, we use BDAS to call a *SLDASM* file form the *Solidworks*, as shown in figure 5.2.

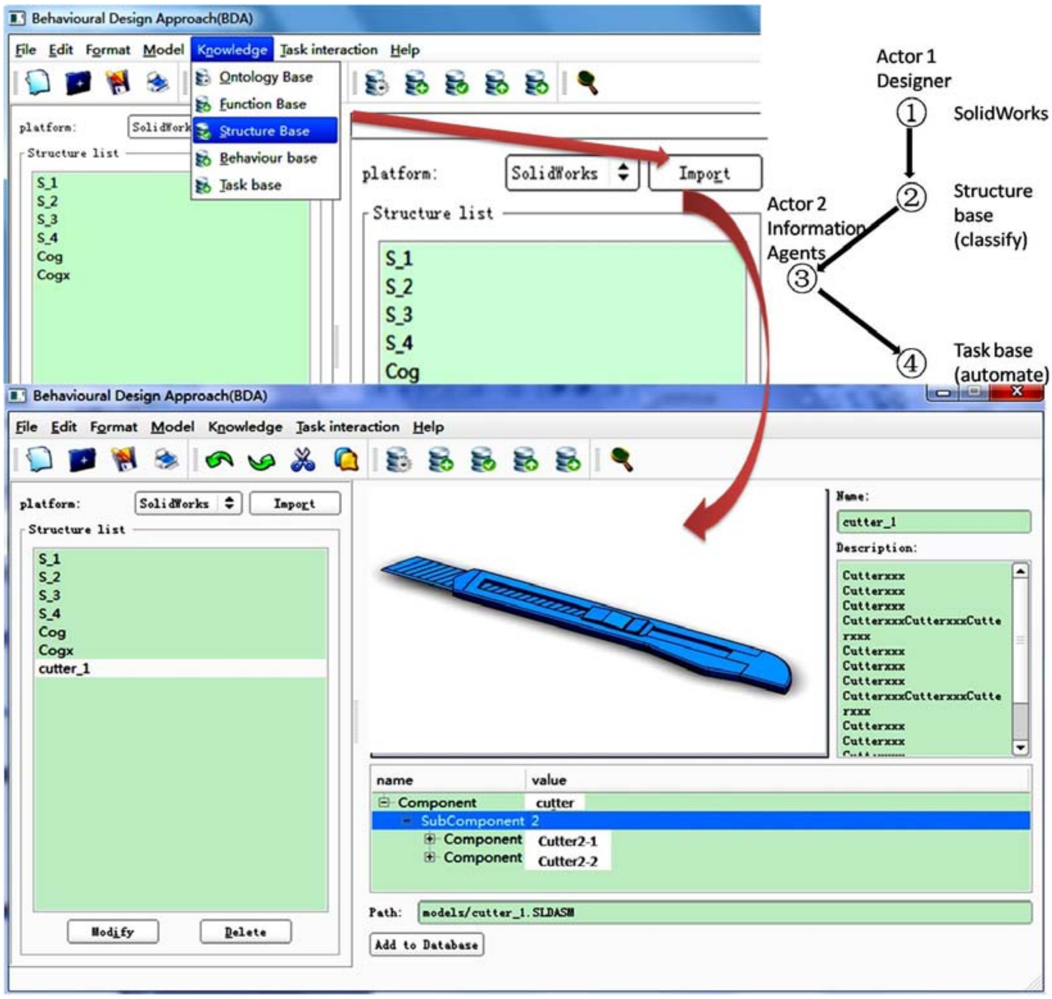


Fig 5.2 Analysis of the utility cutter derived from Solidworks (structure task)

And then, the *Information Agents* receive the *SLDASM* file and transfer the information into the structure base; BDA system classifies the obtained structures into existed structures and new structures. In this case study, after analyzing all the obtained structures, *Information Agents* do not find any new structures. If there is some new tasks, students should add it into the structure base and give the corresponding tasks into the *Technical Task Base* (structure).

Based on our behavioural design approach, the utility cutter design is intended to influence or result in certain user behaviour. The behavioural comparison diagram is created by the interaction of the user and the product, according to Task Plans discussed in Chapter4, as shown in figure 5.3.

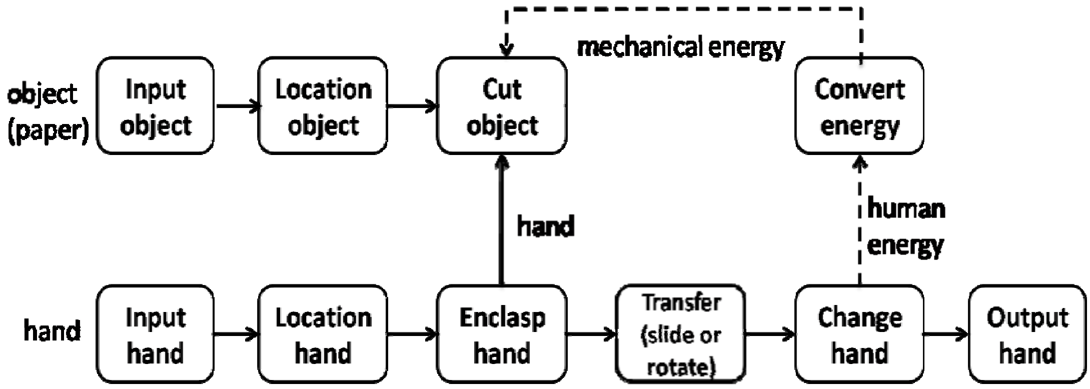


Fig 5.3 A behavioural comparison for a standard utility cutter

Specifically, it represents what product functions are needed for, or correspond to, user behaviour. For example, in this utility cutter example, the product function of input hand is needed to support the user task of grab. Each user task and product function are compared and analyzed. Then, the product functions are grouped with their user behaviours. User behaviours are shown in the behavioural comparison diagram to distinguish them from product functions. Using this sequence of behaviours, the behaviour diagram is combined with the functional model to create a single graphical representation of a user behaviour and a product function that is thus user centric and makes available information that is needed during the early stages of design.

So, in this step, students input manual functions which are derived from the FA, such as grab handle, release blade, translate device to cut and so on; Information Agents receive the information and transfer them into *Function Base (manual)*; BDA system makes an analogy between *Function base (manual)* and *Social Task Base (user)*; if there is no task matches the functions, students input the new task which are analyzed in figure 5.3 according to the Task Plans. The classification and analysis of user tasks is shown in figure 5.4.

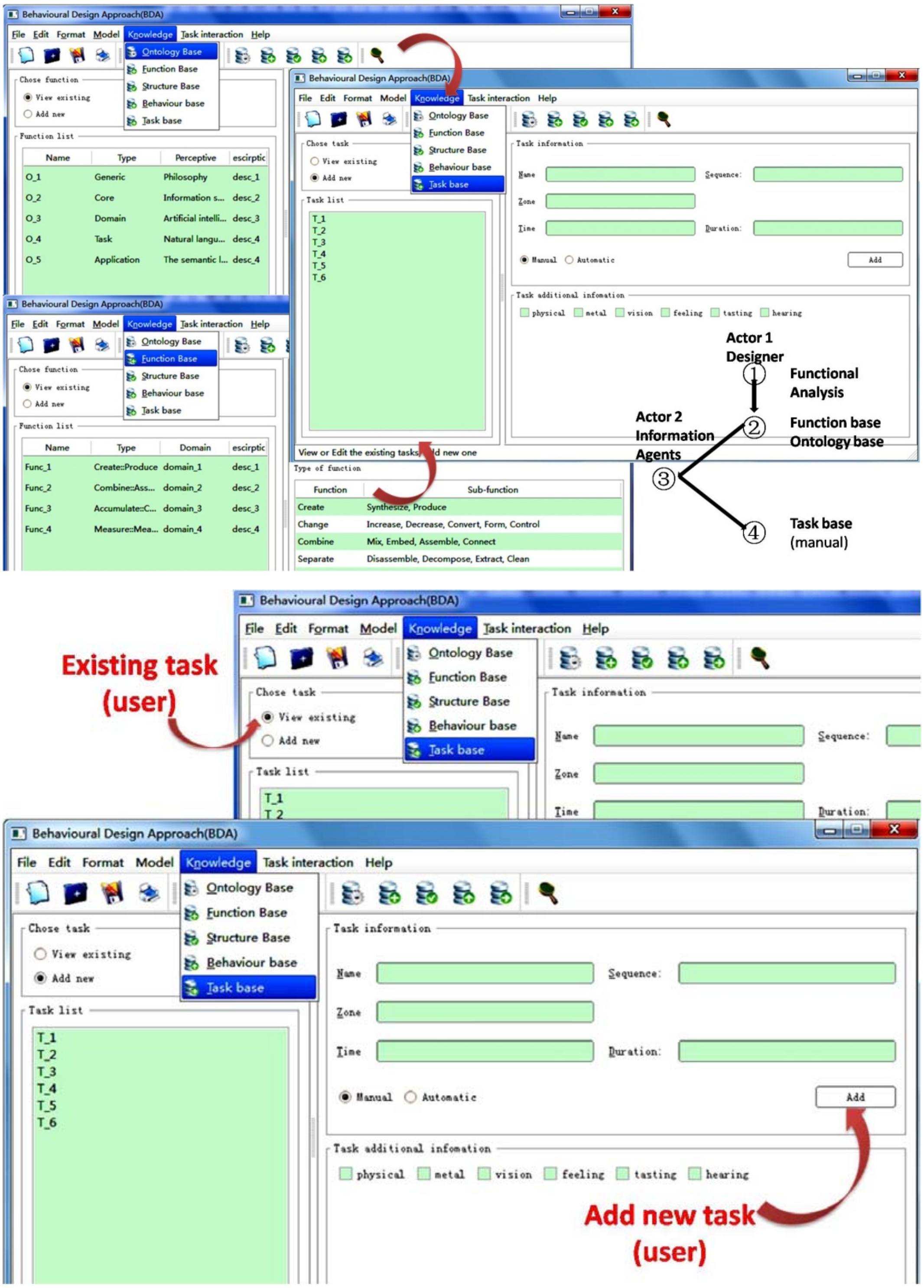


Fig 5.4 Classification and analysis of user tasks (utility cutter)

After that, the *Information Agents* receive two results: analysis of structure of utility cutter (structure task) and analysis of manual function of utility cutter (user task); and then the evaluation step occurs in *Task Comparison*, as shown in figure 5.5.

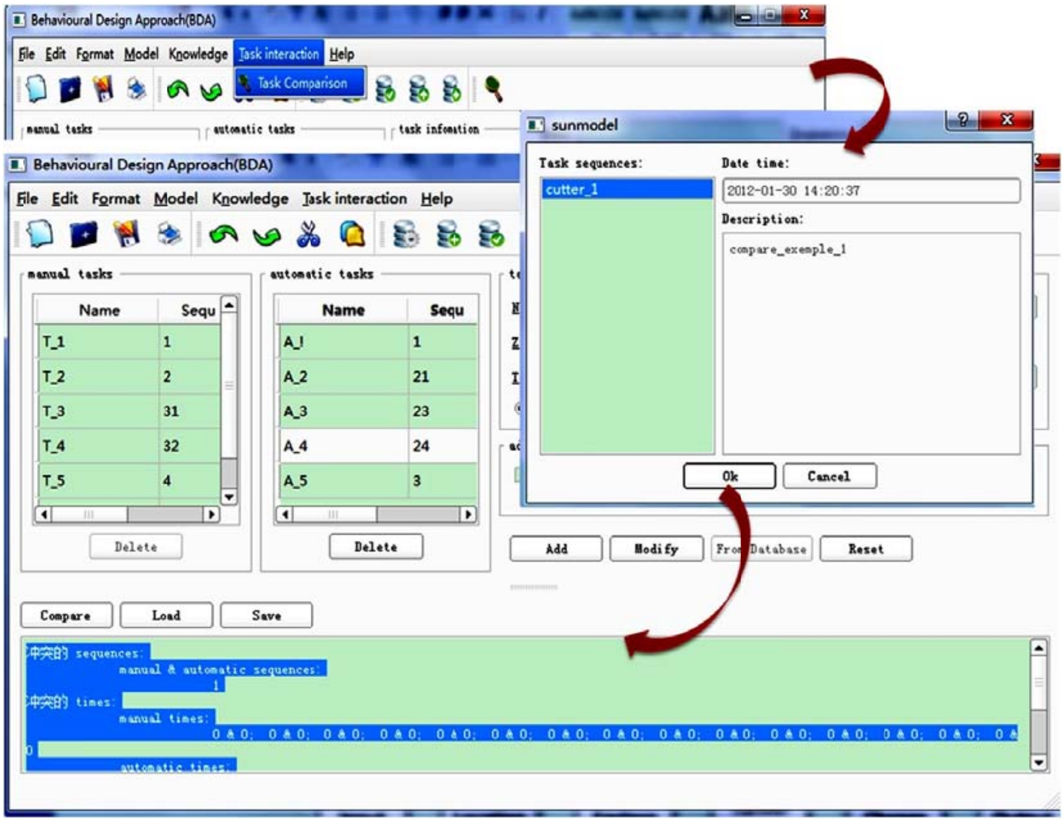


Fig 5.5 The evaluation of Task comparison (utility cutter)

Finally, the Information Agents receive the feedback of result of *Task Comparison*. In this case study, the BDAS suggest one dangerous zone according to safety standard and two suggestions according to ergonomics standard, as shown in figure 5.6.

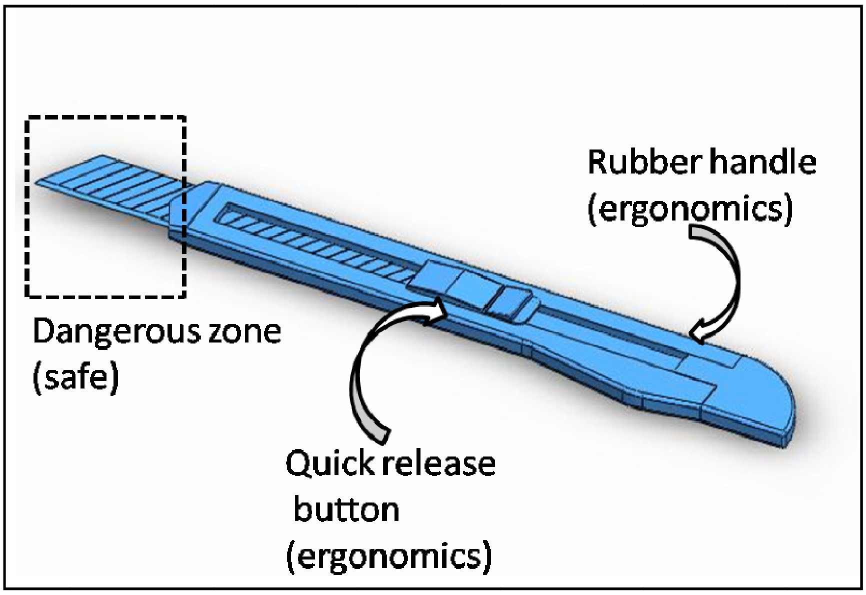


Fig 5.6 The result of Task Comparison

According to the result of *Task comparison* to consider the typical utility design, it contains two major pats: a replaceable blade, and a sturdy handle. The blade is designed to be replaced whenever it starts to dull, while the handle can be used for years. Many utility cutters are made with double-ended blades so that the blade can be flipped around and used again. The housing for the blade may also include a storage space for several extra blades for convenience.

However, utility cutters present a number of disadvantages, particularly in heavy duty applications such as construction. Eventually, when the last segment has been broken off of the blade, the blasé must be replaced. Conventionally, this requires removing the sliding body which retains the blade in the knife handle. This can be a difficult operation to perform without a surface to rest the cutter on (for example while on a ladder). This will increase opportunities for injury because of the manipulation required in order to hold the blade on the sliding body while inserting it into the cutter handle.

Moreover, in heavy duty applications there is a considerable amount of force experienced by the blade. For example when cutting hard surfacing materials, it tends to deflect the blade laterally. The primary resistance to deflection of the blade within the handle is provided by the throat of the handle. This provides a single layer of supporting surface (generally metallic) on each side of the handle. Accordingly, over time the cutting force on the blade will widen the throat of the cutter handle, which results allowing the blade to deflect more when in use. This results in inaccurate cuts and the blade segments breaking off prematurely.

So, after all these analysis, the students give a new plan about the utility cutter. It would accordingly be advantageous to provide a utility cutter with a quick release which allows the blade to be changed and with minimal manipulation. It would further be advantageous to provide a cutter structure which reinforces the throat of the cutter. It will reduce deflection of the blade during use and thus minimize opportunities for inaccurate cuts and premature breakage of the blade

Utility cutters provide the extremely sharp blade which is scored into segments, so that as the tip of the blade become dull and end segment can be broken off. Although the segment can be easy to break off, it also causes the dangerous to the users. For example when cutting hard surfacing materials, it tends to deflect the blade laterally. The redesign of utility cutter is shown in figure 5.7.

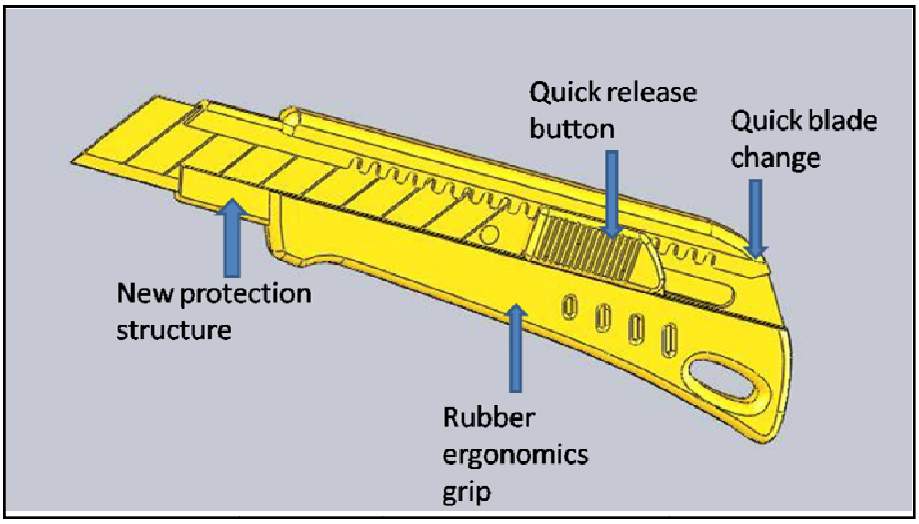


Fig 5.7 New style of utility cutter

The attributes of the new style of utility cutter are listed as follow:

- New protection structure avoids the break off of the segment (apply for a patent);
- Thumb-dial blade release makes blade change easy;
- Blade cartridge automatically slides new blade into position after old blade is removed;
- Smooth, slider mechanism features audible click stop and is self-locking;
- Rubber panels for secure, comfortable grip;
- Yellow ABS plastic body is highly visible.

The BDAS tool has been evaluated using a simple cutter design that allowed determining some problems n designed product and by consequence it draws attention to the necessary modification to improve cutter use. Students reported their opinions and gave suggestions for improvements. However, speaking with the students and analyzing the complaints they reported, proved more useful for the development of the tool. The following issues came up in all evaluations.

BDAS is a tool for designers who:

- are familiar with the design process;
- have extensive knowledge of the product or process under design;
- are able to take assessment results needed to reduce risk and improve product use.

The designer or analyst is responsible for resolving use issues revealed through the behaviour assessment. For this reason, designers who do not meet the above listed qualifications should proceed with caution. BDAS is an engineering tool, and much like other engineering tools the designer's skill and experience are a critical component in obtaining a useful result.

BDAS is a technique that can be employed at early stage of the life cycle of a system. It should be seen as an enhancement to sound design using experience-based and behaviour-based approaches. BDAS is a guide, not an expert system. It does not propose the solution but only identify the problem and its place. If the user inputs poor and/or incomplete data, the results from the program will be no better. The results from BDAS should be considered subjective. Much of the data input to BDAS are subjective decisions based on good engineering judgment. Therefore, the results from the assessment are no more objective than the inputs. BDAS helps provide more complete information organized in a usable form from which engineering decisions can be made based upon good collective judgment.

The following improvement of BDAS has successively interdependent relationship:

- Add the function of dangerous zone into the three-dimensional model: depending on the secondary development technology based on CAD platform, BDAS can realize manual input dangerous zone into specified three-dimensional model.
- Animation of operational mechanism: through the real time simulation database and some simulation functions, connect the simulation interfaces effectively with those characters of dangerous zones to realize the direct demonstration.
- Automatic recognition and tracing of feature data of mechanism parts model: typically, the recordings of three-dimensional CAD modeling tools are mainly about the various characteristics of data of parts, such as diameter, thickness, angle and so on. When BDAS judges the dangerous zones of the product, the system always extracts some key feature sizes as the basis. If the system can effectively distinguish detailed sizes and key feature sizes, the intelligent degree and easy to use of the system could be effectively improved.
- Automatic judgment and enumeration of dangerous zones: on the basis of recognition of mechanism feature data, according to some certain algorithm, the system could judge static feature sizes. Based on the specific artificial intelligence algorithms, the system could judge several possible dangerous zones. Finally, the designers estimate the real dangerous zones.
- Simulation function of mechanism motion: on the basis of judgment of mechanism feature sizes, the system could add features of mechanism simulation, such as chain transmission, gear transmission and so on. According to the combination of feature sizes and motion characteristics, the

system could automatically judge the dangerous zone. Finally, the system could become highly intelligent.

5.2 Summary of the thesis

Both the empirical and theoretical studies have shown that there is a need for design methods which focus on the user aspects in the design activities. This thesis offers a suggestion for a design approach to be used in the synthesis part of the design work. The procedure treats the artefact to be designed as a user-technical system instead of simply a technical system. It makes it possible to consider and also where necessary focus on the user aspects.

The research has shown that it is possible to place the user behaviours next to the technical functions and break them down in a unified hierarchical structure, namely a behavioural design approach. An element in the procedure is to investigate connections among functions, structures and behaviours in the behavioural design approach. By doing so, it is possible to find relevant requirements on the design. For the same reason, a more structured scenario method is also introduced in the procedure.

Chapter 1 establishes the research questions and problem statements, and the purpose of the thesis is the development and evaluation of a top-down technical and socio-technical framework for engineering design, which integrates various knowledge bases and the task model.

In Chapter 2 more specifically, we take a multi-level and comprehensive strategy for researching the multidisciplinary state of the art in the domains of design, engineering design, design theories and design approaches.

First, we try to understand what is design? We have reviewed the concepts related to design as well as their current problems and difficulties. Based on our analyses, design is one of the most crucial sectors of the economy. Design has been considered as both a technical and a socio-technical activity. We have identified the way to summarize the challenges presented by the design form four aspects, such as creativity, complexity, choice and compromise.

Then, we have investigated the literature about engineering design and engineering design process. By comparing the various processes of design, we have noted that 70% of a product’s total cost is determined by its design, especially in the early design stage. In order to improve the product performance, the early design stage is more important.

Finally, In order to understand the realization of the engineering design process, we have reviewed some known design theories and approaches in the following section. We find that product design is usually performed simply taking into consideration product functions and structures, while users' behaviours in terms of using the product are generally not fully considered during the early design phase. So, in order to improve product performance, our research targets a better integration of product and user behaviour during the early design phase.

In Chapter 3, we have presented our behavioural design approach step by step with reference to our research questions. From the perceptive of technical and socio-technical thinking, we have proposed a systemic behavioural design approach according to the characteristics of mechanical engineering design, which is composed by the behavioural design model, task model and various knowledge bases. The behavioural design approach aims to a better integration of system-use conditions into system behaviour, starting from the early design phase. We introduce the task model to realize the mapping process between the function and behaviour, structure and behaviour. These tasks significantly concern the analysis and specification of the utilization conditions. Otherwise, functional allocation and decomposition between user and machine, user machine interface, working space, and information transmission are defined as our research object. For these reasons, Task Plans framework is proposed as a useful tool to help designers to determine the task. Moreover, four concepts of knowledge bases (ontology base, structure base, task base and behaviour base) are introduced to help designers to make decisions, due to the designers' requirements of much more knowledge.

In Chapter 4, in order to integrate the behavioural design approach into the engineering design approach, a combination of the accurate industrial context allows us to define all the factors which are necessary to show and confirm the applicability of our approach. In correspondence with this aim to build a computer aided system based on our behavioural design approach, we have proposed a system framework of BDA based on the hierarchical information model for engineering design. Then, by translating the behavioural design modelling and their information models into computer language, the UML is used to model the overall structure of the BDA system. Also, following the system framework, the detailed static and dynamic structure of the BDA system have been modeled and verified by UML.After introducing the BDA system framework integrated with the task models and knowledge bases which are explained by UML modelling, we have implanted the software by using Visual C++ program language.

5.3 Recommendation for further study

Developing a design approach almost never ends. There are still many aspects that need to be improved both on the theoretical as the practical side. From one hand, theories are important to link together many areas in the process of engineering design. With the help of some theories we can develop practical techniques that make improvements of the process and product possible. On the other hand, many techniques are also developed unplanned when designers face problems. Studying them can also contribute much to theoretical understanding.

The work leading to this thesis has generated many interesting and promising ideas. Some of those future developments are promising ideas that we believe are worth exploring, others are equally interesting ideas that we have dropped during the research program due to the difficult reasons. The opportunities that could be pursued further for extending this work are listed below:

First, to arrive at an efficient procedure for product development, which integrates the user aspects, numerous additional questions have to be answered. How can the designer obtain the more knowledge of the use type?

Second, the behaviour driven database developed in this work includes knowledge bases about only some functional and behavioural features. These databases could be extended to encompass other classes of function base, structure base ontology base, user base and task base (e.g. functional modules, structural standard parts, human behaviour).

Third, the case studies, which were performed for this study, are very simple cases. For the future studies, the procedure can be tested with real cases, in order to understand its appropriateness for providing comparisons of user behaviour and structure behaviour input to the design process. Therefore the BDAS should be constructed in an actual project scenario within an existing company setting. In addition to that, the outcomes of the BDAS were not presented to practicing designers with the aim of conceiving the impact of the data that is provided by the studies. The future applications of the procedure can involve the practicing designers for observing whether they are able to utilize the outcomes as knowledge input for the engineering design process.

Fourth, in the implementation, the constraints (such as environment, economic, sustainability and so on) have not been used during the search for possible solutions. An extension to include constraints in the search process could be implemented in future versions of this BDAS tool. Exploring the integration of the BDAS tool into commercial CAD systems to give the designer a more flexible design environment and reduce the gap with detailed design stage could be an opportunity to improve this work further. Our tool BDAS proves that there is indeed a need for tool support in the early activities of behavioural design. The tool is not finished and more development could lead to a tool of commercial quality. Tools are nowadays essential elements in most design processes. Any method for design could greatly benefit from tool support both for the process itself but also for acceptance in the industry.

The issues mentioned above are not only relevant from a behavioural design perspective. The need for better processes, a better understanding of quality and how to reach it is central in the field of user behaviour and product behaviour design. The most important thing in behavioural design is the central position of the users and their work. The research described in this thesis uses that viewpoint to create better tools and techniques. We feel that we made a small step forwards but still a lot more progress is needed to fully achieve our goals.

Reference

Ackoff, R. L. (1989). "From Data to Wisdom." Applied Systems Analysis **16**: 3-9.

Adolph, S., A. Cockburn, et al. (2002). Patterns for effective use cases, Addison-Wesley Longman Publishing Co., Inc.

Akao, Y. (1990). Quality function deployment, Productivity Press Cambridge MA.

Al-Salka, M., M. Cartmell, et al. (1998). "A framework for a generalized computer-based support environment for conceptual engineering design." Journal of Engineering Design **9**(1): 57-88.

Altshuller, G. and S. Rodman (1999). The innovation algorithm: TRIZ, systematic innovation and technical creativity, Technical Innovation Ctr.

Altshuller, G. and L. Shulyak (2002). 40 principles: TRIZ keys to innovation, Technical Innovation Center, Inc.

Andersson, K., P. Makkonen, et al. (1995). "A proposal to a product modelling language to support conceptual design." CIRP Annals-Manufacturing Technology **44**(1): 129-132.

Ashby, M. F., S. Décarroux, et al. (2000). Choix des matériaux en conception mécanique, Dunod.

Asimow, M. (1962). Introduction to design, Prentice-Hall Englewood Cliffs, NJ.

Ayala, F. J. (2008). "Science, evolution, and creationism." Proceedings of the National Academy of Sciences **105**(1): 3.

Bagge, S. and A. M. Pendrill (2002). "Classical physics experiments in the amusement park." Physics Education **37**: 507.

Bateson, P., P. P. G. Bateson, et al. (1999). Design for a life: How behaviour develops, Vintage.

Beckman, T. J. (1999). "The current state of knowledge management." Knowledge management handbook **1**(5).

Blanchard, H. E. (1997). "User interface standards in the ISO Ergonomics technical committee." ACM SIGCHI Bulletin **29**(1): 20-22.

Blessing, L., A. Chakrabarti, et al. (1998). "An overview of descriptive studies in relation to a general design research methodology." Designers–The Key to Successful Product Development: 42-56.

Booch, G., J. Rumbaugh, et al. (2005). Unified Modelling Language User Guide, The (Addison-Wesley Object Technology Series), Addison-Wesley Professional.

Boothroyd, G. and P. Dewhurst (1987). Product design for assembly, Boothroyd Dewhurst Inc.

Bracewell, R. and J. Sharpe (1996). "Functional descriptions used in computer support for qualitative scheme generation-" Schemebuilder"." Artificial Intelligence for Engineering Design Analysis and Manufacturing **10**: 333-346.

Braha, D. and O. Z. Maimon (1998). A mathematical theory of design: foundations, algorithms, and applications, Kluwer Academic Publishers.

Bray, T., J. Paoli, et al. (2000). Extensible markup language (XML) 1.0, W3C recommendation.

Breitman, K. K. and J. C. S. do Prado Leite (2003). Ontology as a requirements engineering product, IEEE.

Brown, D. C., M. B. Waldron, et al. (1998). Intelligent computer aided design, North-Holland.

Buttars, S. and R. Rowland (2006). "Design for manufacture." ELEKTRON JOURNAL-SOUTH AFRICAN INSTITUTE OF ELECTRICAL ENGINEERS **23**(2): 24.

Buzacott, J. A. (2002). "The impact of worker differences on production system output." International Journal of Production Economics **78**(1): 37-44.

Carayon, P. (2006). "Human factors of complex sociotechnical systems." Applied Ergonomics **37**(4): 525-535.
Carnap, R. (1956). "Empiricism, semantics, and ontology." The Philosophy of Science: 85-97.
Cavallucci, D., P. Lutz, et al. (2000). "Intuitive design method (IDM): a new framework for design method integration." Journal for Manufacturing Science and Production **3**(2-4): 95-102.
Chakrabarti, A. and T. P. Bligh (1994). "An approach to functional synthesis of solutions in mechanical conceptual design. Part I: Introduction and knowledge representation." Research in Engineering Design **6**(3): 127-141.
Chandrasekaran, B. (1990). "Design problem solving: A task analysis." AI Magazine **11**(4): 59.
Chandrasekaran, B. (2005). "Representing function: Relating functional representation and functional modelling research streams." Artificial intelligence for engineering design, analysis and manufacturing **19**(2): 65-74.
Chandrasekaran, B. and J. R. Josephson (2000). "Function in device representation." Engineering with Computers **16**(3): 162-177.
Ciocoiu, M., D. S. Nau, et al. (2001). "Ontologies for integrating engineering applications." Journal of Computing and Information Science in Engineering **1**: 12.
Clarkson, J. (2003). Inclusive Design: Design for the whole population, Springer Verlag.
Clarkson, J. and C. Eckert (2005). Design process improvement: a review of current practice, Springer Verlag.
Cole Jr, E. L. (2002). "Functional analysis: a system conceptual design tool [and application to ATC system]." Aerospace and Electronic Systems, IEEE Transactions on **34**(2): 354-365.
Collins, J., B. Hagan, et al. (1976). "The failure-experience matrix: a useful design tool." Journal of Engineering for Industry **98**(3): 1074-1079.
COM (2011). from http://www.microsoft.com/com/default.mspx.
Community Design Regulation (2002). " Commission Regulation (EC) No 2246/2002." European Union regulation.
Conway, J. B. (1990). A course in functional analysis, Springer.
Coulibaly, A., R. Houssin, et al. (2008). "Maintainability and Safety indicators at design stage for mechanical products." Computers In Industry **59**(5): 438-449.
Cross, N. (2000). Engineering Design Methods: Strategies for Product Design. Chichester, Wiley.
Cross, N. (2008). Engineering Design Methods: Strategies for Product Design. London, John Wiley & Sons.
Cross, N. and Knovel (2000). Engineering design methods: strategies for product design, Wiley Chichester.
Crowther, P. (1999). "Design for disassembly." BDP Environment Design Guide.
Csikszentmihalyi, M. (1996). Creativity: Flow and the Psychology of Discovery and Invention. New York, HarperCollins.
Cullen, L. (2007). "Human factors integration-Bridging the gap between system designers and end-users: A case study." Safety science **45**(5): 621-629.
Curry, H. B. (1977). Foundations of mathematical logic, Dover Pubns.
Danish, G. (2007). Design Denmark, Danish government's 2007 white paper.
Darses, F. and M. Wolff (2006). "How do designers represent to themselves the users' needs?" Applied Ergonomics **37**(6): 757-764.
Das, B. and A. K. Sengupta (1996). "Industrial workstation design: A systematic ergonomics approach."

Applied Ergonomics **27**(3): 157-163.
Davies, J., R. Studer, et al. (2005). "Next generation knowledge management." BT Technology Journal **23**(3): 175-190.
Debenham, J. K. (1989). Knowledge systems design, Prentice-Hall, Inc.
Deng, Y. M., G. Britton, et al. (1998). "A design perspective of mechanical function and its object-oriented representation scheme." Engineering with Computers **14**(4): 309-320.
Deng, Y. M., S. Tor, et al. (2000). "Abstracting and exploring functional design information for conceptual mechanical product design." Engineering with Computers **16**(1): 36-52.
Department of Trade and Industry (2005). Creativity, Design and Business Performance. DTI (Department of Trade and Industry) ECONOMICS PAPER No.15**:** 76.
Dictionary, O. E. (2005). "Concise Oxford English Dictionary." Retrieved December **5**: 2005.
Dieter, G. E., L. C. Schmidt, et al. (2009). "Engineering design." Journal of Mechanical Design **131**: 056501.
Dietz, P. H. and W. S. Yerazunis (2001). Real-time audio buffering for telephone applications, ACM.
Diggle, G. (1995). "Science and Judgement in Risk Assessment." Occupational and Environmental Medicine **52**(11): 784.
Dini, G., F. Failli, et al. (2001). "A disassembly planning software system for the optimization of recycling processes." Production Planning & Control **12**(1): 2-12.
Dixon, J. R. and C. Poli (1995). Engineering design and design for manufacturing: a structured approach: text and reference for mechanical engineers, Field Stone Publishers.
Dorst, K. and P. E. Vermaas (2005). "John Gero¡¯s Function-Behaviour-Structure model of designing: a critical analysis." Research in Engineering Design **16**(1): 17-26.
Drew, S. (1999). "Building knowledge management into strategy: making sense of a new perspective." Long Range Planning **32**(1): 130-136.
Dym, C. L. (1994). Engineering design: a synthesis of views, Cambridge Univ Pr.
Dym, C. L. and R. E. Levitt (1991). Knowledge-based systems in engineering, McGraw-Hill.
Dym, C. L. and R. E. Levitt (1991). "Toward the integration of knowledge for engineering modelling and computation." Engineering with computers **7**(4): 209-224.
Edwards, I. R., B. E. Wiholm, et al. (1996). "Concepts in risk-benefit assessment." Drug safety **15**(1): 1-7.
EIMaraghy, W. and R. Urbanic (2004). "Assessment of manufacturing operational complexity." CIRP Annals-Manufacturing Technology **53**(1): 401-406.
Eppinger, S. D., D. E. Whitney, et al. (1994). "A model-based method for organizing tasks in product development." Research in Engineering Design **6**(1): 1-13.
Erden, M., H. Komoto, et al. (2008). "A review of function modelling: Approaches and applications." Artificial Intelligence for Engineering Design Analysis and Manufacturing **22**(2): 147.
Eriksson, H. E. and M. Penker (2000). Business modelling with UML, John Wiley & Sons.
Ertas, A. and J. C. Jones The engineering design process, John Wiley, New York.
European Communities (2009). design as a driver of user-centered design, Commision of staff working document.
Feng, S. C., W. W. Nederbragt, et al. (1999). Incorporating process planning into conceptual design, Citeseer.
Feng, S. C. and E. Y. Song (2003). "A manufacturing process information model for design and process planning integration." Journal of Manufacturing Systems **22**(1): 1-15.

Finger, S. and J. R. Dixon (1989). "A review of research in mechanical engineering design. Part I: Descriptive, prescriptive, and computer-based models of design processes." Research in Engineering Design **1**(1): 51-67.

Fowler, T. C. (1990). Value analysis in design, Van Nostrand Reinhold New York.

Franklin, S. and A. Graesser (1997). "Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents." Intelligent Agents III Agent Theories, Architectures, and Languages: 21-35.

Frederic Noel, M. (2006). "A dynamic multi-view product model to share product behaviours among designers: how process model adds semantic to the behaviour paradigm." International Journal of Product Lifecycle Management **1**(4): 380-390.

French, M. J. (1999). Conceptual design for engineers, Springer Verlag.

Gómez-Pérez, A., M. Fernández, et al. (1996). Towards a method to conceptualize domain ontologies, Citeseer.

Gardoni, M. (2005). "Concurrent engineering in research projects to support information content management in a collective way." Concurrent Engineering Research and Applications **13**(2): 135-144.

Geer, D., R. Bace, et al. (2003). "Cyberinsecurity: The cost of monopoly how the dominance of microsoft's products poses a risk to security." Computer and Communications Industry Association Report.

Gero, J. S. (1990). "Design prototypes: a knowledge representation schema for design." AI Magazine **11**(4): 26-36.

Gero, J. S. and U. Kannengiesser (2004). "The situated function-behaviour-structure framework." Design Studies **25**: 373–391.

Gero, J. S. and U. Kannengiesser (2007). Locating creativity in a framework of designing for innovation. Trends in Computer Aided Innovation. ?, ?**:** 57-66.

Gero, J. S. and M. L. Maher, Eds. (1992). Modelling Creativity and Knowledge-Based Creative Design. New Jersey, USA, Lawrence Erlbaum Associates.

Gero, J. S. and M. L. Maher (1997). A framework for research in design computing, Citeseer.

Gero, J. S. and M. A. Rosenman (1990). "A conceptual framework for knowledge-based design research at Sydney University's Design Computing Unit." Artificial Intelligence in Engineering **5**(2): 65-77.

Godfrey-Smith, P. (2003). Theory and reality: An introduction to the philosophy of science, University of Chicago Press.

Goel, A. K. (1997). "Design, analogy, and creativity." IEEE expert **12**(3): 62-70.

Gomaa, H. (2006). Designing concurrent, distributed, and real-time applications with UML, ACM.

Gorti, S., A. Gupta, et al. (1998). "An object-oriented representation for product and design processes." Computer Aided Design **30**(7): 489-502.

Grabowski, H., S. Rude, et al. (1998). Universal design theory, Shaker.

Greenbaum, J. M. and M. Kyng (1991). Design at work: Cooperative design of computer systems, CRC.

Gyi, D., R. Cain, et al. (2010). "The value of computer-based product representations in co-designing with older users." Journal of Engineering Design **21**(2): 305-313.

Handbook, V. D. I. D. (1987). "2221: Systematic Approach to the Design of Technical Systems and Products (translated by K. Wallace)." VDI-Verlag, Düsseldorf.

Harris, C. E., M. Davis, et al. (1996). "Engineering ethics: what? why? how? and when?" JOURNAL OF ENGINEERING EDUCATION-WASHINGTON- **85**: 93-96.

Hasan, R., A. Bernard, et al. (2003). "Integrating safety into the design process: elements and concepts

relative to the working situation." Safety science **41**(2-3): 155-179.
Hatchuel, A. (1996). "Théories et modèles de la conception, Cours d'ingénierie de la conception." Paris, Ecole des mines de Paris.
Hatchuel, A. and B. Weil (2003). A new approach of innovative design: an introduction to C-K theory. Proceedings of the international conference on engineering design (ICED'03), Stockholm, Sweden.
Hayman, A. and T. Elliman (2000). "Human elements in information system design for knowledge workers." International Journal of Information Management **20**(4): 297-309.
Hedrick, B., R. Urbanic, et al. (2004). Multi-tasking machine tools and their impact on process planning. CIRP 2004 International Design Conference.
Henderson, T. and S. Bhatti (2001). Modelling user behaviour in networked games, ACM.
Hendrick., H. W. (2003). "Determining the cost-benefits of ergonomics projects and factors that lead to their success." Applied Ergonomics **34**(5): 419-427.
Hendriks, P. H. J. and D. J. Vriens (1999). "Knowledge-based systems and knowledge management: friends or foes?" Information & Management **35**(2): 113-125.
Hernandez, J. O. (1995). Les systèmes de production automatisés: une approche socio-technique. U.F.R. des Sciences et des Techniques, Automatique et Informatique, Université de Franche Comté.
Heumann, J. (2003). "Introduction to business modelling using the unified modelling language (UML)." IBM Developer Works, http://www. ibm. com/developerworks/rational/library/360. html.
Hickey, A. M. and A. M. Davis (2003). "Requirements elicitation and elicitation technique selection: a model for two knowledge-intensive software development processes."
Hicks, B. J. and S. J. Culley (2002). "An integrated modelling environment for the embodiment of mechanical systems." Computer-Aided Design **34**(6): 435-451.
Hinrichs, T. R. (1992). Problem solving in open worlds: A case study in design, Lawrence Erlbaum.
Houkes, W. and P. Vermaas (2004). "Actions versus functions: A plea for an alternative metaphysics of artifacts." Monist **87**: 52-71.
Houkes, W., P. E. Vermaas, et al. (2002). "Design and use as plans: an action-theoretical account." Design Studies **23**(3): 303-320.
Houssin, R., A. Bernard, et al. (2006). "Information system based on a working situation model for a new design approach in concurrent engineering." Journal of Engineering Design **17**(1): 35-54.
Houssin, R., A. Bernard, et al. (2006). "Information system based on the working situation model for an original concurrent engineering design approach " Journal of engineering design **17**(1).
Houssin, R. and A. Coulibaly (2011). "An approach to solve contradiction problems for the safety integration in innovative design process." Computers in industry.
Houssin, R. and M. Gardoni (2009). "Software framework for the approach: Computer aided safety integration in design process(cased)." Journal of Advanced Manufacturing Systems **8**(1): 27-45.
Hsu, S. H., M. C. Chuang, et al. (2000). "A semantic differential study of designers' and users' product form perception." International Journal of Industrial Ergonomics **25**(4): 375-391.
Hsu, W. and I. M. Y. Woon (1998). "Current research in the conceptual design of mechanical products." Computer-Aided Design **30**(5): 377-389.
Huang, G. Q. (1996). Design for X: concurrent engineering imperatives, Springer.
Hubka, V. and W. E. Eder (1982). Principles of Engineering Design. London, Butterworth Scientific.
Huisg, S. and S. Kohn (2009). "Computer aided innovation--State of the art from a new product development perspective." Computers in Industry **60**(8): 551-562.
Hundal, M. (1990). "A systematic method for developing function structures, solutions and concept

variants." Mechanism and Machine theory **25**(3): 243-256.
ICSID (2008). "International Council of Societies of Industrial Design (Icsid) ". from www.icsid.org.
Ireson, W. G., C. F. Coombs, et al. (1996). Handbook of reliability engineering and management, McGraw-Hill Professional.
Ishii, K. (1995). "Life-cycle engineering design." TRANSACTIONS-AMERICAN SOCIETY OF MECHANICAL ENGINEERS JOURNAL OF MECHANICAL DESIGN **117**: 42-42.
Ivory, C., A. Thwaites, et al. (2001). Design for maintainability: the innovation process in long term engineering projects.
Johannessen, J. A., B. Olsen, et al. (1999). "Aspects of innovation theory based on knowledge management." International Journal of Information Management **19**(2): 121-139.
Juran, D. C. and L. W. Schruben (2004). "Using worker personality and demographic information to improve system performance prediction." Journal of Operations Management **22**(4): 355-367.
Kavakli, M., S. A. R. Scrivener, et al. (1998). "Structure in idea sketching behaviour." Design Studies **19**(4): 485-517.
Kazman, R., M. Klein, et al. (1999). "Evaluating software architectures for real-time systems." Annals of Software Engineering **7**(1): 71-93.
Keuneke, A. M. (1991). "Device representation-the significance of functional knowledge." IEEE EXPERT INTELLIGENT SYSTEMS and THEIR APPLICATIONS: 22-25.
Kidd, J. B. (1998). "Knowledge creation in Japanese manufacturing companies in Italy." Management Learning **29**(2): 131.
Kim, J. (2010). Philosophy of mind, Westview Pr.
Kirwan, B. E. and L. K. Ainsworth (1992). A guide to task analysis, Taylor & Francis.
Kitamura, Y. and R. Mizoguchi (2004). "Ontology-based systematization of functional knowledge." Journal of Engineering Design **15**(4): 327-351.
Koch, P., J. Peplinski, et al. (1994). "A method of design using available assets: Identifying a feasible system configuration." Behavioral Science **39**(3): 229-250.
Kusiak, A. and E. Szczerbicki (1992). "A formal approach to specifications in conceptual design." Journal of Mechanical Design **114**: 659.
Labrousse, M. (2004). Proposition d'un modèle conceptuel unifié pour la gestion dynamique des connaissances d'entreprise.
Lehmann, D. and M. Magidor (2002). "What does a conditional knowledge base entail?" Arxiv preprint cs/0202022.
Leplat, J. (1990). "Thème 3: Evolution Technologique et r le de l'ergonomie. Rapport de synthèse." Le Travail Humain **4**: 356-361.
Lerche, I. and W. Glaesser (2006). Environmental risk assessment: quantitative measures, anthropogenic influences, human impact, Springer Verlag.
Lewis, C. and J. Rieman (1993). "Task-centered user interface design." Available via ftp. cs. colorado. edu/pub/cs/distribs/clewis/HCI-Design-Books.
Lewis, T. (2005). "Coming to terms with engineering design as content." From the Editor **16**(2).
Li, W., W. Lu, et al. (2005). "Collaborative computer-aided design--research and development status." Computer-Aided Design **37**(9): 931-940.
Liao, S. (2000). "Case-based decision support system: architecture for simulating military command and control." European Journal of Operational Research **123**(3): 558-567.
Liao, S. (2002). "Problem solving and knowledge inertia." Expert systems with applications **22**(1):

21-31.
Liebowitz, J., Ed. (1999). The Knowledge Management Handbook. Boca Raton, CRC Press.
Liebowitz, J. (2001). "Knowledge management and its link to artificial intelligence." Expert Systems with Applications **20**: 1-6.
Liebowitz, J. and T. Beckman (1998). Knowledge Organizations: What Every Manager Should Know. Boca Raton, FL, St. LucierCRC Press.
Liebowitz, J. and L. C. Wilcox (1997). Knowledge management and its integrative elements, CRC.
Lim, S., A. H. B. Duffy, et al. (2001). Intelligent computational sketching support for conceptual design.
Lind, M. R. and J. M. Sulek (2000). "A methodology for forecasting knowledge work projects." Computers & Operations Research **27**(11-12): 1153-1169.
Little, A., K. Wood, et al. (1997). "Functional analysis: A fundamental empirical study for reverse engineering, benchmarking and redesign." Proceedings of the 1997 Design Engineering Technical Conferences **97**.
Luth, G. P., D. Jain, et al. (1991). "A formal approach to automating conceptual structural design, Part I: Methology." Engineering with computers **7**(2): 79-89.
Maher, M., M. Balachandran, et al. Case-Based Reasoning in Design. 1995, Lawrence Erlbaum Associates.
Marsot, J. (2005). "QFD: a methodological tool for integration of ergonomics at the design stage." Applied Ergonomics **36**(2): 185-192.
Martin, J. L. and W. Roddis Integrating qualitative and quantitative reasoning in structural engineering, ASCE.
McElroy, M. W. (2003). The New Knowledge Management: Complexity, Learning, and Sustainable Innovation, Butterworth-Heinemann.
McMahon, C., X. Meng, et al. (1995). A parallel multi-attribute transformation model of design.
Medvidovic, N., D. S. Rosenblum, et al. (2002). "Modelling software architectures in the Unified Modelling Language." ACM Transactions on Software Engineering and Methodology (TOSEM) **11**(1): 2-57.
Miles, L. D. (1972). Techniques of value analysis and engineering, McGraw-hill New York.
Mizoguchi, R. (2004). "Part 3: Advanced course of ontological engineering." New Generation Computing **22**(2): 193-220.
Mondragon, S. (2005). "Semantic Differential applied to the evaluation of machine tool design." International Journal of Industrial Ergonomics **35**(11): 1021-1029.
Moustapha, I. (2006). Concurrent Engineering In Product Design And Development, New Age International.
Munroe, A. S. (1995). "Is your design a life sentence?" machine design: 156.
Nakashima, K., H. Arimitsu, et al. (2002). "Analysis of a product recovery system." International journal of production research **40**(15): 3849-3856.
Neboit, M., E. Fadier, et al. (1993). "Analyse systémique et analyse ergonomique. Application conjointe à la reconception d'une cellule robotisée d'usinage."
Nissen, U. (1995). "A methodology for the development of cleaner products:: The ideal-eco-product approach." Journal of Cleaner Production **3**(1-2): 83-87.
Nonaka, I. and H. Takeuchi (1995). The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation. New York, Oxford University Press.
Nonaka, I., K. Umemoto, et al. (1996). "From information processing to knowledge creation: a

paradigm shift in business management." Technology in society **18**(2): 203-218.

Nonaka, I. and G. von Krogh (2009). "Tacit knowledge and knowledge conversion: Controversy and advancement in organizational knowledge creation theory." Organization Science **20**(3): 635-652.

Norm, D. (1996). "EN 1325-1-1996: Value management, value analysis, functional analysis, vocabulary-Part 1: Value analysis and functional analysis." German version EN: 1325-1321.

Norman, D. A. (1990). The design of everyday things (1st Doubleday/Currency ed.), New York: Doubleday.

Norman, D. A., C. Lewis, et al. (1986). "User Centered System Design:: New Perspectives on Human-Computer Interaction."

NZIER, N. Z. I. o. E. R. (2003). Building a case for added value through design. Report to Industry New Zealand.

O'Driscoll, M. (2002). "Design for manufacture." Journal of materials processing technology **122**(2-3): 318-321.

O Brien, C. (2002). "Global manufacturing and the sustainable economy." International journal of production research **40**(15): 3867-3877.

Obradovich, J. H. and D. D. Woods (1996). "Users as designers: how people cope with poor HCI design in computer-based medical devices." Human Factors: The Journal of the Human Factors and Ergonomics Society **38**(4): 574-592.

Otto, K. N. and K. L. Wood (2001). Product design: techniques in reverse engineering and new product development, Prentice Hall Upper Saddle River, NJ.

Owens, M. (2006). The definitive guide to SQLite, Apress.

Pahl, G. and W. Beitz (1996). Engineering Design. 2nd edition. London, Springer.

Pentti, H. and H. Atte (2002). Failure mode and effects analysis of software-based automation systems.

Pilemalm, S., P. O. Lindell, et al. (2007). "Integrating the Rational Unified Process and participatory design for development of socio-technical systems: a user participative approach." Design Studies **28**(3): 263-288.

Planchard, D. C. and M. P. Planchard (2010). Assembly Modelling with SolidWorks 2010, SDC Publications.

Pokojski, J. (2004). IPA--concepts and Applications in Engineering, Springer Verlag.

Polanyi, M. and A. Sen (1966). "The tacit dimension."

Potter, S. S., E. M. Roth, et al. (2000). "Bootstrapping multiple converging cognitive task analysis techniques for system design." Cognitive task analysis: 317-340.

PREVIEW (1995). "National Manufacturing Week '95: National Design Engineering Conference Sessions 1995." Design for Manufacturability Track, Session DFM 3.

Prouvost, P. (2004). Automatique: contr le et régulation: cours et exercices corrigés, Paris :Dunod.

Pugh, S. (1991). Total Design - Integrated Methods for Successful Product Engineering. Strathclyde, Addison-Wesley Publishers Ltd.

Pullan, W. (2000). Structure: in science and art, Cambridge Univ Pr.

Qian, X. S., J. Y. Yu, et al. (1990). "A new discipline of science—The study of open complex giant system and its methodology " Nature Magazine **13** (1): 3-10.

Qt (2011). from http://labs.qt.nokia.com/2011/09/01/qt-4-7-4-released/.

Rao, M. (2005). Knowledge Management Tools and Techniques: Practitioners and Experts Evaluate KM Solutions. Burlington,MA, Butterworth-Heinemann.

Rasmussen, J. (1983). "Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models." IEEE TRANS. SYS. MAN CYBER. **13**(3): 257-266.

Rasmussen, J. (1997). "Risk management in a dynamic society: a modelling problem." Safety science **27**(2-3): 183-213.

Redstrom, J. (2006). "Towards user design? On the shift from object to user as the subject of design." Design Studies **27**(2): 123-139.

Redstrom, J. (2008). "RE: Definitions of use." Design Studies **29**(4): 410-423.

Reich, Y. (1995). "A critical review of general design theory." Research in Engineering Design **7**(1): 1-18.

Rittel, H. W. J. and M. M. Webber (1984). "Planning problems are wicked problems." Developments in design methodology: 135-144.

Rodenacker, W. G. and J. Sch fer (1978). "Methodisches Konstruieren einer Anlage zur Herstellung von Atemkalk." Chemie Ingenieur Technik **50**(9): 669-673.

Rose, C. M. (2000). Design for environment: a method for formulating product end-of-life strategies, Citeseer.

Rosenberg, R. C. and D. C. Karnopp (1983). "Introduction to physical system dynamics." McGraw-Hill Book Co., 1983: 429.

Rosenman, M. and J. Gero (1994). "The what, the how, and the why in design." Applied artificial intelligence **8**(2): 199-218.

Roy, U., N. Pramanik, et al. (2001). "Function-to-form mapping: model, representation and applications in design synthesis." Computer-Aided Design **33**(10): 699-719.

Rubenstein-Montano, B., J. Liebowitz, et al. (2001). "A systems thinking framework for knowledge management." Decision support systems **31**(1): 5-16.

Rudolph, S. (2000). Visualization and Comparison of Solution Paths with Artificial Metrics.

Russell, S. J. and P. Norvig (2010). Artificial intelligence: a modern approach, Prentice hall.

Sagot, J. C., V. Gouin, et al. (2003). "Ergonomics in product design: safety factor." Safety science **41**(2-3): 137-154.

Sandis, C. and MyiLibrary (2009). New essays on the explanation of action, Palgrave Macmillan.

Santander, V. F. A. and J. F. B. Castro (2002). Deriving use cases from organizational modelling, IEEE.

Schön, D. A. (1987). Educating the reflective practitioner, Jossey-Bass San Francisco.

Schumann, H., H. Wendel, et al. (2010). Concurrent systems engineering in aerospace: From excel-based to model driven design.

Shimomura, Y., M. Yoshioka, et al. (1998). "Representation of design object based on the functional evolution process model." TRANSACTIONS-AMERICAN SOCIETY OF MECHANICAL ENGINEERS JOURNAL OF MECHANICAL DESIGN **120**: 221-229.

Shu-Yang, F., B. Freedman, et al. (2004). "Principles and practice of ecological design." Environmental Reviews **12**(2): 97-112.

Sieger, D. B. and R. E. Salmi (1997). "Conceptual design tool for the systematic design method." Concurrent Product Design and Environmentally Conscious Manufacturing **94**: 145–152.

Simon, H. (1969). The Sciences of the Artificial. Cambridge, MIT Press.

Siu, K. W. M. (2003). "Users' creative responses and designers' roles." Design Issues **19**(2): 64-73.

Soni, D., R. L. Nord, et al. (1995). Software architecture in industrial applications, ACM.

Sousa, C. A. A. and P. H. J. Hendriks (2006). "The diving bell and the butterfly—the need for grounded theory in developing a knowledge-based view of organizations." Organizational Research Methods **9**(3): 315-338.

Sowa, J. F. (2000). Knowledge representation: logical, philosophical, and computational foundations,

MIT Press.
Sriram, D., M. L. Maher, et al. (1985). "Knowledge-based expert systems in structural design." Computers & Structures **20**(1-3): 1-9.
Sriram, D., G. Stephanopoulos, et al. (1989). "Knowledge-based system applications in engineering design: Research at MIT." AI Magazine **10**(3): 79.
Stalker, R. (2002). "A function-behaviour-structure framework for the lifecycle of an artefact." Creativity and Cognition 4.
Stamatis, D. H. (2003). Failure mode and effect analysis: FMEA from theory to execution, Asq Pr.
Standardization, E. C. f. (1994). Activities of the European Committee for Standardisation (CEN): Note, UN.
StarUML, U. (2005). Modelling Tool. Multilingual Project. Version 5.0. 2.1570, StarUML.
Stone, H. and J. L. Sidel (2004). Sensory evaluation practices, Academic Press.
Stone, R. B. and K. L. Wood (2000). "Development of a functional basis for design." Journal of Mechanical Design **122**: 359.
Studer, R., S. Decker, et al. (2000). "Situation and perspective of knowledge engineering." Knowledge Engineering and Agent Technologies.
Suh, N. (1990). The Principles of Design. New York, Oxford University Press.
Suh, N. P. (2001). Axiomatic design, Oxford university press.
Suh, N. P. (2001). "Axiomatic Design: Advances and Applications (The Oxford Series on Advanced Manufacturing)."
Summers, J. D., N. Vargas-Hernández, et al. (2001). "Comparative study of representation structures for modelling function and behavior of mechanical devices." Proceedings of DETC2000: Computers in Engineering: 775–787.
Sun, H., R. Houssin, et al. (2010). La conception comportementale : Une nouvelle approche pour améliorer la performance de produit. 17ème Congrès de Maîtrise des Risques et de Sûreté de Fonctionnement Gentilly, France.
Sundin, A., M. Christmansson, et al. (2004). "A different perspective in participatory ergonomics in product development improves assembly work in the automotive industry." International journal of industrial ergonomics **33**(1): 1-14.
Sycara, K. P. and D. Navinchandra (1992). Retrieval strategies in a case-based design system, Academic Press Professional, Inc.
Takala, R. (2005). "Product demonstrator: a system for up-front testing of user-related product features." Journal of Engineering Design **16**(3): 329-336.
Takeda, H., T. Tomiyama, et al. (1992). A logical and computable framework for reasoning in design, Institutt for verkstedteknikk, Norges tekniske h gskole.
Tan, S. and V. Allada (2003). "Enhanced Functional Analysis for Engineering Projects."
Tay, F. E. H. and J. Gu (2002). "Product modelling for conceptual design support." Computers in industry **48**(2): 143-155.
Tessier, C. (1984). MESSAGE: un outil d'analyse ergonomique de la gestion d'un vol, ingénieur: Automatique: Toulouse ENSAE: 1984.
Tischner, U., E. Schmincke, et al. (2000). How to do ecodesign?: a guide for environmentally and economically sound design, Verlag form.
Toffler, A., H. Toffler, et al. (1995). Creating a new civilization: The politics of the third wave, Turner Pub.

Tomiyama, T. (1995). "A design process model that unifies general design theory and empirical findings." Proc. of Design Engineering Technical Conferences, ASME **2**.

Ullman, D. G. (1992). The mechanical design process, McGraw-Hill New York.

Ullman, D. G. (1997). The Mechanical Design Process, McGraw-Hill Science Engineering.

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

Umeda, Y., S. Kondoh, et al. (2005). "Development of design methodology for upgradable products based on function¨Cbehavior¨Cstate modelling." Artificial intelligence for engineering design, analysis and manufacturing **19**(3): 161-182.

Umeda, Y. and T. Tomiyama (1997). "Functional reasoning in design." IEEE expert **12**(2): 42-48.

Umeda, Y., T. Tomiyama, et al. (1995). FBS modelling: modelling scheme of function for conceptual design, Citeseer.

van Engers, T. M., R. Gerrits, et al. (2001). POWER: using UML/OCL for modelling legislation-an application report, ACM.

Van Heijst, G., R. van der Spek, et al. (1997). "Corporate memories as a tool for knowledge management." Expert systems with applications **13**(1): 41-54.

Van Schaik, A. (2001). "Design for recycling." Faculty of Civil Engineering and Geosciences, Department of Applied Earth Sciences, Section of Raw Materials Technology, TU-Delft, Netherlands.

Vancza, J. (1999). Artificial intelligence support in design: a survey, Kluwer Academic Publishers.

VDI-2222 (1982). Design engineering methodics; setting up and use of design of catalogues, VDI Society.

Vermaas, P. E. and K. Dorst (2007). "On the conceptual framework of John Gero's FBS-model and the prescriptive aims of design methodology." Design Studies **28**(2): 133-157.

Vermaas, P. E. and W. Houkes (2006). "Technical functions: A drawbridge between the intentional and structural natures of technical artefacts." Studies In History and Philosophy of Science Part A **37**(1): 5-18.

Verstijnen, I., C. Van Leeuwen, et al. (1998). "Sketching and creative discovery." Design Studies **19**(4): 519-546.

Von Stamm, B. (2003). Managing Innovation, Design and Creativity. London, John Wiley and Sons.

Walsh, N. (1998). "What is XML." XML. commune.

Wang, L., W. Shen, et al. (2002). "Collaborative conceptual design—state of the art and future trends." Computer-Aided Design **34**(13): 981-996.

Watson, I. and F. Marir (1994). "Case-based reasoning: A review." The knowledge engineering review **9**(04): 327-354.

Welch, R. V. and J. R. Dixon (1992). Representing function, behavior and structure during conceptual design.

Wiig, K. M. (1997). "Knowledge management: Where did it come from and where will it go? ." Expert Systems with Applications **13**(1): 1-14.

Wiig, K. M. (2004). People-Focused Knowledge Management. How Effective Decision Making Leads to Corporate Success. Burlington,MA, Butterworth-Heinemann.

Wiig, K. M., R. De Hoog, et al. (1997). "Supporting knowledge management: a selection of methods and techniques." Expert systems with applications **13**(1): 15-28.

Wiley, D. A. (2000). "Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy." The instructional use of learning objects **2830**(435): 1-35.

Wilkins Bert, J. and R. de Hoog (1997). "Understanding and valuing knowledge assets: Overview and method." Expert systems with applications **13**(1): 55-72.

Wixson, J. R. (1999). Function analysis and decomposistion using function analysis systems technique, Citeseer.

Wu, M. C., Y. F. Lo, et al. (2006). "A case-based reasoning approach to generating new product ideas." The International Journal of Advanced Manufacturing Technology **30**(1): 166-173.

Xu, J., R. Houssin, et al. (2011). "Fostering continuous innovation in design with an integrated knowledge management approach." Computers in industry.

Yoshikawa, H. (1989). "Design philosophy: the state of the art." Annual CIRP **38** (2): 579–586.

Youngs, R., D. Redmond-Pyle, et al. (1999). "A standard for architecture description." IBM Systems Journal **38**(1): 32-50.

Zack, M. H. (1999). Knowledge and strategy, Butterworth-Heinemann.

Zhang, W., S. Tor, et al. (2001). "A prototype knowledge-based system for conceptual synthesis of the design process." The International Journal of Advanced Manufacturing Technology **17**(8): 549-557.

Annex: List of publications

- **Publications in the international journals**:

1. R. Houssin, Sun, H., and Gardoni, M. (2011). "A Behavioural Design Approach to Improving Mechanical System Design with Integration of Use conditions" International Journal of Design and Innovation Research.

- **Communications in the international and local conferences:**

1. Sun, H., R. Houssin and Gardoni, M. (2010). L'amélioration de la performance du produit par l'intégration des tâches d'utilisation dés la phase de conception: une approche de conception comportementale, GDR MACS, 2010, Strasbourg, FRANCE.
2. Sun, H., R. Houssin and Gardoni, M. (2010). La conception comportementale : Une nouvelle approche pour améliorer la performance de produit. 17ème Congrès de Maîtrise des Risques et de Sûreté de Fonctionnement Gentilly, FRANCE.
3. Sun, H., R. Houssin and Gardoni, M. (2011). Improving Product Performance With Integration Of Using Tasks During The Design Phase: A Behavioural Design Approach. International Conference on Industrial Engineering and Systems Management. METZ FRANCE.
4. Sun, H., R. Houssin, Gardoni, M. and RENAUD J. (2012). A Behavioural Design Approach to Improving Engineering Design. CIRP Design 2012-Sustainable Product Development, Bangalore INDIA.