Knowledge-based adaptable design to support customer oriented production system of industrial equipments

Nattawut Janthong, Daniel Brissaud, Suthep Butdee

Knowledge-based adaptable design to support customer-oriented production system of industrial equipments

N. Janthong[1,2,3], D. Brissaud[1], S. Butdee[2]

[1]G-SCOP research laboratory, University of Grenoble, France

[2]Integrated Manufacturing System Research Center (IMSRC), Department of Production Engineering, Faculty of Engineering, King Mongkut's University of Technology North Bangkok (KMUTNB), Thailand

[3]Thai-France Innovation Center (TFIC), King Mongkut's University of Technology North Bangkok (KMUTNB), Thailand

Abstract

The customer is the core element of business based on current market situation, even for industrial equipment which customers' requirements are now very various. It means that the inside processes of the manufacturer should adjust to customer's requirements: the company must be very flexible among other properties. The challenge is that the flexibility comes from the management of the knowledge of the company by improving the adaptive design process. A knowledge-based design process has been developed by combining the axiomatic design and case-based reasoning approaches, to enhance the performance of the design personnel and is the key factor for the success of the company.

Keywords:

Knowledge-based, Customisation, Adaptable design

1 INTRODUCTION

In many business fields, customers have become very demanding to the point that they only ask for individualized products and services. These changes at the customer's level have forced companies to react by developing new operations strategies in order to secure market shares and improve profits. Furthermore, companies have realized that they have to concentrate on their strengths and core competencies while closely working with their partners if they want to effectively and efficiently respond to ever change or customer's requirements.

Related works can be split in two complementary approaches: Product configuration design and customization. Product configuration design tries to design a product from a selection of elements from a more or less set of pre-existing things: Modular design aims to develop product architecture consisting of physically detachable units/modules [1]; Platform design entails the identification of common attributes within a series of products [2]. Those two design strategies are producer's offer oriented but the industrial issue is now individualized customer's demand oriented. Customization is an approach to meet customers' wishes [3]. It consists of modular and platform strategies dedicated to customize major functions or aesthetic qualities. All those approaches needs to formalize a priori all that should occur in product demand. A more flexible approach to answer unexpected demand and to adapt to SMEs where resources are limited is now needed.

Design adaptability refers to the adaptability in the design of a product, so that the design can be modified and adapted to produce a new product [4]. The first advantage is that the producer can adapt the same design to different requirements and produce different products. The main objective of adaptable design is to meet customers' requirements in their specificity and personalization. The second advantage comes from that design adaptability aims at reusing the same 'design' for the creation of different products. It means that adaptive design helps reduce time to develop the new product. The adaptable design process development should make the company successful in quick response to the customer strategy. By reusing previous designs, an engineer can reduce duration and cost of development cycle and risks on product quality and performances. Moreover, the relevant and innovative information in any discipline may also be mobilized and used to update or adapt a previous design in response to changes in technology or market preferences. The challenge is that the flexibility comes from the management of knowledge of the company by improving the adaptive design process.

Hence, organizing, storing and retrieving information on previous product designs are the most important tasks in knowledge utilization to provide both designers of new products and (re)designers of existing products with the suitable solution that meets or almost meets to customer's needs.

Our motivation is to develop a methodology in (re)design of products for supported (re)design of industrial equipment based on the concept of customer centered strategy called customer-oriented production system.

It will be illustrated in the industrial warehouse truck context, where quite often products are (re)designed based on individual customer's needs. Customers' requirements are very different from one to another due to their own organization of the production lines, the physical constraints of stock areas, the workshop building and the products machined themselves. In this sense, customers' needs have become very personalized and the major factor to guide the design of such products.

Developing this approach will give a great advantage to the company against its competitors by succeeding in developing customized products and in reducing lead time by a strong reusability of designs, components and process. A knowledge based adaptation design has been developed to enhance performance of design personnel and is the key factor for the success of the company.

The paper aims at presenting the knowledge-based design process to (re)design products meeting customer's requirements. Section 2 will present the literature review then section 3 will develop the methodology proposed.

Section 4 illustrates how the system was implemented in a case study. Conclusion is given in section 5.

2 LITERATURE REVIEW

The main technique reviewed was case-based reasoning (CBR) applying on design. The basic idea of case-based reasoning is that new problems can be tackled with adapting solutions that were used to solve previous problems [5]. Case-Based Reasoning is a general paradigm for problem solving based on the recall and reuse of specific experiences. The practice shows that often it is more efficient to solve a problem by starting from a solution of a previous similar problem than to generate the entire solution from scratch. Due to the mentioned properties, CBR systems have a diversity of applications in architecture design [6], in chemical process engineering [7], and in mechanical design [8] as well as design for mass customization [9] etc. The two major research issues in CBR approaches to design are the representations of design cases and the process model for recalling and adapting design cases. Representing design cases requires an abstraction of the experience into a symbolic form that the system can manipulate. Design-case recalling involves finding a relevant design experience: it is decomposed into the subtasks of indexing, retrieving, and selecting. Indexing design cases is a critical issue in CBR approach, and CBR systems suffer from an inability to distinguish between cases if indexing is inadequate. Design-case adaptation recognizes the differences between the selected design case and the new design problem, and changes the design case so that it solves the new design problem. This process is decomposed into three steps: propose, evaluate, and modify.

As the literature review showed, case-based reasoning techniques have been investigated and the principles and technology are now mature. The concept of case based reasoning can be defined in the way to organize information or data, and this concept is applied to either 'idea', innovation or any other kinds of information that is to be stored and used somehow afterwards.

However, in industrial products designs are quite difficult to represent as a well-structured list of features. The representation of design cases requires various models of knowledge from each domain. Highly structured representations of design knowledge can be used for reasoning. However, case-based reasoning usually requires manual pre or post processing, structuring and indexing of design knowledge to identify the information needed by designers. There is a need to develop a method that clearly determines industrial products design requirements. Such a method that rigorously defines the design requirements could be axiomatic design among others.

Axiomatic design defines design as the creation of synthesized solutions in the form of products, processes or systems that satisfy perceived needs through mappings between Functional Requirements (FRs) and Design Parameters (DPs) [10]. The implementation issues were discussed by many publications [11, 12]. A fundamental aspect of the mapping process is the idea of decompositions through zigzagging. The design progresses from a higher, abstract level down to a more detailed level. These results in the formation of design hierarchies in the FRs and DPs are similar in nature to standard product functional and structural hierarchies. Thus it can identify which parts of the design structure are used to perform specific functions.

To facilitate (re)designing industrial product, this paper combines the axiomatic design and case-based reasoning approaches. The case based reasoning is used as a general framework for the reuse of design purpose and applied when a similar function is required. The axiomatic design principle is used for creating cases by analyzing existing products which FRs and DPs were decomposed. These FRs and DPs are utilized as case index and case representation in case library. It is also used for creating design database or design library by identifying relations between FRs and possible DPs of each component in design library. The information content is used for evaluating design solutions (DPs) from design library or design database composed of various components to fulfill new functional requirements which do not exist yet in the case library. The design with satisfied independence axiom provides the sequence to modify DPs in the adaptation process.

3 DESIGN METHODOLOGY

The core of the reuse design principle is structured in two main stages, preceded by the expected design description and followed by the capitalization of the work as shown in figure 1. The first stage deals with a research of similarity between the new case to be designed and all the previous cases that were capitalized in a library. The similarity search leans on the comparison of the matching of the values of the different requirements in the cases and an evaluation of the gap. If an exact matching is found, the new case is considered to be identical of the previous one and consequently that the solution of the previous case can be considered to be the solution of the new case. If no case matches the new one, the system tries to propose to designers cases that seems close to the problem to be dealt with. When one or several are considered to be close, based on designers' experience, the solution of one of those cases is considered to be the referent solution and the designers will improve it to meet the new requirements (knowledge-based adaptation of the product definition and adaptation of the business matters that are considered to be the second stage of the methodology). When no case can be considered as reference, the search strategy is unsuccessful and the designers have to manually design a new product.

The second stage deals with the adaptation of the referent solution to create the new one when a referent solution has been decided. A solution associated with a case and stored in the library, includes data about the manufacturing process and consequently on time and cost. These referent data are adapted to evaluate their values for the new case. This adaptation is very quick and enables engineers to negotiate the manufacturing contract with the customer as soon as possible and before time and cost have been deployed in the company to develop the new solution.

When a new design has been done, it is stored in the library to be capitalized and eventually reused as reference for a new design.

The design methodology has been developed based on reuse design principle afore mentioned. The methodology as shown in figure 2 is based on the assumption that the designers do not need to design products from scratch every time. They go through their ability to access to existing designs from related products and components then to revise them to fulfill specific customers' needs. The function structure and the physical structure of products from past design experiences were stored in case library. Moreover, the design library kept the designs which included components information and their function definition, which came from supplier's standard catalogue. Both case library and design library were utilized to create suitable design solutions to achieve new functionalities. Reuse case when new customers' requirements have similar functions combines with new design sub functions when the retrieved case does not have function that customer wanted. It is the basic concept for combining case based reasoning and axiomatic design principles. The process started by the comparison of new customers' requirements and constraints to function structures and physical structures of existing products that perform similar requirements and constraints. The result is that functions can be separated in product functions that have already been developed in existing designs, and add-on functions that did not exist and require to be fulfilled through the designs process. To achieve that, add-on functions are decomposed in terms of functional requirements; physical solutions are retrieved by comparison to other products of the family and by searching in designs database and standard components library. The retrieval process based on functionality and other specifications is accomplished by the aid of inference engine. Both rules and cases are necessary for the reasoning process. Then, adaptations of the design are needed to re-configure and integrate components to achieve the new design. Thus, product architecture, platforms, modules as well as functional and physical structures are the main drivers to create the case base. The adaptation process needs to follow the most suitable sequence.

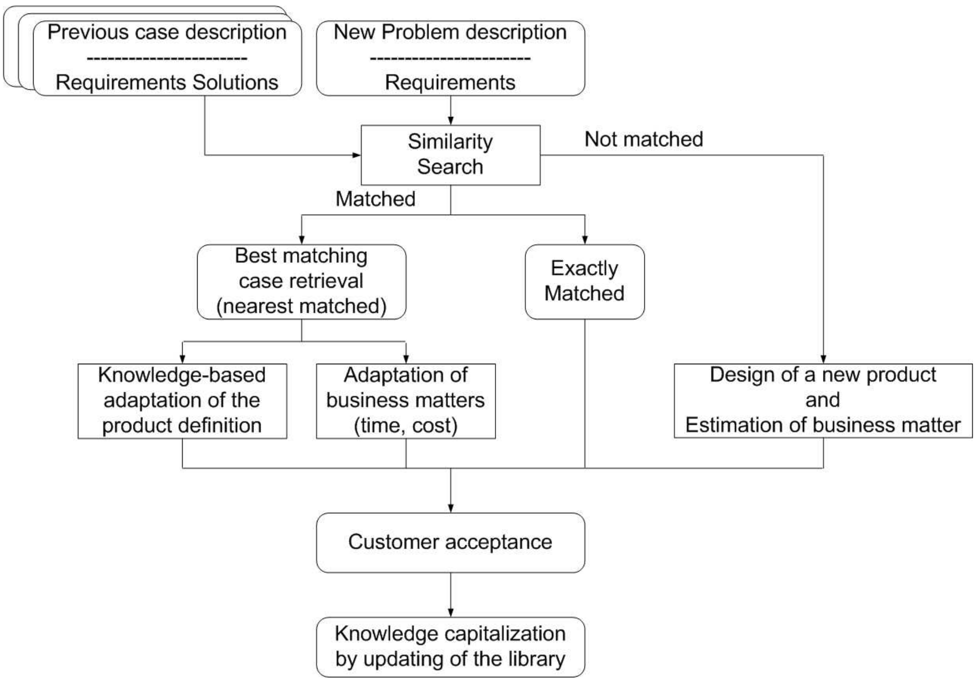


Figure 1: Design reuses principle

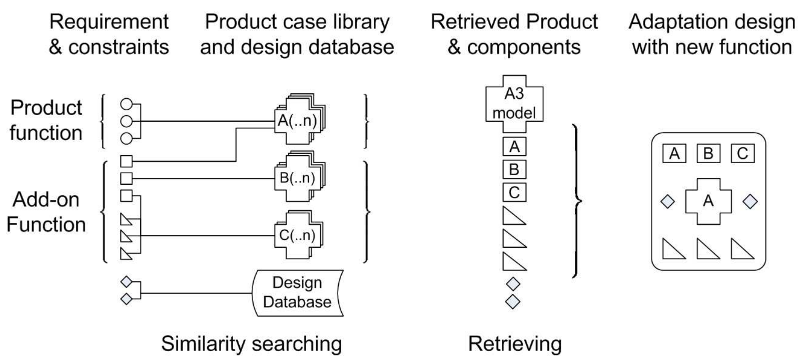


Figure 2: Design methodology

3.1 Case Representation

The basic idea is to organize specific cases, which share similar properties under a general structure. The scheme of a case consists of four parts, including customer's requirements, customers' constraints, functional requirements and design parameters. The case is represented in terms of a design hierarchy in each of the domains: functional and physical. The hierarchical structures in the FR-domain and the DP-domain correspond to customers' requirements. An advantage of this representation is that it allows a case to be accessed on its whole or by its parts when a new problem must be solved. Similar cases at appropriate levels of abstraction are retrieved from the case base and the solutions from these cases are combined and refined; the constraints can be used to guide adaptation.

3.2 Case Indexing

Case indexing involves assigning indices to cases for their quick and easy retrieval from case library. Axiomatic design decomposition principles are used to determine the indexing of both design cases and their solutions as shown in figure 3. A hierarchical case library is similar in nature to the product architecture; designers often care of designing the entire systems down to the lowest component levels that compose the systems. Thus, cases are indexed by their functions allowing a case to be retrieved in several ways.

However, it usually requires manual pre or post processing, structuring and indexing of design knowledge to identify the information needed by designers. Based on axiomatic design principles, designers map from the requirements what they want the design to be to the solutions of how the design will achieve these. As the design progresses, broad, high-level requirements are broken down, or decomposed, into smaller sub-requirements, which are then satisfied by sub-solutions.

This indexing structure scheme also allows the composition of different case pieces to create a new solution. It shows that there are many different ways to satisfy the FRs. FR skeleton sets can be generated for each of the design cases in case library. Each step down in the hierarchy represents a refinement of the unit design. It helps distinguish between cases which lead to efficient case matching and retrieving. This expresses that firms can manage single products and platforms to deliver the different products while sharing components.

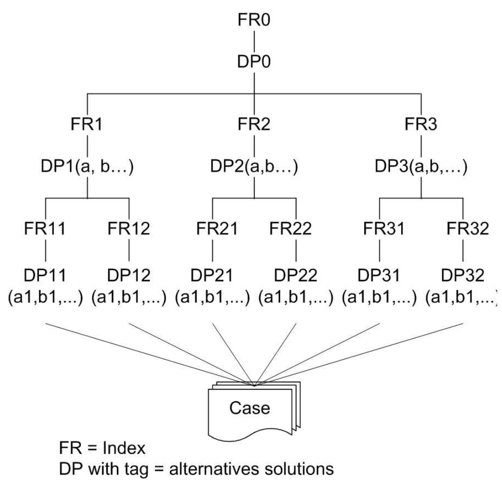


Figure 3: An index structure in case library

3.3 Case retrieval

As afore mentioned, when new customers' requirements and constraints are given, similar historical design cases are searched, matched and retrieved. The result is that two major functions are classified, namely product functions from existing products in case library that have similar functions according to problem inputs and add-on functions that are not on the retrieved existing products. Thus, the case retrieval process includes two phases – (i) similarity matching of product functions and (ii) similarity matching of add-on functions. Each phase relies on achieving two goals: finding a similar case set and finding the most similar case in this set.

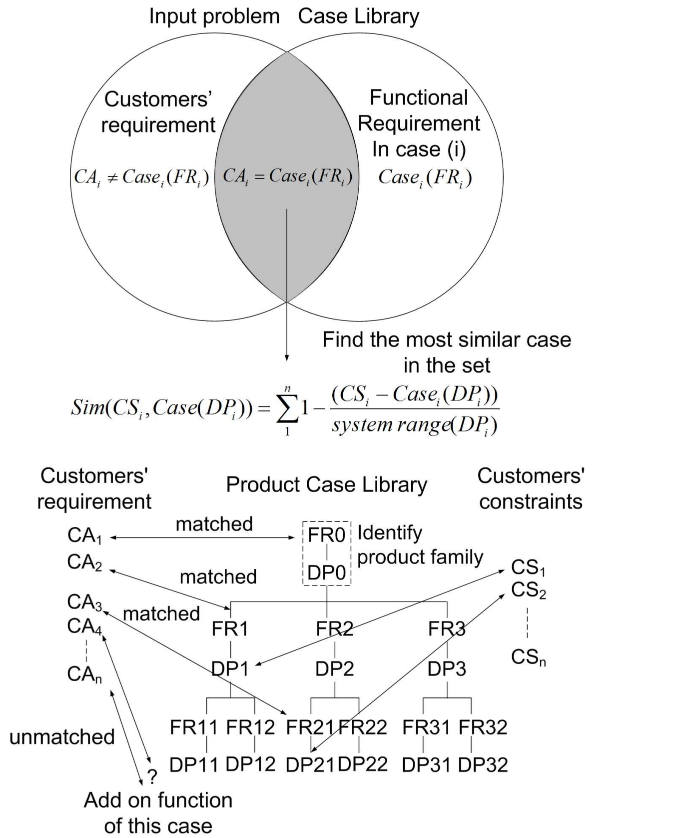


Figure 4: Case retrieval based on similarity of product function matching

In the first phase, the similarity matching of product functions as shown in figure 4, finds the similar case set from customers' requirements *(CA_i)* that are compared to product function hierarchy of each case *($Case_i(FR_i)$)*. The simplest similarity measure is to score 1 for equality and 0 for inequality as follow:

$$sim(CA_i, Case_i(FR_i)) = \begin{cases} 1 & if\ CA_i = Case_i(FR_i); \\ 0 & otherwise \end{cases}$$

Thus, a set of cases from the case base that are similar to the current input case is equal to the intersection of *(CA_i)* and *($Case_i(FR_i)$)* as follows :

$$\{P\} = CA_i \cap Case_i(FR_i) \quad ; \{P\} = set\ of\ relevant\ products$$

After all similar cases are found, a mechanism to find the most similar case in this set is needed. The input constraints *(CS_i)* are used to compare to design parameters of each retrieved case. Then, the $sim(Cs_i, Case_i(DP_i))$ can be calculated by:

$$sim(Cs_i, Case_i(DP_i)) = \sum_{1}^{n} 1 - \frac{(Cs_i - Case_i(DP_i))}{system\ range(DP_i)}$$

where: Cs_i - $Case_i(DP_i)$ is the difference between the feature values of the input and the retrieved case and *system range(DP_i)* is the range which each DP can satisfy FR based on the capacity of the producer. Then to turn a normalized distance function into a similarity measure, its value subtracts from 1. The set of cases is ranked by these similarity scores and the retrieved case is this with the highest similarity score.

In the second phase, the add-on functions (as shown in figure 5) are the customer's attributes *(CA_i)* that did not match ($Case_i(FR_i)$) in the first phase. There are two possibilities for the remaining CAs. The first one is when the function does not exist in the retrieved cases but could be in other product families: the system is called to search in the other product libraries by the same procedure as in the first phase.

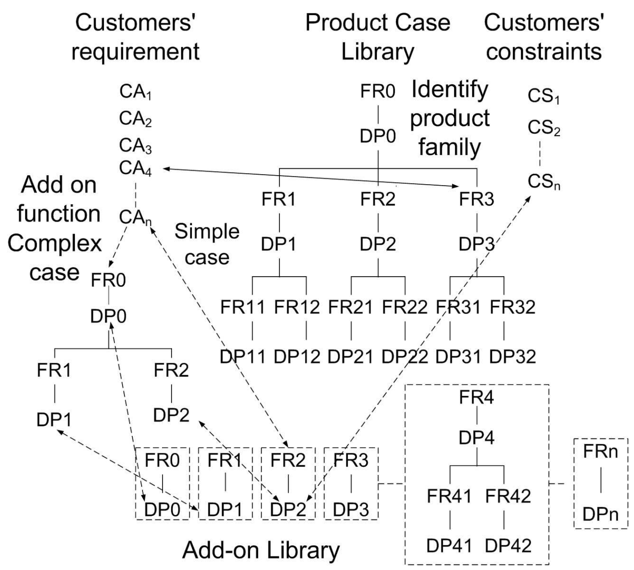


Figure 5: Case retrieval based on similarity of add-on functions matching

The second one is when the function does not exist at all in any cases of the database: the producer never did this function before for any product they did. A new design of the function of the product must be created. The add-on library and the designs database include mechanical parts, electrical parts, software modules etc. These add-on components are defined as pairs of FR and DP for single component and hierarchy of FR and DP in case of assembly components.

Similar to matching CA_i with $Case_i(FR_i)$ in the first phase, $Design_i(FR_i)$ are defined to distinguish the sources of information between case library for reuse design and design library for new design. The components in design database were evaluated to find a solution that satisfies the add-on function. If a CA corresponds to one function, a solution can be found by *sim(CA_i,$Design_i(FR_i)$)*. If a CA corresponds to a hierarchy of functions the solution of the CAs can be found by sim*($CA_i(FR_i)$,$Design(FR_i)$)*. Thus, matching a CA_i with $Design_i(FR_i)$ is as follows :

$$sim(CA_i, Design_i(FR_i)) = \begin{cases} 1 & if\ CA_i = Design_i(FR_i); \\ 0 & otherwise \end{cases}$$

or

$$sim(CA_i(FR_i), Design_i(FR_i)) = \begin{cases} 1 & if\ CA_i(FR_i) = Design_i(FR_i); \\ 0 & otherwise \end{cases}$$

The solution to satisfy each FR must be evaluated by minimizing the information content of the design based on Suh's axiomatic design principle as follows:

$$I = \log\left(\frac{system\,range}{common\,range}\right)$$

$$I_{total} = \sum_{1}^{n} I_i$$

The idea mentioned above is shown in figure 6. In the general case-base system, cases and designs experience are usually used to solve new problems by evaluating similar cases and modifying or adapting the retrieved cases. In our work, we found that designing is a complex task and it is unreasonable to expect a case base to contain all the possible design cases. It is the reason why our methodology combines case-based reasoning to initiate an appropriate design due to past experience and axiomatic design rules to provide this design with the new functions needed. This provides a combined advice that better satisfies design constraints and compatibility requirements compared to only CBR system.

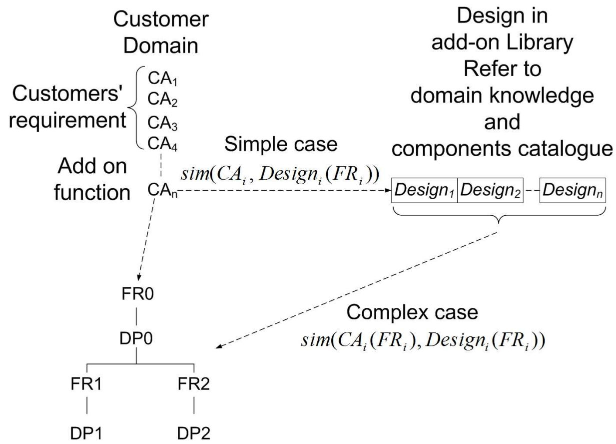


Figure 6: The design retrieval concept

3.4 Case Adaptation

If an exact matching case is found from the case retrieval process, its design can be used for the new order without any modification. Otherwise, an adaptation process is invoked to detect the discrepancies between the most similar case and the new order, and to reconcile the discrepancies by adapting the past design to the new situation.

The adaptation knowledge is usually represented as rules. The adaptation rules specify, under a certain situation, how to modify the value of a feature, or how to insert or delete certain features of the case representation in order to generate a solution for the new problem. According to axiomatic design principles when the relationships between FRs and DPs is uncoupled design, the set of adaptation rules can be easily and automatically selected by the system to make effect on similar old cases and to produce the new one. Uncoupled design occurs when each FR is satisfied by exactly one DP. The resulting matrix is diagonal and the design equation has an exact solution. The selection of adaptation rules is done easily by comparing the conflicting differences between the new problem and the current retrieved case.

In addition, the sequence of applying the adaptation rules is also important because when the design matrix is lower triangular the resulting design is decoupled, which means that a sequence exists, where the FRs can be satisfied by adjusting DPs in a certain order. This is a very important finding, as the design process is determined to a great extent by this sequence. If the case adaptation process does not follow the sequence specified by the triangular design matrix, the system appears to be very complex, which is defined as the imaginary complexity [13].

While the traditional approach of case based reasoning does not specify how to consider the sequence to adjust parameters of old case features, so there is no clear way to guaranty the correct sequence to apply adaptation rules. The axiomatic design principles can help designers to make decision in order to adapt the old case to solve the new case without randomly manner to satisfy the desired system function.

4 CASE STUDY: DESIGNING A STACKER TRUCK

4.1 The stacker trucks industrial issue

Stacker trucks are industrial warehouse trucks with a horizontal structural member supporting wheels extending forward from the main body of the trucks. They are equipped with an elevating mechanism designed to permit tiering. These trucks are designed to operate for storing and retrieving unit loads in narrow aisles. Stackers are available as walkies and in many stand-up rider designs as well.

The main performance of these trucks is their lift height. Each manufacturer has clustered their offers in a small number of classes with their own maximum lift height based on both their market knowledge and production facilities. Customers' needs are very different from one to another due to their own organization of the production lines and stock areas. Selling a stacker trucks means first of all redesigning a truck from one of the referred trucks of the company while minimizing cost by using company standard materials and devices; this redesign activity is a reconfiguration of the referred product to the customer's needs. There are many constraints for the selection and the reconfiguration of referred products to a new product to meet satisfaction of warehouse customers.

Traditionally, reconfigurative or adaptive design of stackers is performed by highly skilled design engineers. However, that still takes time to design – at least three days per job order – and the company cannot efficiently compete in the current market environment. Consequently customers' orders are lost. Moreover, an improper adaptation design leads to a higher cost and long revising times. Risks for being late to deliver the product to customers are very high.

4.2 (Re)designing product for customers' preference

Figure 7 shows the small transfer line and load configuration. The task is to move the load on dolly to conveyor then to proceed to next process. In the existing operation, one worker transports the load on dolly from the storage area and another one stands at the conveyor. At the transfer station, they work jointly to pick up the load on dolly and put it on the conveyor.

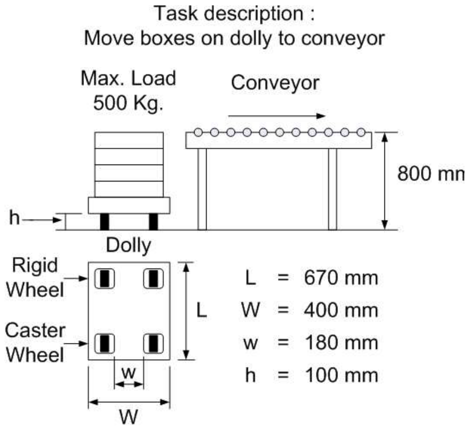


Figure 7: The transfer line and the load configuration

In order to (re)design an equipment or machine, we first need to clearly identify the customers' requirements and constraints. Case matching was proceeded by comparing customer's requirements and constraints to find the nearest case and the product model in product library which producer has been manufactured already. A product model with specification lift height 2 m. and load capacity 500 kg was matched and selected as the reference product to be adapted as shown in figure 8. This product model is battery powered for lifting and propelled manually that is rather economical to fix on floor (CA3) and exactly matches the constraints load capacity (CS2) and almost matches height constraint (CS3). The dolly configuration (CS4) and requirement CA2 do not matched at all and it means that a new design must be created.

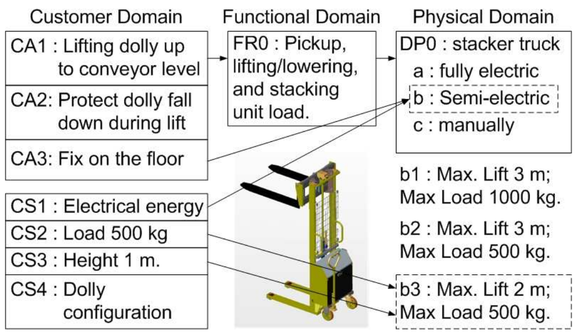


Figure 8: Selected base-product to (re)design

Then, we map CAs and constraints with product functions to identify which components modules in product library need to be (re)designed as shown in figure 9. DP1 needs to be (re)designed to fix on the floor under lift height constraint 1 m. DP2 need to (re)design fork to carry dolly based on dolly configuration and prevent dolly fall down during lift. DP3 needs to reconfigure with respect to lift height 1 m. The choice of DP4 can be selected between dc current and ac current. DP5 will be changed according to the choice DP4.

[CAs]	[FRs]	[DPs]	Components module	Constraints [CSs]	Result
CA1 : Lifting dolly up to conveyor level	FR0 : Pickup, lifting/lowing, and stacking unit load.	DP0: Semi-electric stacker truck		CS1 : electric CS2 : load 500 Kg. CS3 : Max. Lift 1 m.	(Re) design
CA2 : Fix on the floor	FR1 : Transport load in horizontal direction.	DP1: Chassis		CS2 : Max. Lift 1 m.	(Re) design
CA3 : Protect dolly fall down during lift	FR2 : carry load	DP2: For carrier & fork arms		CS4 : Dolly configuration (L,W,w,h)	(Re) design
	FR3 : Lifting load	DP3: Hydraulic cylinder & rod		CS2 : Max. Lift 1 m.	(Re) design
	FR4 : Generate force	DP4: Power unit		CS1 : electric energy	Selective
	FR5 : Supply power	DP5: Battery		CS1 : electric energy	Selective

Figure 9: Identification of components to adapt (first level)

The initial FRs and DPs were decomposed down to lower level according to zigzagging process. As shown in figure 10, FR1 went to two subs FRs namely FR11 and FR12 and FR2 to FR21 and FR22. The modules of components corresponding to DPs at this level were identified. The constraints related to FRs were also distributed to related DPs. The searching process run through library to find suitable solutions if existed otherwise design engineers were asked for (re)designing such modules.

[CAs]	[FRs]	[DPs]	Components module	Constraints [CSs]	Result
CA2 : Fix on the floor	FR11: Support structure	DP11: Leg		CS4 : Dolly configuration (L,W,w,h)	(Re) design
	FR12: Guided fork carrier	DP12: Mast		CS2 : Max. Lift 1 m.	(Re) design
CA3 : Protect dolly fall down during lift	FR21: Hold load	DP21: Fork arms		CS4 : Dolly configuration (L,W,w,h)	(Re) design
	FR22: Carry load	DP22: Fork carrier			(Re) design

Figure 10: Components during the adaption process (next level)

The design matrix of figure 9 and figure 10 are:

$$\begin{Bmatrix} FR1 \\ FR2 \\ FR3 \\ FR4 \\ FR5 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 \\ X & X & X & 0 & 0 \\ 0 & 0 & 0 & X & 0 \\ 0 & 0 & 0 & X & X \end{bmatrix} \begin{Bmatrix} DP1 \\ DP2 \\ DP3 \\ DP4 \\ DP5 \end{Bmatrix}$$

$$\begin{Bmatrix} FR11 \\ FR12 \end{Bmatrix} = \begin{bmatrix} X & 0 \\ X & X \end{bmatrix} \begin{Bmatrix} DP11 \\ DP12 \end{Bmatrix}$$

$$\begin{Bmatrix} FR21 \\ FR22 \end{Bmatrix} = \begin{bmatrix} X & 0 \\ X & X \end{bmatrix} \begin{Bmatrix} DP21 \\ DP22 \end{Bmatrix}$$

The design matrix helped cluster in two groups: the first one includes FR1, FR2 and FR3 and the second one FR4 and FR5. These two groups are independent and can be (re)designed in parallel without any effect to each other.

Moreover, the design matrix highlights the necessary decoupling of design and design engineering is then guided when (re)designing the component modules by following the sequence of design matrix to satisfy new requirements and constraints.

This example illustrated how the two approaches contribute simultaneously and complementary to the design progress: the searching process according to case based reasoning approach operates first to find the similar solutions in library the secondly design engineers are expected to (re)design or adapt by themselves from axiomatic design principles. This double approach guides design engineers to achieve (re)designing without any trials and errors, at least minimize them.

The result in (re)designing DPs to satisfy each FR is shown in figure 11. The new case was created by adapting the existing design based on the selected product model. The solutions come from two sources, namely re-produced parts and new purchased components from suppliers based on new specifications to satisfy new FR. For re-producing source, each component module was considered and re-parameterized with respect to customer's attribute and constraints. When the solution met every FR, the detail design was sent to production planning to create plans for producing parts. For re-purchasing, the engineering design team retrieved information from the design database and applied axiom2 principle to evaluate and select the one that has the minimum information content to be the solution. The components specifications were sent to the production planning through suppliers. The delivery time of suppliers is the main criteria to create the plan for end product assembly. The complete production plan and cost was updated after the information was completed. The cost and delivery date was proposed to the customer when the result was accepted by the production. Then the process of (re)designing started. However, the delivery time was

Create new case

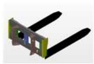
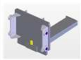
Components module		Solution	[DPs]	[FRs]
	(re) Design		DP0: Lift	FR0 :lifting/ lowing dolly.
	(re)Design CA2 : Fix on floor CS2 : Max. lift 1 m.		DP1: Chassis	FR1 : Fix on floor
	(re)Design CS4 : Dolly configuration		DP2: Fork carrier & fork arms	FR2 : carry dolly
	(re)Design CS2 : Max. lift 1 m.		DP3: Hydraulic cylinder & rod	FR3 :Lifting load
	Selected in design database CS1 : electric energy CS2 : Max load 500 kg.		DP4: Power unit	FR4 : Generate force
	Selected in design database CS1 : electric energy		DP5: AC current	FR5 : Supply power

Figure 11: the result of the (re)design process

According to customer's order characteristics, it is very often needed that manufacturers should (re)design their product for meeting customer's requirements, workplaces and environments. The manufacturer must organize their operations to support the natures of customer's orders as shown in figure12. Engineering design function is become the daily operation to work closely with production planning to achieve produced end product by customized parts and assemblies with regard to customer's preference. The existing design is retrieved, adapted and sent to relevance production stages, which include part production stage and assembly end product stage.

The parts and components specifications resulting from the new design (after the adaption process was achieved and validated) were documented. Then these documents went on to automatically create the production plan that was distributed to suppliers and related production departments. The production department could produce the parts according to the new part specifications (customized parts) and new production time and processes (adapted production plan).Suppliers produced also new parts and delivered new components from these documents. Standard parts and customized parts were assembled to customized product and tested. Many new products were produced from this process for many different configurations and specifications.

However, the engineering design department had to be involved during the realization phase of customized parts and until customized products were completed. They had to gather the actual information from shop floor about times and resources to document and update cases in library.

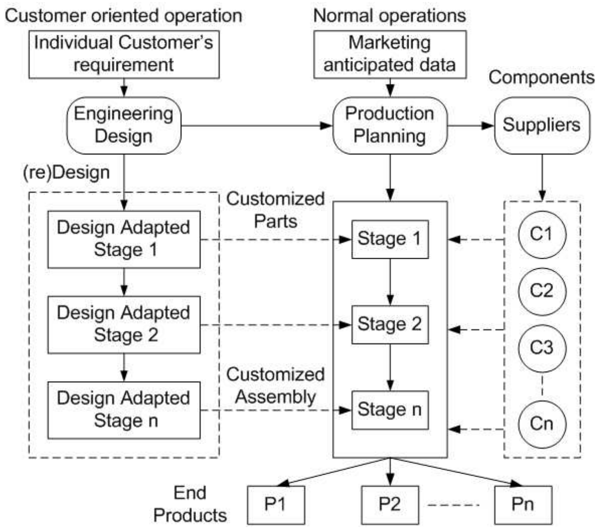


Figure 12: Mix production system for customized product

The process of responding to individual customer's requirement has become strategic to increase customer's satisfaction. The engineering design department is though the main player that lead the company answer and control the manufacturing department that could be reactive and adaptive. But the role of manufacturers is still very significant in the quality of the process. It should adjust the realized process to actual practices and resources and must give accurate feedback to the engineering department: the data quality used to design and adapt mainly depends on this feedback quality.

5 CONCLUSION

This paper has presented the concept of combining axiomatic design and case-based reasoning to assist the design process of evolving systems of industrial products. The paper illustrates how companies can react to customers' demand of industrial products in very competitive market. The company knowledge can be stored then reused and integrated with various technologies from design database and generate new functionalities for improving the existing products. Function and physical decompositions are the basic methods to represent cases and are also used to define indexes in case library as well as used to determine new designs when no case exists in case library. It also supports design engineers to achieve the adaptable design by the defined sequence of the adaptable process when the design is decoupled. However, the quality of the design solution depends on the set of FRs and DPs in the case and add-on library. Design engineers must carefully decompose the set of FRs and DPs in existing product functions and add-on component library functions that will be further reused.

6 REFERENCES

[1] Gu, P., Hashemian, M., Sosale, S. (1997). An integrated design methodology for life cycle engineering: *Annals of CIRP*,1997,Vol46/1,pp 71-74.

[2] Simpson, TW., Maier, JRA., Mistree, F. (2001). Product platform design: method and application: *Engineering Design*,Vol 13, pp 2-22.

[3] Tseng, MM., Du, X. (1998). Design by customers for mass customization products: *Annals of CIRP*, Vol 47/1, 1998, pp 103-106.

[4] Gu, P., Hashemian, M., Nee, AYC. (2004). Adaptable design: *Annals of CIRP*, 2004, Vol 53/2

[5] Watson, I., "Case-based reasoning is a methodology not a technology", Knowledge-Based Systems, Vol 12, 1999, pp. 303-308.

[6] Heylighen, A. and Neuckermans, H., "A case base of Case-Based Design tools for architecture", Computer-Aided Design, Vol33, 2001,pp.1111-1122.

[7] Avramenko, Y. and Kraslawski, A., "Similarity concept for case-based design in process engineering", Computer & Chemical Engineering, Vol 30, 2006, pp.548-557.

[8] Vong C.M., Leung, T.P. and Wong, P.K., "Case-based reasoning and adaptation in hydraulic production machine design", Engineering Application of Artificial Intelligence, Vol 15, 2002, pp.567-585.

[9] Tseng, M.M. and Jiao, A, "Case-Based Evolutionary design for mass customization", Computers and Industrial Engineering, Vol. 33, No. 1-2, 1997, pp. 319-324.

[10] Suh, N.P., The Principles of Design, Oxford University Press, 1990.

[11] Deo, H.V., and Suh, N.P., Mathematical Transforms in Design: case Study on Feedback Control of a Customizable Automotive Suspension, Annals of the CIRP, Vol53/1, 2004,pp.125-128.

[12] Goncalves-Coelho, A.M., and Mourao, J.F., Axiomatic design as support for decision-making in a design for manufacturing context: A case study, International journal of production economics, Vol.109, 2007, pp. 81-89.

[13] Suh, N.P., Complexity in Engineering, Annals of the CIRP, Vol53/1, 2004.