Expert Systems with Applications 29 (2005) 913–925

Expert Systems with Applications

www.elsevier.com/locate/eswa

Applying case-based reasoning for product configuration in mass customization environments

Hwai-En Tseng[a,*], Chien-Chen Chang[b], Shu-Hsuan Chang[c]

[a]*Institute of Production System Engineering and Management, National Chin-Yi Institute of Technology, 35, Lane 215, Section 1, Chung-Shan Road, Taiping City, Taichung County, 411 Taiwan, ROC*

[b]*Department of Industrial Design, Huafan University, No. 1, Huafan Road, Shihtin Hsiang, Taipei Hsien, Taiwan, ROC*

[c]*Department of Industrial Education, National Changhua University of Education, 1 Jin-De Road, Changhua 500, Taiwan, ROC*

Abstract

Product variation and customization is a trend in current market-oriented manufacturing environment. Companies produce products in order to satisfy customer's needs. In the customization environment, the R&D sector in an enterprise should be able to offer differentiation in product selection after they take the order. Such product differentiation should meet the requirement of cost and manufacturing procedure. In the light of this, how to generate an accurate bill of material (BOM) that meets the customer's needs and gets ready for the production is an important issue in the intensely competitive market.

The purpose of this study is to reduce effectively the time and cost of design under the premise to manufacture an accurate new product. In this study, the Case-Based Reasoning (CBR) algorithm was used to construct the new BOM. Retrieving previous cases that resemble the current problem can save a lot of time in figuring out the problem and offer a correct direction for designers. When solving a new problem, CBR technique can quickly help generate a right BOM that fits the present situation.

Keywords: Mass-customization; Product configuration; Case-based reasoning; Bill of material; Feature tree

1. Introduction

The manufacturing trend of producing a smaller number but wider variety of products forces enterprises to adopt differentiation strategy to offer customers more choices of products. Such kind of variation strategy often makes the interwoven constraint relationship of products even more complicated, which is one of the characteristics of in a customization manufacturing environment (Jiao, Ma, & Tseng, 2003; Salvado & Forza, 2004). Fohn, Liau, Greef, Young, and O'Grady (1995) once used computers as a case study and demonstrated that approximately 30–85% of product information was wrong and that this kind of mistake would causes in engineering design and substantial burden to an enterprise. Therefore, how to bring the complexity and accuracy of product configuration into control has become one of the important challenges enterprises have to face nowadays. In dealing with product configuration, it is easy to lose control of product configuration due to the incomplete communication or cognition conflict if an enterprise totally depends on the knowledge or experience of the professional personnel. This will increase the difficulty of design alternation and the pressure of cost.

Different approaches have been adopted to solve the product configuration problem. For example, the generic bill of material (GBOM) concept had been used to solve the problem of product configuration management (Hegge & Wortmann, 1991; Jiao, Tseng, Ma, & Jhou, 2000; Olsen & Saetre, 1997) and the object-oriented concept had been used to replace traditional database viewpoint (Kobler & Norrie, 1997). Constraint Satisfactory Problem (CSP) Algorithm offers another way to solve the product configuration problem (Ryu, 1999). Jiao et al. (2003) claimed that it is necessary to build a Product Family Structure (PFA), which could adjust the new product variation and satisfy

[*] Corresponding author. Tel.: +886 4 2392 4505x6001; fax: +886 4 2393 0062.

E-mail address: hwai_en@seed.net.tw (H.-E. Tseng).

0957-4174/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2005.06.026

the customer's needs. Simpson (2001) attempted to establish a product variety tradeoff evaluation method, which applied goal programming and statistical analysis techniques to optimization of product family. Du, Jiao, and Tseng (2002) dealt with product variation and flexibility by a graph method similar to a programming syntax with the viewpoint of product family design. In general, current researchers in the field mostly focus on the issues about creating information system environment and solving optimization-based problem for product family design.

As a mater of fact, the maintenance of accurate product configurations starts right after an order is placed. After the confirmation of customers, an initial product configuration can be quickly generated. With the data transmitted to R&D sector, it is sometimes necessary to redesign and reorganize these data so as to generate accurate BOM that will guarantee the smooth production procedure. If previous successful cases can be fully applied to the design alternative derived from customization, the error rate of BOM will be lowered, thus enhancing the commonality of parts of products and reducing the total cost of an enterprise. Different from traditional views of customization, the case-based database is built to solve product configuration problem. case-based reasoning (CBR) can help solve the problem through the retrieval of similar previous cases (Kolodner, 1993). In terms of product configuration, this approach has the following the advantages:

(1) It reuses the previous successful reasoning case to solve a new problem an enterprise is encountered with.
(2) Through previous successful cases, the same mistakes can be avoided and alternatives can be generated to improve the quality of problem solving.
(3) It is easy to collect previous failed or successful cases, which reduces the bottleneck of knowledge retrieval.
(4) CBR can prevent the loss of an enterprise know how when experienced technicians leave a company.

In this paper, integration of graph-based BOM tree and CBR is explored for the mass customization environment. Basic ideas regarding CBR are reviewed in Section 2. In Section 3, the proposed CBR algorithms are discussed with a ballpoint pen as an illustrated example. In Section 4, a CNC lather is used as an example to verify the mythology mentioned in this study. Finally, conclusions are made and future work is suggested in Section 5.

2. Basic concepts of case-based reasoning

There are two fundamental concepts for CBR. One is that similar problems will have similar solutions. The other is that the same problems will often occur. More importantly, CBR simulates the human problem-processing model and can have the self-learning function by constant accumulation of past experience. When the user enters a new problem in CBR, CBR will search for the data that have the highest similarity with the existing cases and adjust the previous cases to suit the new problem. General CBR algorithms are composed of the following steps (Kim & Han, 2001; Kolodner, 1993):

Step 1: Index assignment. Classify cases in the database through different features that serve as indexes.
Step 2: Case retrieval in the database. For a new problem, enter the index values for its features and compare cases to look for the one that has the highest similarity.
Step 3: Old case adaptation. Adjust the retrieved cases to fit the solution to the current status.
Step 4: New case evaluation. Evaluate the adjusted case to ensure its feasibility.
Step 5: Case storage. Store the newly adapted case in the database to achieve the self-learning function.

Generally, CBR deals with the experience previously set and turns it into a dependent one in the database for further retrieval. For a related case, the user only needs to key in the known indexes and CBR will look for a case that has the highest similarity in the database to serve for problem solution. Then, through partly adapting of the content of the retrieved case, it is possible to solve the new problem from the old experience. At last, saving the case of new problem solution in the database will reach the purpose of knowledge regeneration for future reuse.

In the past, CBR had been successfully applied to the solution to many problems. For example, on-line services to help desk application (Göker & Roth-Berghofer, 1999), scheduling and process planning (Chang, Dong, Liu, & Lu, 2000; Schmidt, 1998), hydraulic machine design (Vong, Leung, & Wang, 2002), architecture design (Heylighen & Neuckermans, 2001), customer relationship management (Choy, Lee, & Lo, 2003), fault diagnosis (Liao, Zang, & Mount, 2000; Yang, Han, & Kim, 2004), design and implementation of knowledge management (Lau, Wong, Hui, & Pun, 2003; Wang & Hsu, 2004), prediction of information systems outsourcing success (Hsh, Chiu, & Hsh, 2004), customer and market plan (Changchien & Lin, 2005; Chiu, 2002). In summary of the review, the evolution of CBR methods depends on the integration of domain problem and application specific.

Traditionally, BOM deals with a database through tables (Cunningham, Higgins, & Browne, 1996; Olsen & Saetre, 1997; Vang & Wortmann, 1992). Such a structure cannot handle the product configuration in a customization environment. In a BOM hierarchical structure, situations vary. Sometimes, only node values will be changed while sometimes a part of or even the whole structure will be changed. To solve this kind of problem, a tree hierarchy method and CBR technique are incorporated for product configuration.