Engineering Applications of Artificial Intelligence 25 (2012) 1723–1737

Contents lists available at SciVerse ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A dynamic constraint satisfaction approach for configuring structural products under mass customization

Dong Yang [a,*], Ming Dong [b], Xiao-Kun Chang [b]

[a] Department of Information Management, School of Management, Donghua University, 1882 Yan-An Road, Shanghai 200051, China
[b] Department of Operation Management, School of Management, Shanghai Jiao Tong University, 800 Dong Chuan Road, Shanghai 200240, China

ARTICLE INFO

Article history:
Received 4 August 2011
Received in revised form
5 July 2012
Accepted 17 July 2012
Available online 9 August 2012

Keywords:
Constraint satisfaction problem
Product configuration
Configuration modeling
Mass customization

ABSTRACT

Configuring structured products poses new challenges to the solving technologies for product configuration. This paper presents a novel and direct approach to encoding configuration models into the Dynamic Constraint Satisfaction Problems (DCSP). In the presented approach, components are encoded as DCSP variables while structural relationships are represented as DCSP activity constraints. Furthermore, the configuration constraints such as the requisition and exclusion constraints are treated as DCSP compatibility constraints, which allow a low-level component to join in the solving process only after its high-level component is selected in the configuration. The presented method allows a more compact encoding representation, compared to CSP and generative CSP. Experimental study shows that the presented DCSP encoding approach makes a significant improvement in the performance of product configuration.

1. Introduction

Mass customization aims to deliver an increasing product variety to satisfy diverse customer needs while maintaining near mass production efficiency (Pine, 1993). Under the mass customization paradigm, products are designed as those of consisting of modules or components where commonality and variability exist. The task of product configuration is to automatically or interactively configure a product by assembling a set of components without the violation of any constraints such that individual needs of a customer are satisfied. It has been recognized as an enabling technology for the mass customization production because the configuration processes have a significant impact on both sale-delivery processes and engineering processes of products. The application of configuration systems to the mass customization production can avoid delay and rework in the engineering processes of products, which are caused by potential errors existing in manual configuration (Soininen et al., 1998). In this way, the lead-time of tailored products can be effectively reduced. Since the successful application of the first configuration system, namely R1/XCON (Barker and O'Connor, 1989), research on product configuration technologies has received much attention (Sabin and Weigel, 1998; Stumptner, 1997). Various technologies for carrying out inferences on configuration processes have been widely studied to solve a variety of product configuration problems like configuring personal computers (Fohn et al., 1995), automobiles (Felfernig and Zanker, 2000) and communication device (Fleischanderl et al., 1998). Main technologies for configuring products include the rule-based approach (Barker and O'Connor, 1989), the case-based reasoning (CBR) (Tseng et al., 2005; Lee and Lee, 2005), genetic algorithm (GA) technique (Hong et al., 2008; Zhou et al., 2008; Li et al., 2006; Yeh et al., 2007), the constraint satisfaction problem (CSP) method (Fohn et al., 1995; Ong et al., 2006; Huang et al., 2008), and pre-compilation approach (Subbarayan et al., 2004, Subbarayan, 2005). However, those researches mainly focused on the problem-solving technologies for product configuration. In those research studies, configuration problems were regarded as flat ones where the structures of products were ignored (Huang et al., 2008; Veron and Aldanondo, 2000). To deal with the configuration problem of complex product structures, Stumptner et al. (1998) developed the approach of Generative Constraint Satisfaction Problem (GCSP) as an extension to DCSP, allowing to partially describe the structural relationships (such as is-a and part-of relationships) among components of a product. However, some problems with modeling and solving structured configuration problems still exist in GCSP approaches:

- Lacking a well-founded modeling formalism to represent structures of products.
- Encoding structured configuration problem into DCSP is indirectly carried out via port-connection ways and structure knowledge of

* Corresponding author.
E-mail address: yangdong@dhu.edu.cn (D. Yang).

0952-1976/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.engappai.2012.07.010

products is not utilized during the transformation. For example, all relationships and configuration constraints are transformed as the virtual ports and connections which are then represented as DCSP variables. This lead to a number of intermediate variables to be generated and involved in the solving processes and thus solving efficiency of the configuration process is reduced since much search space is explored.

- Configuration optimization problem, namely that there existed multiple feasible solutions and thus the optimal solution needed to be found, was not dealt with those approaches.

To overcome the first problem with the representation of configuration structures, i.e. configuration modeling, Felfernig et al. (2000) employed the Unified Modeling Language (UML) (OMG, 2004), a standard for object-oriented notations, to represent product configuration models. The advantage of object-oriented modeling approach was that reusability and maintainability of configuration models could be ensured. Nevertheless, actual configuration solving was still carried out by ILOG JConfigurator, a commercial solver developed by ILOG Corporation that implemented the GCSP algorithm (Stumptner et al., 1998).

To address the problem of structured product configuration problem, we present in this paper a novel and direct encoding approach to transforming object-oriented configuration model into DCSP. In our approach, the structural relationships are directly encoded as component variables and corresponding activity constraints by taking advantaging of the semantics of the structural relationships in the encoding process. As a result, additional virtual port variables for representing the structural relationships are no longer needed in our approach. Thus less search spaces need to be explored and the solving efficiency of configuration processes can be effectively enhanced. Furthermore, configuration constraints are encoded as DCSP compatibility constraints, instead of DCSP activity constraints as in GCSP. During the solving process, components are joined in the solving processes in accordance with strict hierarchy levels. In other words, a low-level component can only be involved in the solving process through the compositional relationship and inheritance relationship with its immediate high-level components (the aggregate or parent component). Therefore, the problem with withdrawal of a chain of previous component assignments due to the inconsistency of dynamically created components in the GCSP is overcome. Furthermore, we further extend DCSP with Branch and Bound (B&B) to handle the configuration optimization problem where multiple feasible solutions exist and the optimal solution thus needs to be found.

This paper is organized as follows. Technical background and related work are summarized in Section 2. In Section 3, configuration concepts and object-oriented configuration models are introduced. Encoding an object-oriented configuration model as a DCSP is presented in Section 4. The solving algorithm based on DCSP is described in Section 5. Experimental study and conclusion are given in Sections 6 and 7, respectively.

2. Technical background and related work

2.1. Product configuration

Given a finite set of pre-defined components, the task of product configuration is to assemble a set of components, set the attributes of components and build connections (if possible) between components to satisfy the individual requirements of a customer without violating any constraints imposed on components. These constraints are often specified due to economical, technical and processing factors.

2.2. The solving approach for product configuration

Research on product configuration systems can date back to early 1980s when R1 system (later called XCON) (Barker and O'Connor, 1989) was developed using the rule-based method. Since then, product configuration has become a commercially successful application of artificial intelligent techniques (Sabin and Weigel, 1998). Various techniques have been suggested to solve product configuration problems, including the Genetic algorithm (GA)-based approach, case-based reasoning (CBR) method, rule-based approach, CSP technique, etc. Some literature used the genetic algorithm to solve the product configuration problem. Hong et al. (2008) addressed the problem of optimal product configuration under the One-of-a-Kind production (OKP) paradigm. Variations for customizable products and parameters in the OKP product family were modeled with the AND-OR tree and parameters of nodes in this tree. They employed the genetic algorithm as the solving mechanism to obtain optimal configuration, taking customer requirements on different aspects such as performances and costs into consideration. Zhou et al. (2008) adopted the AND-OR graph to represent the configuration spaces of a customized product. The optimization of product configuration was done by means of the genetic algorithm. The objective function of optimization considered both customer preferences on product attributes, which were measured using the utility function, and costs of a product. Different from the work in Hong et al. (2008), a distinguishing characteristic in their research is that configuration constraints, such as inclusive relations and exclusive relations, are considered in the genetic algorithm for solving and optimizing product configuration problems. Similar work on the use of genetic algorithms for solving product configuration problems was also reported by Li et al. (2006) and Yeh et al. (2007). Nevertheless, the GA approach for product configuration is only suitable for the configuration problems where only a few configuration constraints exist in configuration models and thus a numerous feasible solutions can be obtained. Obviously, it is not ideal candidate for the configuration problems with complex structures of products and constraints.

A different viewpoint on product configuration is that the product configuration problem can be viewed as one of case-based reasoning (CBR) (Tseng et al., 2005; Lee and Lee, 2005). To configure a new customer order, similar previous cases are retrieved and the one with best similarity degree is recommended. For instance, Tseng et al. (2005) adopted the CBR to perform actual product configuration, aiming to reuse previous successful reasoning cases. Similar research that used the CBR for product configuration was also reported by Lee and Lee (2005). However, the CBR approach is only useful when knowledge is incomplete. Therefore, the structures of products are not supported in the CBR-based product configuration.

The rule-based reasoning is efficient in handling product configuration problems because no backtracking occurs. The earliest product configurator, namely R1 system, used the rule-based reasoning for product configuration (Barker and O'Connor, 1989). However, the weakness of R1 system lies in the problem with the maintenance of rule bases because both configuration knowledge (including product structures and constraints) and policy knowledge (namely that concerning how to solve configuration) were interweaved in rules.

Constraint Satisfaction Problem (CSP) (Tsang, 1993; Dechter, 2003) is one of well-known problem-solving technology in artificial intelligent domain. Mittal and Frayman (1989) first present that production configuration can be regarded as components and connections between components. And the connection between two components is built through the ports of the corresponding components. Port is a concrete or abstract place for connecting to or relating to other