Decision Support Systems 54 (2013) 1174–1191

Contents lists available at SciVerse ScienceDirect

Decision Support Systems

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

A graph rewriting system for process platform planning

Linda L. Zhang [a,*], Roger J. Jiao [b]

[a] IESEG School of Management (LEM-CNRS), Catholic University of Lille, 3 rue de la Digue, 59000 Lille, France
[b] The George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, 813 Ferst Drive, 30332-0405 Atlanta, GA, USA

ARTICLE INFO

Article history:
Received 26 September 2011
Received in revised form 18 April 2012
Accepted 11 November 2012
Available online 19 November 2012

Keywords:
Process platform planning
Product family
Planning automation
Graph rewriting system
PROGRES

ABSTRACT

Facilitating production process planning for product families, process platform planning (P^3) has been well recognized as an effective means of achieving production efficiency. To support decision making in P^3 automation, this study adopts graph rewriting systems to 1) organize large volumes of product and process data and 2) model production process planning reasoning. The model developed represents the structural and behavioral aspects of process platforms as family graphs and related graph transformations, respectively. In view of its modeling advantage, the system is formally defined using PROGRES. It includes meta, generic, and instance models at three different levels of abstraction. Meta models are defined for family graphs to generalize the patterns common to planning production processes for different product families; generic models are defined to describe entities pertaining to production processes of specific product families; instance models represent production processes producing product variants in a family. The graph rewriting system-based P^3 model is applied to textile spindles' production process planning. The results obtained have demonstrated its potential and feasibility to support decision making in P^3 automation.

1. Introduction

In the front end of product development, design is to convert customer requirements into a list of technical specifications, such as materials and dimensions. Based on a production process, companies produce a physical product confirming to the defined technical specifications. In this regard, production processes have major influences on production performance, such as product quality, production lead time, and cost [2,7,8,17]. A production process consists of operations, manufacturing resources carrying out operations, and operations precedence. In accordance with component items at different levels of a product hierarchy, operations can be grouped into a number of subproduction processes. Each of these subprocesses is either to manufacture a part or to form an assembly. A production process thus provides necessary input (e.g., items, machines) to downstream computer-aided process planning and computer-aided assembly planning such that the detailed process parameters (e.g., feed rate, cutting speed) can be determined for either manufacturing a part or forming an assembly [28]. (For clarity, the term: production process is used for a final product and the term: process for a component item, be it a part or an assembly.) In spite of the above important role of production processes, in practice companies plan production processes by trial-and-error based on individual 'know-how', experience, and intuition [8,22]. Consequently, it is not uncommon that for a same product, different planners come up different production processes with many variations. Decision on the production processes to be adopted is often made by the responsible planner with his subjective planning knowledge and skills [15]. Nevertheless, this trial and error-based production process planning is still viable in such manufacturing environments, where a very limited number of product variants are involved.

In manufacturing environments nowadays, companies strive to design product families, instead of single products, to satisfy various customer requirements while maintaining design efficiency and reducing design costs and time [9,11,14]. Despite the similarities in product structures and/or product items, product variants belonging to a family require different production processes due to their distinctiveness. As a consequence of the above traditional planning, companies struggle with many frequent yet unnecessary production changeovers on shop floors when producing diverse customized products. On the other hand, the key to efficiently produce product families is to maintain production to be as stable as possible [1,23,24]. Such stable production can only be achieved by eliminating the unnecessary production process variations, which are often caused by the traditional planning approaches [26,27]. This highlights the importance in developing solutions that plan production processes for customized products by considering the optimal production performance of the cohort of the family. In view of the importance of new planning solutions, a concept of process platform planning (P^3) is put forward to address production processes planning for product families such that product family production efficiency can be achieved [10,25]. The rationale of P^3 lies in the fact that it capitalizes on design similarity in planning. In line with the consensus on

* Corresponding author. Tel.: +33 1 5591 1010; fax: +33 1 4775 9375.
E-mail address: l.zhang@ieseg.fr (L.L. Zhang).

0167-9236/$ – see front matter
http://dx.doi.org/10.1016/j.dss.2012.11.011

product platforms: a product platform is an abstract structure including subsystems and their interfaces, from which a family of customized products can be derived [16], whereas a process platform refers to a common, unified structure consisting of all process elements and their relationships necessary for producing a product family (see more details in Section 2). In this regard, a process platform limits the planning solution space to a priori determined framework, thus enabling configuration and reuse of proven planning knowledge. In addition, it helps plan for necessary changes in production processes in accordance with design variations of product family members. With process changes pre-planned, companies can timely produce diverse products while ensuring the optimal utilization of available manufacturing resources and capabilities [17].

In P^3, a family of production processes (i.e., a process family) is planned to produce a product family. In this regard, a process family refers to the set of production processes producing the corresponding product variants in the product family. In a related study [26], the authors address the construction of the generic routing structure underpinning a process platform. In this study, we go a step further by providing insight in process platform planning so as to support decision making in its automation. We accomplish this by modeling process families and their planning. By modeling a process family, we attempt to show how the large volumes of data involved in a process platform should be organized. By modeling the planning process, we aim to shed light on the reasoning behind planning production processes for product families based on process platforms. By nature, such modeling involves both static representation and dynamic modeling. It, in turn, necessitates such a tool that is able to simultaneously represent system static structures and model system dynamic processes.

Unlike most modeling tools designed to either represent a system from the static aspect or model a system from the dynamic aspect (e.g., Petri nets, simulation, mathematical programming, data diagrams, flow charts), graph rewriting systems are developed to model systems by capturing both systems' constituent elements, their relationships, and system dynamic behavior [19]. Their applications have been seen in a wide range of areas, such as data structure specification, process modeling, and configuration management [21]. In a recent related study [4], the authors apply graph rewriting systems to model how product families are designed. Their work demonstrates the advantages of modeling complex systems using graph rewriting systems. Therefore, in this study, we adopt graph rewriting systems to model P^3. Furthermore, we define the proposed graph rewriting system-based P^3 in a high level, multi-paradigm specification language: PROgrammed Graph Rewriting System (PROGRES), which supports the specification of hierarchical graph schema and parametric rewriting rules for graph transformations.

The rest of the paper is organized as follows. In Section 2, the general process of P^3 and the fundamental issues are introduced. Consistent with the general process and the fundamental issues, the constructs of graph rewriting systems for P^3 are discussed in Section 3. Also discussed are the guidelines of graph rewriting system-based P^3 modeling. Sections 4 and 5 present the graph schema and graph transformations for P^3 modeling, respectively. The results of a case study are presented in Section 6 to demonstrate the potential and feasibility of the proposed model to support decision making in P^3 automation. After a further discussion of the significance of P^3, we end this paper in Section 7 by pointing out the limitations and the corresponding potential avenues for future research.

2. Process platform planning

In the literature, a product family is represented by a generic product structure (GPdS; e.g. [3]). From a design perspective, the GPdS essentially captures all data describing common and optional product components and their relationships. Similarly, a process family can be represented by a generic process structure (GPcS; [25]). As with the GPdS, the GPcS captures process data defining common and optional operations, manufacturing resources, and relationships among them. Based on the interconnections between product and process data, the GPdS and the GPcS pertaining to a product family can be integrated into one unified structure: the generic routing structure (GRS; [26]). Hence, the GRS models both the product and process data and their relationships involved in a product family. The inclusion of all these necessary data and relationships enables the GRS to act as a generic umbrella, under which production processes can be planned for product family members. Thus, planning production processes for diverse products is anchored to one platform, which helps realize the benefit of design similarity in production.

Within the GRS, P^3 entails 1) specifying product variants from the design perspective and 2) determining the corresponding production processes from the production perspective, as shown in Fig. 1. In the design view, P^3 is characterized by the GPdS involving a set of design parameters, constraints among parameters, component items, and relationships among items. The valid combinations of different parameter values define product variants (i.e., end-products). These end-products consist of specific component items, which are either primary or secondary. Unlike a secondary item, a primary item cannot be decomposed into child items, and itself is a child item of a secondary item. In this regard, an end-product is a special secondary item having no parent item. The variants of a secondary item are determined by the variations of its child items. Moreover, some product items (be they primary or secondary) are common to all product variants in the family, while some are optional and appear in several, but not all, product variants. Same as the end-product family, an item family is characterized by a set of parameters. The different combinations of parameter values are associated with different item variants. With a mechanism of parameter propagation [3], the parameter values of items are determined based on these of end-products. In other words, the parameters of end-products propagate from parent items to child items along the hierarchy of the GPdS. And the parameter values of a parent item determine these of its child items.

In the production view, P^3 is characterized by the GPcS. As a generic data structure of the process family, the GPcS is a tree involving processes, sequence relationships, operations, and operations precedence. The processes are associated with component items located in the GPdS. More specifically, a process is to produce a parent item by taking several child items as input. These processes are connected by sequence relationships (i.e., one process has to be completed before the start of another process). In accordance with the common and optional items in the GPdS, some processes are necessary to produce all product variants, whereas some are optional. Accordingly, the sequence relationships can be either fixed indicating the associated processes are necessary for all product variants, or variable suggesting the processes concerned are not involved in the production of all product variants. While these processes are abstract concepts, they are detailed by operations and operations precedence. Unlike the output of processes, the outputs of some operations are pseudo items, which cannot be found in the GPdS. As with a sequence relationship between two processes, an operations precedence demands that the following operation cannot be started without the completion of the proceeding operation. Same as the processes, some operations are necessary to all item variants in a family and some are optional. This is in accordance with the variations among item variants.

With the above understanding, planning a specific production process involves two phases: product variant specification and production process determination. With given customer requirements, a user (e.g., a designer) assigns values to parameters, resulting in a list of compatible parameter value pairs defining an end-product. These parameter values then propagate along the GPdS hierarchy, determining parameter values of component items. Each item is