



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: <http://hdl.handle.net/10985/17099>

To cite this version :

Qing XIA, Ali SIADAT, Jean-Yves DANTAN, Alain ETIENNE - Reconfigurable machining process planning for part variety in new manufacturing paradigms: Definitions, models and framework - Computers & Industrial Engineering - Vol. 115, p.206-219 - 2018

Any correspondence concerning this service should be sent to the repository

Administrator : scienceouverte@ensam.eu



Reconfigurable machining process planning for part variety in new manufacturing paradigms: Definitions, models and framework

Qing Xia*, Alain Etienne, Jean-yves Dantan, Ali Siadat

LCFC laboratory, Arts et Métiers Paristech, Metz campus, 4, rue Augustin Fresnel, 57078 Metz Cedex 03, France

A B S T R A C T

Conventional machining process planning approaches are inefficient to handle the process planning complexity induced by part variety. Reconfigurable process planning is a new process planning approach which has been well recognized as a key enabler for current manufacturing paradigms. However, in the literature, there is neither a comprehensive part variety representation model to support reconfigurable process planning nor a global solution framework to instruct the generation of the feasible process plans for a specific part variant. Therefore, this paper extends the concept of reconfigurable process planning to a concept of reconfigurable machining process planning which targets the process plan generation for a part family. A solution framework is developed for reconfigurable machining process planning. In this framework, a feature-based part variety model is proposed to represent a part family; A reconfigurable machining process plan is defined as a set of modular components which can be configured/reconfigured into the machining process plans for any part variant in the family; a novel configuration approach is proposed to generate the process plan components for a specific part variant while configuring this part variant from the family. The feasibility and effectiveness of the proposed framework and models are tested in a real case study.

1. Introduction

In order to survive in today's fierce global competition, companies are striving to provide customized products to satisfy varying customer demands. Consequently, the number of product variants offered in the industries has increased dramatically over the past few decades. Although product variety can boost companies' competitive power, this positive outcome is not always guaranteed unless variety is well-managed during all the stages of product life-cycle. Especially, when considering from part's point of view, one product usually involves a number of parts. As the number of product variants rises up, the total number of parts for these product variants multiplies. Consequently, a manufacturing system is required to handle the manufacturing complexity brought by a range of similar part variants (ElMaraghy, ElMaraghy, Tomiyama, & Monostori, 2012).

Process planning is a knowledge intensive activity that transforms product design model into manufacturing processes and determines the optimal operation sequences for production. As a connecting bridge between product design and production, process planning plays a key role in maintaining high levels of responsiveness and adaptability while propagating variety from product to process. However, in conventional machining process planning approaches, a machining process plan is

generated for one single part (Xu, Wang, & Newman, 2011). When it comes to a part family, the conventional approaches become inefficient. The process planning complexity climbs up in proportion to the number of part variants.

Two manufacturing paradigms have become prevalent in the academia and industries (Hu, 2013; Tolio et al., 2010). Mass customization aims at providing the strategies to deliver affordable product variety with near mass production efficiency. Co-evolution of product, process and production system considers the activity-'configuration' as an ideal way to realize maximal flexibility, responsiveness and adaptability with minimal costs and efforts. Both of these two paradigms ask for new machining process planning approaches to cope with part variety with lower planning complexity and higher planning efficiency (ElMaraghy et al., 2013).

In the literature, three generic technologies have been well recognized as the effective ways for handling the complexity caused by product variety:

- Modularity. It refers to the use of independent and encapsulated modules. By clearly defining the functions and interfaces of each module, ones can easily extend/change the capabilities of a system (Salvador, Forza, & Rungtusanatham, 2002).

- Platform-based technology. The term-platform indicates a high degree of commonality among a set of components and processes. It is a common architecture from which different variants can be derived (Bonev, Hvam, Clarkson, & Maier, 2015; Koren, Hu, Gu, & Shpitalni, 2013).
- Configuration technology. It is a key enabler for dealing with the combinatorial complexity of variant generation. By using configuration technology, a new variant can be generated from a set of pre-defined components instead of being designed from scratch (Zhang, 2014).

Much of current research on these three technologies has focused on managing product variety in product design domain and manufacturing system design domain. Very few studies have been reported on developing modularity, platform-based and configuration-supported approaches in process design domain. Moreover, process planning for product variety is a current trend of computer-aided process planning, it is also an inevitable demand for enabling "Industry of the Future". Hence, our interest in this paper is to extend the conventional machining process planning approaches from targeting at one single part to designing processes for a domain of part variants with the consideration of modularity, platform-based and configuration-based technologies in mind. A domain of part variants, also called a part family in this paper, refer to a part design model whose design parameters are represented by sets or intervals instead of single-values. By using this kind of design model, a part family may contain infinite design solutions. Our process planning system takes the information model of a part family as input and then generates the process plans for this part family. The proposed approach is combined with the information communication techniques of 'Industry of the Future'. For this purpose, a concept of reconfigurable machining process planning (RMPP) and a solution framework as well as its relevant definitions, information models, generation approaches are developed in this paper.

The remainder of this paper is organized as follows. Section 2 reviews the relevant research. Section 3 presents a solution framework for RMPP. Then, each part in the framework are detailed in the following sections. Section 4 presents a modular feature-based part variety model which is capable of representing both configuration-related and process-related information of a part family. Section 5 proposes the definitions, models and generation method for reconfigurable machining process plan. Section 6 introduces a configuration approach to generate the process plan elements for a specific part variant in a part family. In order to valid the proposed models and approaches, Section 7 discusses the result of a case study. Finally, the research contributions are summarized in Section 8.

2. Related work

The literature related to this research are reviewed in this section, including product/part variety management in design and manufacturing, machining process planning for part variety and representation model for part variety.

2.1. Product/part variety management in design and manufacturing

In the literature, many research efforts have been devoted into managing product/part variety in design and manufacturing. Most of the existing research embodies the philosophy of modularity, platform and configuration.

In design domain, researchers find that product variety can be properly managed by employing product family design and platform-based development (Pirmoradi, Wang, & Simpson, 2014). Salvador et al. (2002) argue that by pursuing proper type of modularity in product family design, the operational complexity of product variety can be mitigated. Hirshburg and Siddique (2014) present a platform design method using modularity for creating product variants. Vickery,

Bolunmole, Castel, and Calantone (2015) discover that modularity has a positive direct impact on product platform and both modularity and product platform has a positive direct impact on manufacturing flexibility, which then directly affects launch speed.

In addition, product configuration has been recognized as an effective solution to offer customized products (Salvador & Forza, 2004; Zhang, 2014). Product configuration refers to an activity to build customized products from a set of predefined components while complying with a set of well-defined constraints. Some researchers find that product configuration system can bring potential benefits to companies, including effective and efficient order acquisition and fulfilment process, increased sales, reduction of costs, reduction of item numbers, reduction of lead time (Haug, Hvam, & Mortensen, 2011; Hvam, Haug, & Mortensen, 2010).

In manufacturing domain, some researchers have attempted to apply modularity, platform-based technology and configuration-based technology on process platform development (Jiao, Zhang, & Pokharel, 2007), process configuration (Pitiot, Aldanondo, & Vareilles, 2014; Wang, Zhong, & Zhang, 2015) and reconfigurable manufacturing system design (Zhang, Xu, Yu, & Jiao, 2012; Koren et al., 2013).

Modularity, platform-based technology and configuration-based technology are the three generic technologies for product/part variety management. Most of current research on variety management has chosen product family as their focuses, only a minor part of research deals with process variety for part family. In addition, according to our best knowledge, no reference can be found on machining process configuration for part family.

2.2. Machining process planning for part variety

Machining process planning (MPP) consists of a sequence of planning activities that define in detail the process elements to instruct manufacturing system to remove material from raw material so as to obtain the desired form of a final part. The process elements determined by MPP include machining processes, operations, resources and relevant parameters. Many technologies from the domain of computer science have been used to automate the activities in MPP, including: (1) Feature-based technologies (Givechi, Haghghi, & Wang, 2015); (2) Knowledge-based technologies (Denkena, Shpitalni, Kowalski, Molcho, & Zipori, 2007); (3) Artificial-intelligent-based technologies (Quintanilla, Cardin, L'Anton, & Castagna, 2016); (4) Standard-compliant technologies (Xu, Klemm, Proctor, & Suh, 2006); and (5) Internet-based technologies (Wang, 2013).

A few researchers consider part variety in their approaches for MPP. A way to generate machining process plan for a new part variant is the reuse of the existing plans, such as group-technology-based approach (Burbidge, 1993) and case-based reasoning approach (Markus, Váncza, & Horvath, 1997). However, this kind of approaches is lack of flexibility and adaptability in terms of new design changes on part variants. In order to overcome the shortcoming of the existing approaches, ElMaraghy (2006) introduces a new concept of process planning approach - Reconfigurable Process Planning (RPP). It is recognized as an important enabler of changeability for evolving products and manufacturing systems (ElMaraghy, 2009). Azab, Perusi, ElMaraghy, and Urbanic (2007) propose a hybrid MPP approach for RPP. In Azab's method, a composite part is used to represent a part family and the precedence graphs of this composite part are used to represent precedence constraints in all part variants, then the precedence graph of a specific part variant is derived by modifying the precedence graph of the composite part. Although Azab's method generates the new process plan for a new part variant from the predefined precedence graph of a composite part, the generation of the precedence graph could incur a large computation burden.

2.3. Representation model for part variety

Part representation model is an information model which can be processed by a computer system. From MPP's point of view, a part representation model should be capable of describing all the necessary information for accomplishing the activities in MPP. Since it determines how a part is defined, a well-defined part representation model is essential for MPP. Feature-based model is a widely-accepted part representation model because of its modular, object-oriented and geometry-free characteristics. In the literature, most of the feature-based models are developed to represent the design information of one single part (Amaitik & Kilic, 2005; Etienne, Dantan, Siadat, & Martin, 2006).

Very few of the existing research deals with the representation model for a part family. Classification and Coding in group technology is considered as a representation approach for a part family (Tatikonda & Wemmerlöv, 1992). For classification and coding, the parts that are similar in shape or in process are grouped into a family, and then a coding system assigns a code to each part. The codes of the part family provide the necessary information for the part variants inside this family. However, classification and coding is not a direct way to represent a part family. Moreover, ones have to know the machining process before coding the parts.

2.4. Section summary

Modularity, platform-based technology and configuration-based technology are the three effective technologies for variety management. Many research efforts have been devoted to apply these three technologies for product family, while few researchers choose part family as their focuses. Being different with the conventional process planning approaches, RPP is an emerging process planning technique dedicating to a product family or a part family. Currently, there are two blocks for RPP approaches:

- Lack of representation models to structure all the necessary information for RPP.
- Lack of a solution framework to instruct that how to apply RPP in current manufacturing paradigms.

Therefore, this paper targets to a RPP approach for part family, called RMPP. A solution framework for RMPP, and the key definitions, representation models, generation approaches are developed with the consideration of modularity, platform-based configuration-based technologies.

3. A solution framework for RMPP

The aim of RMPP is to generate the feasible and optimal process plan for any specific part variant in a reconfigurable way. Being different with the conventional machining process planning methods, RMPP starts from the reconfigurable machining process plan of a part family, then the machining process plans for any part variant are configured from this reconfigurable machining process plan. Moreover, if there is any change requirement on current machining process plan of a part variant, a new machining process plan can also be reconfigured from this reconfigurable machining process plan.

Fig. 1 shows a solution framework for RMPP. In this framework, a feature-based part variety model is used to represent all the part variants of a part family. This part variety model provides the process-related design specifications to generate the reconfigurable machining process plan for a part family. It also provides the configuration information of the part family such that any part variant can be configured from this part variety model. Then, a specific part variant is configured together with its process plan elements by applying an activity, named *integrated part configuration and process plan configuration*. At last, the optimal machining process plan for the specific part variant

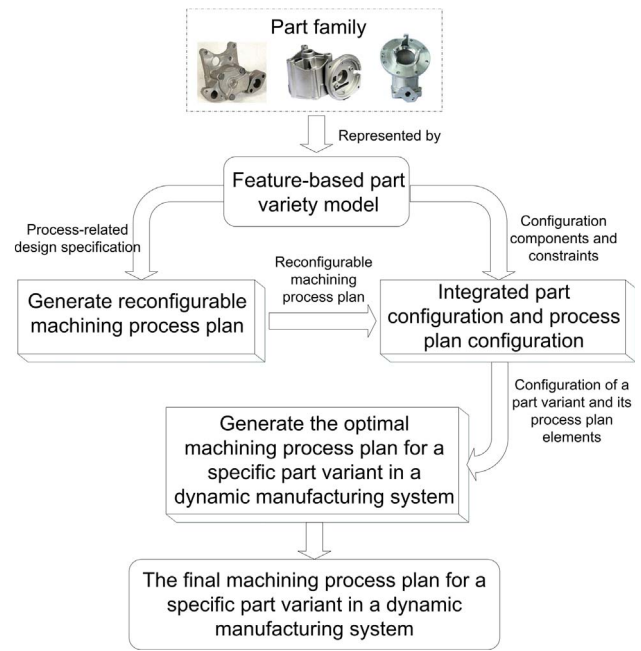


Fig. 1. A solution framework for RMPP.

is built from its process plan elements with the consideration of the constraints from a specific production system. In this way, a process planning system can respond agilely to the changes in product and production system.

The following sections refer to a feature-based part variety model and a reconfigurable machining process plan model as well as an approach to generate the reconfigurable machining process plan and a method for integrated part configuration and process plan configuration. In terms of process plan optimization, many optimization methods have been reported in the literature, therefore it is not the main focus of this paper (Chaubé, Benyoucef, & Tiwari, 2012).

4. Feature-based part variety model for RMPP

Part variety model is essential for RMPP. It represents the design information of each part variant in a part family. In comparison with the part model for one single part, the design information in a part variety model involves not only the geometrical and technical specifications of each part variant, but also the configuration relations between the variety components of all the part variants. A part variety model should also have a structure with enough flexibility and least data redundancy. This section proposes a feature-based part variety model (FPVM) in order to support RMPP. The FPVM is designed with modularity, platform-based and configuration-based thinking in mind. A definition for the concept of part family is given firstly to clarify the representation object of the FPVM. This definition extends conventional definition of a part family to infinite number of variants. Then, a general framework for FPVM is introduced and every representation element inside the FPVM is detailed.

4.1. Definitions and a general framework for FPVM

Definition 1 (Part family). A part family represents a part domain which is further decomposed into architecture and attribute sets. A part variant is derived by choosing a set of variety components from the architecture set and the values for the attributes of these components from the corresponding attribute sets.

Fig. 2 illustrates the concept of part family. The architecture provides the possible variety components for the part variants in a part

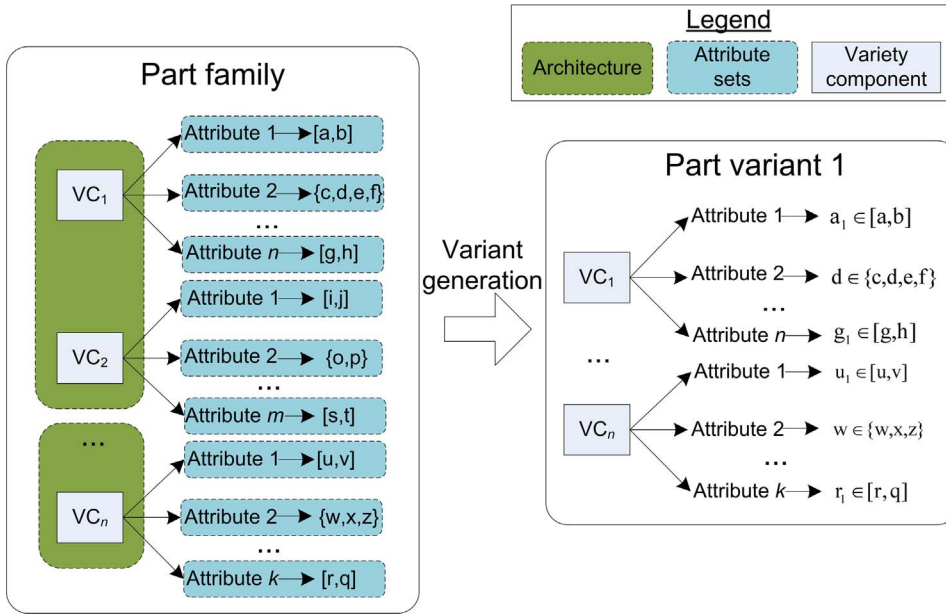


Fig. 2. Illustration of the concept of part family.

family. Variety components are either the functional components or the physical components of a part variant. Attribute sets limit the values that can be chosen for an attribute of a variety component, such as the diameter of a hole feature, the depth of a pocket feature. These sets are either finite or infinite. With the combination of different variety components and different attribute values, a part family can have different but similar part variant solutions. The number of part variants might even be infinite if there is an attribute defined on an infinite set/interval.

Definition 2 (Feature cluster). A feature cluster represents a domain of feature variants. These feature variants have the same feature type, meanwhile, they serve the same design functions in a part variant. A feature variant can be derived by assigning values to its attributes from the predefined attribute sets in a feature cluster.

A feature cluster is defined according to a form feature in feature taxonomy. A form feature is an abstract feature concept whose geometrical structure and attributes are predefined. The attribute structure of a feature cluster is inherited from a form feature, and its attribute values are defined as domains. Different feature variants in a feature cluster could have the same attribute structure but they vary on their attribute values. The domains for the attribute values of a feature cluster are either a finite set or an infinite set. If a feature cluster has an infinite attribute domain, the number of its feature variants could be infinite. Fig. 3 shows the relationships between feature variant, feature cluster and form feature.

The proposed FPVM represents three portions of information for a part family as shown in Fig. 4:

- Variety components. Variety components are composed of the function components and the physical components of all the part variants in a part family. They describe the architecture of a part family. In Section 4.2, a variety decomposition network is developed to structure the variety components of a part family at three correlative decomposition levels.
- Feature interactions. Feature interactions are the essential information for determining the feasible operation sequences. In Section 4.3, a knowledge-based representation approach is introduced for representing the feature interactions inside every part variant.
- Variety configuration constraints. Variety configuration constraints determine the feasible configurations for a specific part variant. In

Section 4.4, a propositional-logic-based scheme is developed for the representation of the configuration constraints of a part family.

4.2. Variety decomposition network and logic operators

In a part family, some part variants may share a group of common variety components, and they may also use different but similar components to achieve the same technical functions. In addition, a part variant may even have some exclusive components. Therefore, a part variety model has to be able to represent the variety components information of all the part variants in a well-structured and less redundant way. For this purpose, in the proposed FPVM, variety decomposition network is developed to represent the architecture of a part family. The architecture of a part family captures the commonalities of part variants both on functional structure and physical structure. It also describes how functional components of a part are arranged into its physical components and how these components interact. Variety decomposition network is a modular part structure with three correlative levels:

- *Function module level.* At this level, all the design functions of a part family are decomposed into function modules. Each function module represents a design function of one or multiple feature(s). According to a specific part requirement, the corresponding functional structure of a part variant is configured from the function modules at this level.
- *Feature cluster level.* In order to embrace the commonality among different part variants in a part family, the physical structure of a part family is firstly decomposed into a group of feature clusters. The feature clusters of all the part variants are structured at this level. Each feature cluster is linked to at least one feature module at feature module level. According to a specific function configuration, the corresponding feature clusters are configured from this level.
- *Feature variant level.* The feature variants in all the feature clusters are structured at this level. The physical components of a specific part variant configured from these feature variants at this level.

In variety decomposition network, there exist mappings from the components at the higher level to the components at the lower level. These mappings are either one-to-one mappings or one-to-many mappings. In part configuration, a mapping explicitly indicates that the lower-level components should be considered as the component

Fig. 3. Relationships between feature variant, feature cluster and form feature.

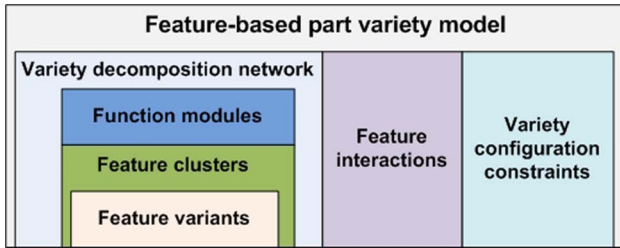
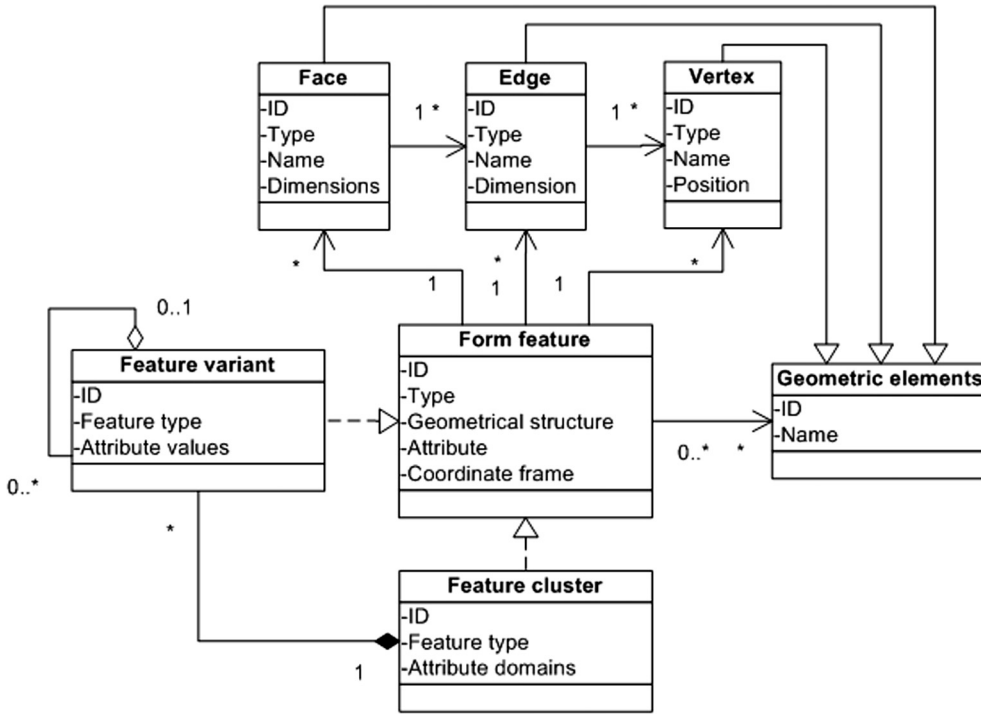


Fig. 4. General framework of the FPVM.

candidates for a desired part variant once the higher-level component has been chosen. However, the configuration relations between the lower-level components are not specified in the mapping. In order to visually express the configuration relations between variety components linked to the same upper level component, three types of logical operators are defined:

- AND operator: All the lower-level components connecting with an AND operator in a mapping must appear in the same part variant;
- XOR operator: Only one of the lower-level components connecting with this operator can appear in a part variant;
- OPTION operator: The lower-level component connecting with this operator is optional for a part variant.

Fig. 5 shows an example of a variety decomposition network. This variety decomposition network has three feature modules, six feature clusters and ten feature variants. The mappings from the components at higher level to the components at lower level are represented by using the three types of logical operators. An AND operator and an OPTION operator are applied together to express the configuration relations among the three function modules (FM_1 , FM_2 , FM_3) at function module level: a part variant can have either $\langle FM_1, FM_2 \rangle$ or $\langle FM_1, FM_2, FM_3 \rangle$ as its functional configuration. An AND operator is used for the mapping from FM_1 to FC_{11} and FC_{12} , which means once FM_1 is chosen as a functional configuration for a part variant, both FC_{11} and FC_{12} must be selected for this variant. A XOR operator is used for the mapping from

FM_2 to FC_{21} and FC_{22} , which means once FM_2 is chosen as a functional configuration for a part variant, either FC_{21} or FC_{22} must be selected, but not both. There is no need to use a logical operator if the mapping is a one-to-one mapping, for instance, the mapping from FC_{12} to FV_{12} .

4.3. Representation for feature interactions

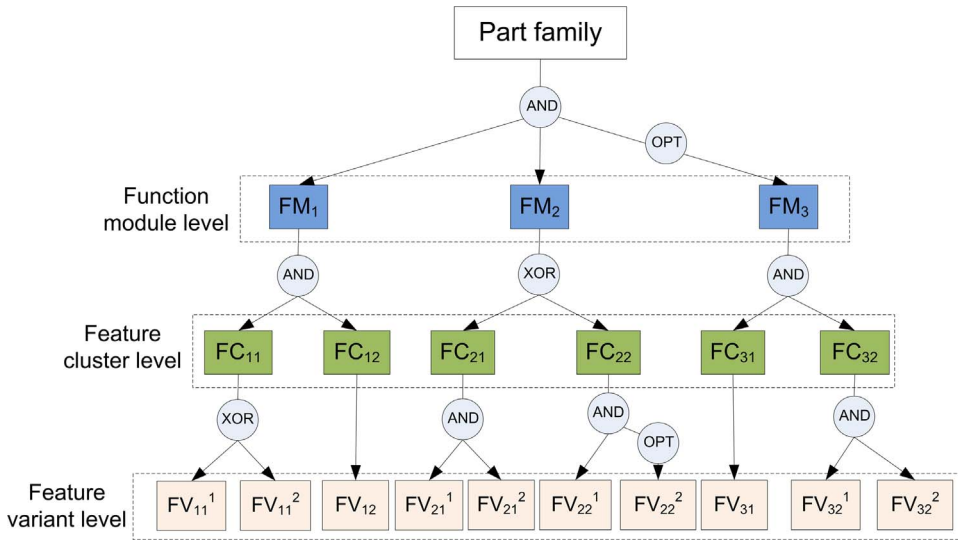
Feature interactions have great influences on the machining operation sequence for the involved features. For example, if there is a thread resides on a cylinder, then the cylinder must be machined to specifications before the thread is cut on it. Generally, feature interactions between the features on a part can be classified into two categories:

- *Tolerance/datum dependencies.* The feature interactions in this category are established by a tolerance or datum specification. In the tolerance/datum specification, some features are assigned as the datum features and other feature are constrained by the tolerance in relation to the datum feature, as shown in Fig. 6(a).
- *Topological interactions.* In this type of interactions, there are either distance relationships between features or volumetric intersections between features. Fig. 6(b) shows a case for distance relationships between features where shaft hole 1 is used as a distance reference for shaft hole 2, tapped hole 1, 2, 3, and 4. Fig. 6(c) illustrates a case for volumetric interactions between different types of features.

In terms of a part family, every part variant has its feature interaction information. Therefore, it is inefficient to represent the feature interaction information individually for each part variant. In order to maximize the benefits of commonality, in FPVM the collective feature interactions shared by all the part variants are represented at feature cluster level. The collective feature interactions between the feature variants of two feature clusters are described as an interaction between these two feature clusters with the same interaction type.

A knowledge-based representation approach is proposed to express the feature interactions in a computer interpretable way. Both the feature interactions between feature clusters and those between feature variants are represented as the facts by using the first-order logic

Fig. 5. Illustration of a variety decomposition network.



predicates. The basic geometric elements and concepts related to feature interactions are expressed by unary predications. For instance, a hole feature variant - h is described by $hole(h)$, a feature cluster - c is described by $fc(c)$, and $axis(h)$ is used to indicate the axis of a hole feature variant - h. Based on the basic unary predicates, feature interactions and the relevant relationships are represented by n-ary predicates. For example, the parallelism tolerance specification in Fig. 6(a) can be represented by using the following predicates:

$\{hasParallelism(a, b, 0.2), hole(hole1), hole(hole2), hasAxis(hole1, a), hasAxis(hole2, b), axis(a), axis(b)\}$.

4.4. Variety configuration constraints

In variety decomposition network, three types of logical operators are used to represent the configuration relations among the lower level components connected to the same upper level component. However, variety decomposition network has no mechanism to determine the configuration relations among the variety components belonging to the different upper-level components. For example, in the variety decomposition network shown in Fig. 4, a feature cluster - FC₁₁ may require another feature cluster - FC₂₁. Therefore, it is necessary to develop a representation mechanism which can be used to describe any situation of configuration constraints in a part family. For this purpose, a

propositional-logic-based scheme is defined as follows:

Definition 3. Let α_p be a proposition formula and Let $P = \{p_1, p_2, \dots, p_n\}$ be a set of atomic propositions appearing in α_p , if $P = \phi$, then α_p is valid.

Definition 4. A configuration constraint has the form: " $\alpha_{p_1} \leftrightarrow \alpha_{p_2}$ " where α_{p_1} and α_{p_2} are two proposition formulas and each atomic proposition in P_1 and P_2 corresponds to a variety component in the variety decomposition network.

Definition 5. Let $\varphi_{(\alpha_{p_1} \leftrightarrow \alpha_{p_2})}$ be an interpretation for $\alpha_{p_1} \leftrightarrow \alpha_{p_2}$ which assigns one of the truth values, (True:1) or (False:0), to every atomic proposition in P_1 and P_2 . $\tau\varphi_{(\alpha_{p_1} \leftrightarrow \alpha_{p_2})}(p_n)$ is the truth value of an atomic proposition p_n under an interpretation $\varphi_{(\alpha_{p_1} \leftrightarrow \alpha_{p_2})}$.

Definition 6. An interpretation $\varphi_{(\alpha_{p_1} \leftrightarrow \alpha_{p_2})}$ is a variety configuration for the variety components in a configuration constraint $\alpha_{p_1} \leftrightarrow \alpha_{p_2}$ if and only if it satisfies $\alpha_{p_1} \wedge \alpha_{p_2}$, that is the truth value of $\alpha_{p_1} \wedge \alpha_{p_2}$ under $\varphi_{(\alpha_{p_1} \leftrightarrow \alpha_{p_2})}$ is True.

For example, assume that in the variety decomposition network shown in Fig. 5, there is a configuration relation among the feature variants, FV₁₁¹, FV₂₂¹ and FV₂₂²: if FV₁₁¹ is selected in a configuration, then FV₂₂¹ should be chosen and FV₂₂² should not be chosen. According to the definitions above, this configuration relation can be expressed as the following propositional formula:

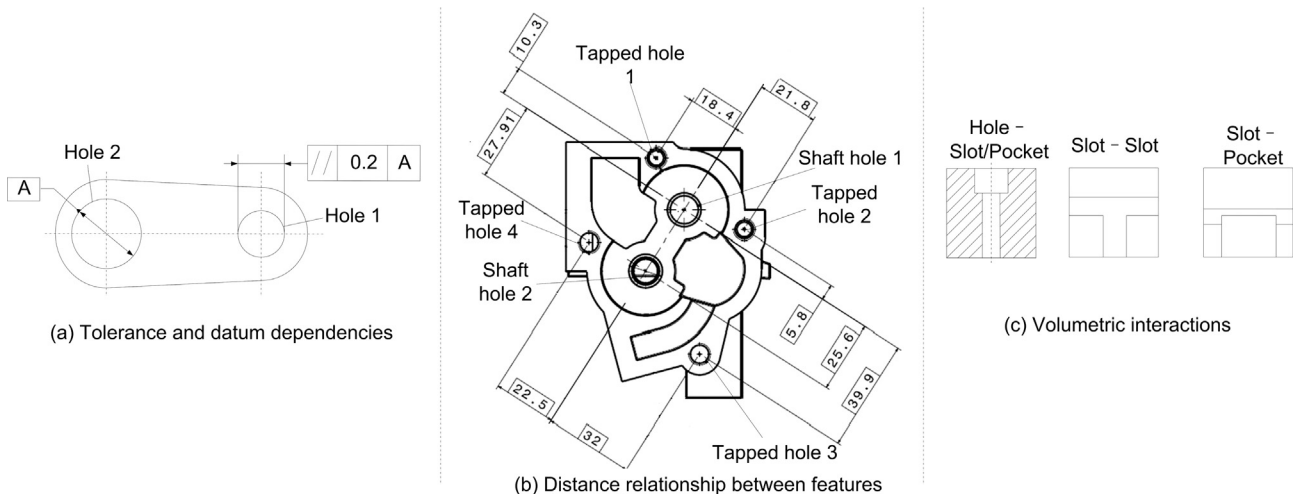


Fig. 6. Different types of feature interactions in a part.

$$FV_{11}^1 \leftrightarrow FC_{22}^1 \wedge \overline{FC_{22}^2},$$

where $FV_{11}^1, FC_{22}^1, FC_{22}^2$ are the atomic propositions in this constraint. An interpretation, $\{FV_{11}^1 = 1, FC_{22}^1 = 1, FC_{22}^2 = 0\}$, of this formula is the variety configuration.

5. Reconfigurable machining process plan model and the generation method

In feature-based machining process planning, machining operations and their sequences for each machining feature are firstly determined, and then the machining process plans are generated by sequencing the machining operations of all the machining features on the part with the consideration of the sequencing constraints, including feature interactions, set-up constraints, tool accessibility as well as some good practice. In this section, a new concept, reconfigurable machining operation plan, is introduced, and then a reconfigurable machining process plan is defined as a set of reconfigurable machining operation plans together with a set of precedence constraints between the interactive variety components. At last, a two-step method to generate a reconfigurable machining process plan is proposed.

5.1. Reconfigurable machining operation plan

Definition 7 (Reconfigurable machining operation plan). Reconfigurable machining operation plan (RMOP) consists of a set of similar machining operation plans that satisfy all the machining requirements of a feature cluster.

In a RMOP, similar machining operation plans are a set of machining operation sequences sharing a part of machining operations and a part of precedence sequences. A RMOP for a feature cluster contains all the feasible machining operation plans for any feature variant in this feature cluster.

According to Definition 2, a feature cluster is a domain of feature variants. The design specifications of a feature cluster are described as domains which are either intervals or sets. A machining operation plan is a sequence of different machining operations; the machining capabilities of a machining operation plan depend on the capabilities of the last operation in the operation sequence, whose values are represented as either sets intervals or sets. Based on this common representation, a rule is defined to determine the condition where the machining capabilities of a machining operation plan satisfies the machining requirements of a feature cluster:

Rule 1. A machining operation plan satisfies all the machining requirements of a feature cluster, if and only if its machining capabilities and the corresponding machining requirements of this feature cluster meet:

$$\forall r \in R, \exists c \in C. \min(c) \leq \min(r),$$

where

R is a set of machining requirements of a feature cluster,
 r is a machining requirement,
 C is a set of machining capabilities of a machining operation plan,
 c is a machining capability,
 $\min(c)$ and $\min(r)$ are the minimum values of c and r respectively.

As shown in Fig. 7, a RMOP is represented as a directed graph $G(V,E)$, where V is the set of nodes and E is the set of ordered pairs of nodes called directed edges. The directed graph has the following properties:

- Each node in V is a machining operation;
- The directed edge written as (Op_1, Op_2) expresses that operation Op_1 precedes operation Op_2 ;

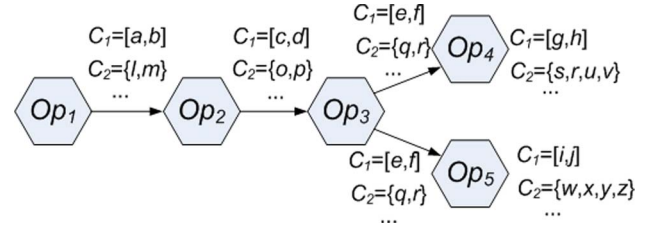


Fig. 7. Illustration of a RMOP.

- There exists at least one starting node in the graph. A node Op is a starting node if there is a directed path from Op towards other nodes of G , and no directed path to Op ;
- There exists at least one ending node in the graph. A node Op is an ending node if it does not precede any other nodes;
- A machining operation sequence for a feature variant is a directed open path starting from a starting node and ending with the node satisfying all the design specifications of this feature variant.
- The capabilities of the machining operations are labelled in the directed graph and their values are represented by using intervals or sets.

5.2. Reconfigurable machining process plan

Definition 8 (Reconfigurable machining process plan). A RMPP is a 2-tuple $\langle \Psi, \Omega \rangle$, where

- Ψ is a set of 2-tuples $\langle fc, G_{fc} \rangle$, where
 - fc is a feature cluster in the part family,
 - G_{fc} is a RMOP for the feature cluster fc represented as a directed graph;
- Ω is a set of precedence relationships:

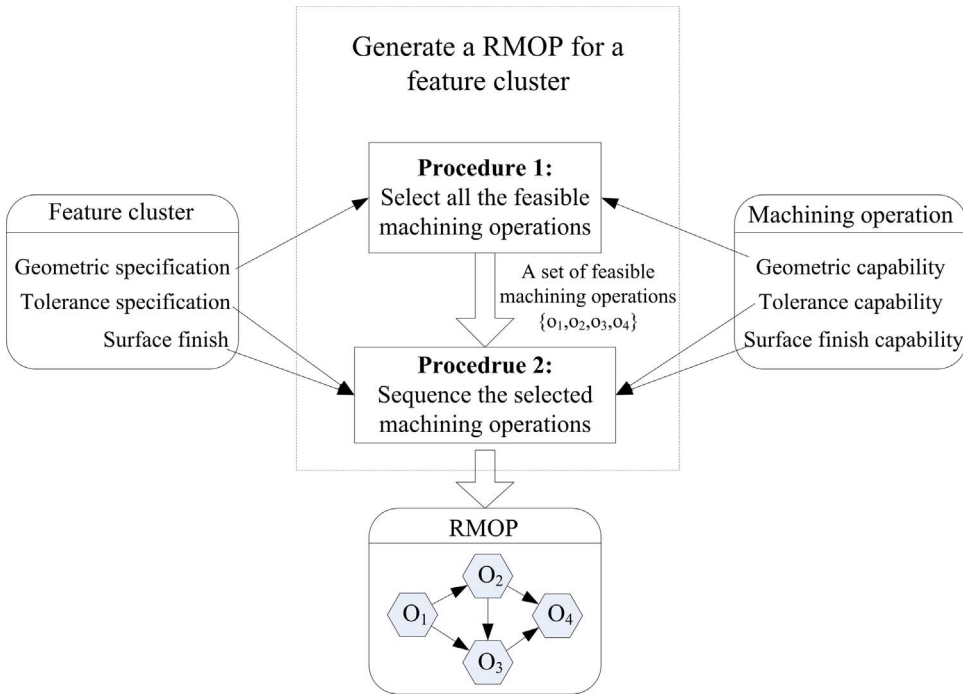
$$\Omega = \{pr(a,b) | a,b \text{ is a feature cluster or a feature variant, } pr \in \{\text{softBefore, hardBefore, softImmeBefore, hardImmeBefore, equal}\}\}.$$

Reconfigurable machining process plan (RMPP) consists of the RMOPs corresponding to all the feature clusters in the variety decomposition network of a part family and the precedence relations between the interactive part variety components. Five types of precedence relations between the interactive part variety components are defined for RMPP:

- $\text{softBefore}(a,b)$: the machining of a has to be started before the machining of b ;
- $\text{hardBefore}(a,b)$: the machining of a has to be ended before the machining of b ;
- $\text{softImmeBefore}(a,b)$: the machining of a has to be started immediately before the machining of b so that there is no other feature can be machined between a and b ;
- $\text{solidImmeBefore}(a,b)$: the machining of feature variant a has to be ended immediately before the machining of b so that there is no other feature can be machined between a and b ;
- $\text{equal}(a,b)$: a and b have to be machined simultaneously.

A reconfigurable machining process plan is incomplete process plan in terms of the feasible process plan of a part variant, but it contains all the process plan elements for the feasible process plans of any part variant in the part family. It can be prepared before knowing the specific configuration of a part variant such that the process planning efficiency and flexibility are improved.

Fig. 8. A matching process for RMOP generation.



5.3. A generation method

According to Definition 8, a RMPP has two parts, the RMOPs and the precedence relations. Therefore, the generation of a RMPP is decomposed into two steps:

- Step 1 - generating the RMOP for each feature cluster from the variety decomposition network of a part family. In this step, the design specifications of each feature cluster are analysed by a RMOP generation mechanism. The principle is to match the machining capabilities of machining operations with the corresponding machining requirements of the feature cluster. The matching process has two procedures as shown in Fig. 8:
 - In the first procedure, all the feasible machining operations are chosen from a set of machining operation candidates. The feasible machining operations selected in this procedure are geometrically feasible after comparing their geometric capabilities with the geometric specifications of the feature cluster. The existing techniques for machining operation selection can be adopted to achieve this step, such as the neural network based method proposed by Deb, Ghosh, and Paul (2006).
 - In the second procedure, these feasible machining operations are sequenced according to their tolerance capabilities and surface finish capabilities. The principle of this procedure is to build an operation sequence by constantly adding more capable operations into current sequence until all the machining requirements are satisfied. The operation sequencing should consider that both the capabilities of operations and the specifications of feature clusters are represented by intervals or sets.
- Step 2 - generating the precedence relation for interactive variety components. In this step, these precedence relations are generated through analysing the feature interactions between the interactive part variety components. As mentioned in Section 4.3, in FPVM feature interactions are represented by using a knowledge-based representation approach, an expert system can be used to automatically infer these feature interaction relationships for the precedence relations.

Both of these two steps are performed before a specific part variant

is configured. In the solution framework mentioned in Section 3, the RMPP for a part family is then used by an activity called integrated part configuration and RMPP configuration which generates the configuration of a part variant together with the modular process plan elements for this part variant; after that, when the part variant is scheduled for production, these modular process plan elements are used to build the final machining process plan according to the specific scenario of a manufacturing system.

6. Integrated part configuration and RMPP configuration

In the proposed solution framework, integrated part configuration and RMPP configuration generates the configuration of a part variant together with the modular process plan elements for this part variant. An IDEF0 diagram for this activity is shown in Fig. 9. The input is a set of function requirements for a part variant. The outputs include the configuration of a part variant and the modular machining process plans for this part variant.

Integrated part configuration and RMPP configuration is modeled as a dynamic constraint satisfaction problem (DCSP) (Yang, Dong, & Chang, 2012). It is a triple $I = \langle V, D, C \rangle$ where:

- $V = \{V_{fm}, V_{fc}, V_{fv}, V_{rmop}, V_{pr}\}$ is the set of variables. V_{fm} corresponds to

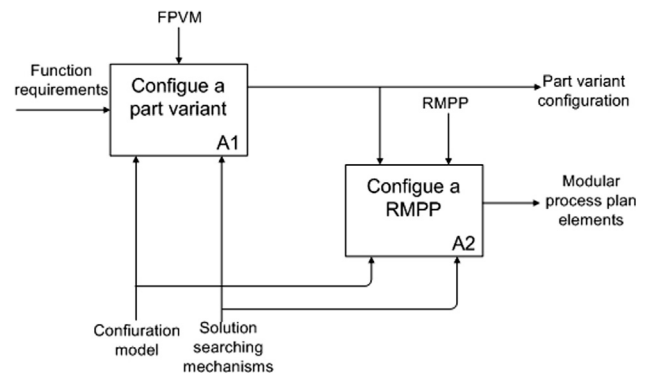


Fig. 9. IDEF0 diagram for integrated part configuration and RMPP configuration.

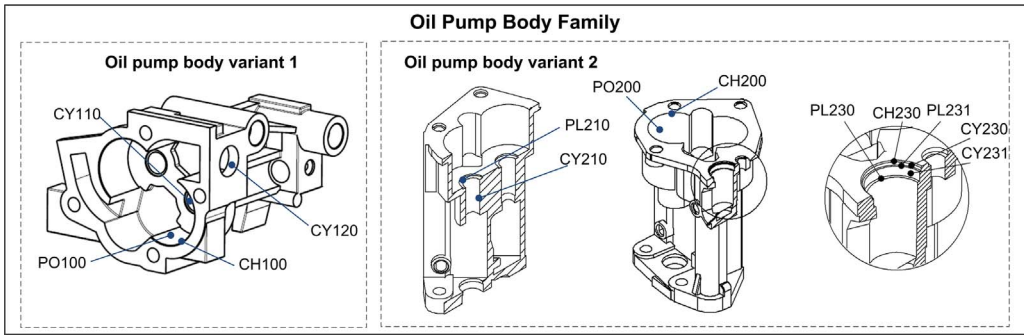


Fig. 10. Two part variants in an oil pump body family.

the function modules in the FPVM, V_{fc} corresponds to the feature clusters, V_{fv} corresponds to the feature variants, V_{rmop} corresponds to the RMOPs in the RMPP, and V_{pr} corresponds to the precedence relations in the RMPP.

- $D = \{0,1\}$ is a Boolean domain for all the variables in V . If the variables in V take the value of 0, then the variety components corresponding to these variables are not chosen in the final configuration; otherwise, the variables take the value of 1, which means that the variety components are chosen in the final configuration.
- $C = \{C_c, C_a\}$ is a set of constraints which are divided into two subsets:
 - C_c consists of compatibility constraints. The compatibility constraints in C_c define the selectivity relations among the variety components of a part family. These compatibility constraints are determined by the variety configuration constraints defined in the FPVM and the mapping relations between the variety components and the process plan elements.
 - C_a is a set of activity constraints. The activity constraints in C_a describe the conditions under which a variable may or may not be activated as a part of final solution. These activity constraints come from the mapping relations among the variety components in variety decomposition network. During the problem solving process, these constraints guarantee the process to always explore the domains of activated variables.

This dynamic constraint satisfaction problem is solved by using a constraint logic programming system. The problem solving process starts with a set of initial variables and the value assignments on these variables. Other variables are activated during the solving process as soon as their activity constraints are satisfied. With this dynamic mechanism, the searching for the irrelevant variables are avoided. The outcomes of the problem solving process are the functional and physical configurations of a specific part variant as well as the RMOPs and the precedence relations for the feasible process plans of this variant.

7. A case study

The proposed models and approaches are applied on a real case - an oil pump body family. This oil pump body family is a sub-assembly family of an oil pump family. In order to simply this case, two part variants of this oil pump body family are shown in Fig. 10, and partial design information of this part family is shown in Fig. 11. The aim of this case study is to show the feasibility and the effectiveness of the proposed representation models as well as how they works in the proposed solution framework.

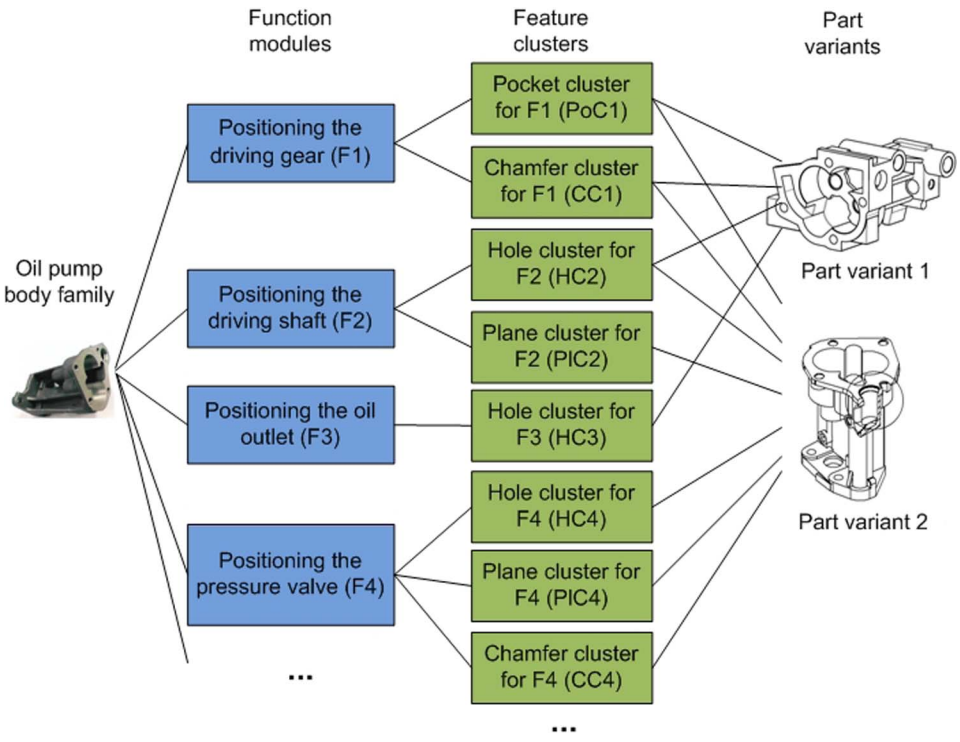


Fig. 11. Design information of an oil pump body family.

Table 1
Feature clusters of the oil pump body family.

Pocket cluster for F1 (PoC1)	Chamfer cluster for F1 (CC1)	Hole cluster for F2(HC2)	Plane cluster for F2 (PIC2)	Hole cluster for F3 (HC3)	Hole cluster for F4 (HC4)	Chamfer cluster for F4 (CC4)	Plane cluster for F4 (PIC4)
PO100,PO200	CH100,CH200	CY110,CY210	PL210	CY120	CY230,CY231	CH230	PL230,PL231

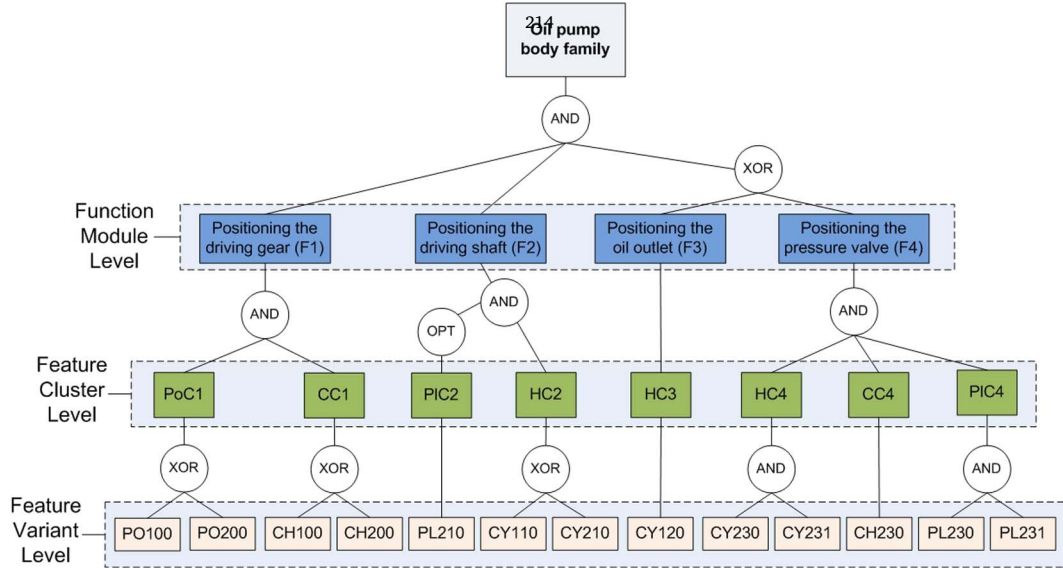


Fig. 12. Variety decomposition network for the oil pump body family.

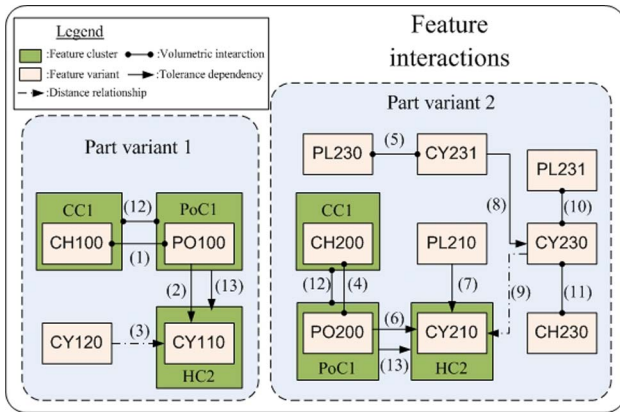


Fig. 13. Feature interactions between the interactive part variety components.

7.1. Building feature-based part variety model

7.1.1. Building variety decomposition network

According to the design information shown in Figs. 10, 11 and the mapping relations between the feature variants and the feature clusters listed in Table 1, the variety decomposition network for this part family is established as shown in Fig. 12. In this variety decomposition network, the function modules, feature clusters and the feature variants of the part family are structured into the three decomposition levels. The configuration relations between variety components linked to the same upper level component are visually expressed by the three types of logical operators.

7.1.2. Representing the feature interactions

As mentioned in Section 4.3, the collective feature interactions for all the part variants are represented at feature cluster level. For this oil pump body family, two feature interactions are defined at feature

Table 2
Predicate-based representations for the feature interactions in Fig. 14.

Code	Type	Feature variants	Feature interaction
(1)	VI	CH100, PO100	$startIn(ch100, po100)$
(2)	TD	PO100, CY110	$\{hasCoaxiality(a, b, \emptyset 0.2), hasAixs(po100, a), hasAixs(cy110, b)\}$
(3)	DR	CY120, CY110	$\{hasDistance(cy120, cy110, (10,15,30)), datum(cy110)\}$
(4)	VI	CH200, PO200	$startIn(ch200, po200)$
(5)	VI	CY231, PL230	$cross(cy231, pl230)$
(6)	TD	PO200, CY210	$\{hasCoaxiality(e, f, \emptyset 0.5), hasAixs(po200, e), hasAixs(cy210, f)\}$
(7)	TD	CY210, PL210	$\{hasPerpendicularity(f, pl210, 0.3)\}$
(8)	TD	CY231, CY230	$\{hasCoaxiality(h, i, \emptyset 0.8), hasAixs(cy231, h), hasAixs(cy230, i)\}$
(9)	DR	CY230, CY210	$\{hasDistance(cy230, cy210, (20,25,32)), datum(cy210)\}$
(10)	VI	PL231, CY230	$cross(cy230, pl231)$
(11)	VI	CH230, CY230	$startIn(ch230, cy230)$
(12)	VI	CC1, PoC1	$startIn(cc1, poc1)$
(13)	TD	HC1, PoC1	$\{hasCoaxiality(j, k, [\emptyset 0.1, \emptyset 0.5]), hasAixs(poc1, j), hasAixs(hc2, k)\}$

*VI: Volumetric interaction; TD: Tolerance dependency; DR: Distance relationship.

Table 3
Variety configuration constraints for the oil pump body family.

Variety configuration constraints	Description
$(f1 \wedge f2 \wedge f3 \wedge \bar{f4}) \vee (f1 \wedge f2 \wedge \bar{f3} \wedge v_{f4}) \leftrightarrow \text{true}$ $f1 \leftrightarrow poc1 \wedge cc1; f2 \leftrightarrow (plc2 \wedge hc2) \vee (\bar{plc2} \wedge hc2);$ $f3 \leftrightarrow hc3; f4 \leftrightarrow hc4 \wedge cc4 \wedge plc4; plc4 \leftrightarrow pl230 \wedge pl231;$ $poc1 \leftrightarrow (po100 \wedge \bar{po200}) \vee (\bar{po100} \wedge po200); plc2 \leftrightarrow pl210;$ $cc1 \leftrightarrow (ch100 \wedge \bar{ch200}) \vee (\bar{ch100} \wedge ch200); hc3 \leftrightarrow cy120;$ $hc2 \leftrightarrow (cy110 \wedge \bar{cy210}) \vee (\bar{cy110} \wedge cy210); cc4 \leftrightarrow ch230.$	Constraints between the upper level components and their lower level components
$plc2 \leftrightarrow hc4 \wedge cc4 \wedge pl4; po100 \leftrightarrow ch100;$ $po100 \leftrightarrow cy110; po100 \leftrightarrow cy120; po200 \leftrightarrow pl210;$	Constraints among the lower level components.

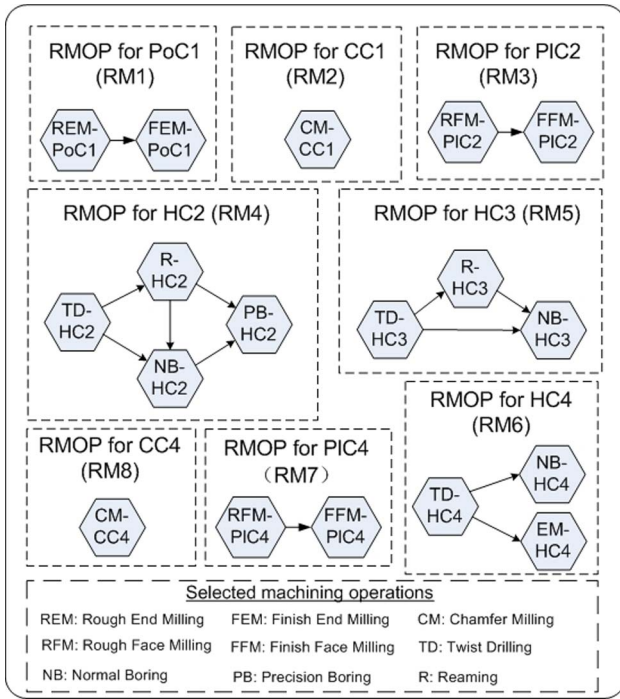


Fig. 14. RMOPs for the feature clusters represented by directed graphs.

Table 4
Design specifications of a feature cluster - HC2.

Position (R_p)	Orientation (R_o)	Surface roughness ($R_{sr}/\mu\text{m}$)	Geometrical specification		Tolerance		
			Diameter (R_{dia}/mm)	Depth (R_{depth}/mm)	Dimensional tolerance (R_{dt})	True Position (R_{tp}/mm)	Cylindricity (R_{cy}/mm)
{(0,0,0)}	{(1,0,0)}	[0.2,3.2]	{ \emptyset 10, \emptyset 15}	{50,81}	{IT6,IT8}	[0.2,0.6]	[0.01,0.02]

Table 5
Machining capabilities of the hole making operations.

	Twist drilling (O^{td})	Reaming (O^r)	Normal boring (O^{nb})	Precision boring (O^{pb})
Tool diameter/mm	[0.5,88.9]	[1.6,101.6]	[9.6,304.8]	[9.6,304.8]
Depth/Dia limit	12	16	6	6
Surface roughness/ μm	[1.6,6.3]	[0.4,3.2]	[0.2,6.3]	[0.2,6.3]
Dimensional tolerance/IT	{9...15}	{7...10}	{7...13}	{6...10}
True position/mm	[0.05,0.5]	[0.05,0.3]	[0.03,0.2]	[0.003,0.2]
Cylindricity/mm	[0.1,0.5]	[0.01,0.1]	[0.01,0.1]	[0.001,0.01]

cluster level. They are a coaxiality between PoC1 and HC2 and a volumetric relationship between PoC1 and CC1. The other feature interactions for this part family are represented at feature variant level. Fig. 13 shows the feature interactions for this oil pump body family and their predicate-based representations are given in Table 2.

7.1.3. Representing the variety configuration constraints

According to Section 4.4, variety configuration constraints indicate the selective relationships between the part variety components at the three decomposition levels. The propositional-logic-based scheme is applied to represent the variety configuration constraints for this oil pump body family. Table 3 lists these variety configuration constraints. A part of these constraints come from the configuration relations between the upper level components and their lower level components in Fig. 12, while the others are the constraints among the lower level components in Fig. 12.

7.2. Generating the RMPP

7.2.1. Generating the RMOPs for the feature clusters

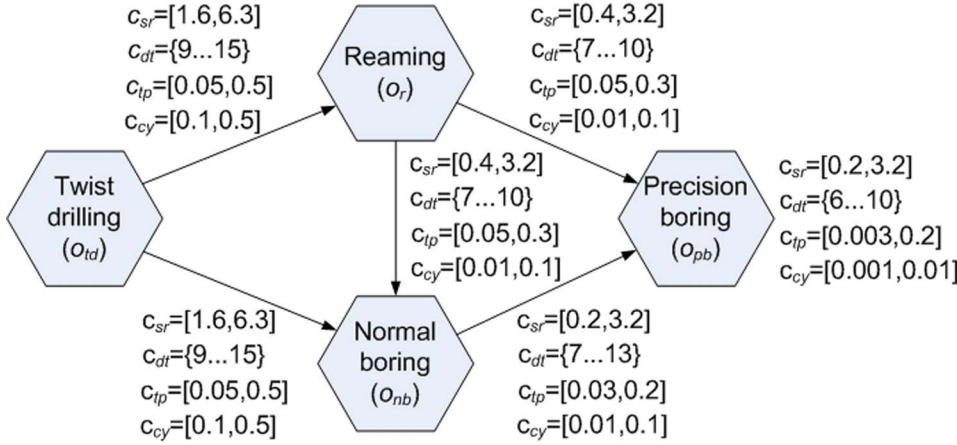
By following the two generation procedures proposed in Section 5.3, eight RMOPs are generated for the eight feature clusters in this part family as shown in Fig. 14, all of them are represented as directed graphs. Here we show the generation process of the RMOP for the hole cluster - HC2, for the other feature clusters, the processes are the same.

Table 4 lists the process-related design specifications of HC2. The attribute values of HC2 are defined as domains using either intervals or sets. Four types of hole machining operations have been selected for machining HC2. These hole-making operations are found from a machining operation library by applying a knowledge-based approach. Their machining capabilities are then extracted from the library, which are shown in Table 5. After applying a sequencing algorithm to find all the possible sequences of these feasible operations, the RMOP for HC2 is derived as shown in Fig. 15.

7.2.2. Generating the RMPP for the part family

The precedence relations between the interactive part variety components are derived by analysing the feature interactions listed in Table 2. Fig. 16 shows these precedence relations. There are two precedence relations at feature cluster level for this part family: *solidBefore*(poc1,cc1) and *solidImmeBefore*(hc2, poc1). A precedence relation between two feature clusters determines the precedence relations between the feature variants in these feature clusters. For example, because there is a precedence relation - *solidBefore* between PoC1 and CC1, there is the same precedence between PO100 and CH100: *solidBefore*(po100,

Fig. 15. The RMOP for HC2.



ch100).

Given the RMOPs and the precedence relations, the RMPP for this part family is represented as a 2-tuple $\langle \Psi, \Omega \rangle$, where:

- $\Psi = \{ \langle poc1, rm1 \rangle, \langle cc1, rm2 \rangle, \langle plc2, rm3 \rangle, \langle hc2, rm4 \rangle, \langle hc3, rm5 \rangle, \langle hc4, rm6 \rangle, \langle plc4, rm7 \rangle, \langle cc4, rm8 \rangle \}$
- $\Omega = \{ solidBefore(poc1, cc1), solidImmeBefore(hc2, poc1), solidBefore(po100, ch100), solidImmeBefore(cy110, po100), solidBefore(cy110, cy120), solidBefore(po200, ch200), softBefore(pl230, cy231), solidImmeBefore(cy210, po200), solidImmeBefore(pl210, cy210), solidImmeBefore(cy230, cy231), solidBefore(cy210, cy230), softBefore(pl231, cy230), solidBefore(cy230, ch230) \}$

7.3. Configuring a part variant and its process plan

The FPVM and RMPP for this oil pump body family are generated before a part variant is specified. According to the solution framework, once there is a requirement for a specific oil pump body variant, the configuration of this part variant and its process plan elements are generated through integrated part configuration and RMPP configuration. The integrated part configuration and RMPP configuration in this case is modeled as a DCSP - $\langle V, D, C \rangle$, where.

- $V = \{ V_{fm}, V_{fc}, V_{fv}, V_{rmop}, V_{pr} \}$ where
 - $V_{fm} = \{ f1, f2, f3, f4 \}$, each element in V_{fm} corresponds to a function module in Fig. 12;
 - $V_{fc} = \{ poc1, cc1, plc2, hc2, hc3, hc4, cc4, plc4 \}$, each element in V_{fc} corresponds to a feature cluster in Fig. 12;
 - $V_{fv} = \{ po100, po200, ch100, ch200, pl210, cy110, cy210, cy120, cy230, cy231, ch230, pl230, pl231 \}$, each element in V_{fv} corresponds to a feature variant in Fig. 12;
 - $V_{rmop} = \{ rm1, rm2, rm3, rm4, rm5, rm6, rm7, rm8 \}$, each element in V_{rmop} represents a RMOP in Fig. 14;
 - $V_{pr} = \{ pr3, pr5, pr7, pr8, pr9, pr10, pr11, pr12, pr13 \}$, each element in V_{pr} represents a precedence relation in Fig. 16; pr1, pr2, pr4 and pr6 are not included in V_{pr} because these precedence relations are determined by either pr12 or pr13.
- $D = \{ 0, 1 \}$ and each variable in V takes a value from D .
- $C = \{ C_c, C_a \}$ where
 - C_c consists of the variety configuration constraints listed in Table 3 as well as the mapping relations between the physical components and the components of the RMPP. The mapping relations are also represented as a set of proposition formulas;
 - C_a is a set of activity constraints shown in Table 6. These constraints guide the problem solving process to explore only the values of activated variables.

This DCSP problem is programmed in a constraint logic programming system, named ECLIPSe, and solved by a boolean CSP solver in this system. In the program, the initial configuration requirements for a part variant are set on the function modules as $\{ f1 = 1, f2 = 1, f3 = 0, f4 = 1 \}$, then the CSP solver found the following configurations satisfying all the constraints in C :

- Function Modules: F1, F2, F4;
- Feature Clusters: PoC1, CC1, HC2, PLC2, HC4, PLC4, CC4;
- Feature Variants: PO200, CH200, PL210, CY210, CY230, CY231, CH230, PL230, PL231;
- RMOPs: RM1, RM2, RM3, RM4, RM6, RM7, RM8;

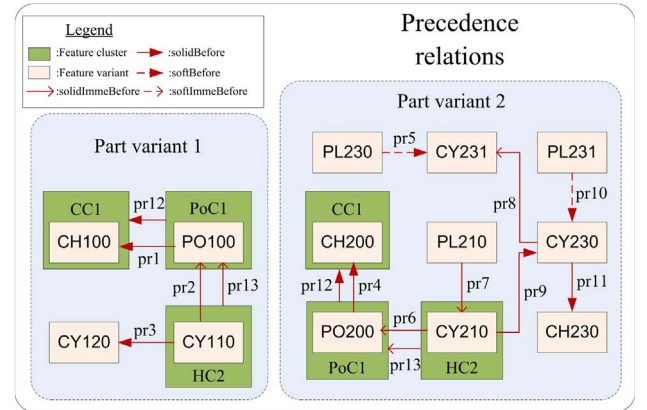


Fig. 16. Precedence relations between the interactive part variety components.

Table 6

Activity constraints in the DCSP for integrated part configuration and RMPP configuration.

Activity constraints	Description
$f1 = 1 \rightarrow Active: poc1 \wedge Active: cc1;$ $f2 = 1 \rightarrow Active: plc2 \wedge Active: hc2;$ $f3 = 1 \rightarrow Active: hc3;$ $f4 = 1 \rightarrow Active: hc4 \wedge Active: cc4 \wedge Active: plc4;$	The hierarchical relations between function modules and feature clusters
$poc1 = 1 \rightarrow Active: po100 \wedge Active: po200;$ $cc1 = 1 \rightarrow Active: ch100 \wedge Active: ch200;$ $plc2 = 1 \rightarrow Active: pl210; hc3 = 1 \rightarrow Active: cy120;$ $hc2 = 1 \rightarrow Active: cy110 \wedge Active: cy210;$ $hc4 = 1 \rightarrow Active: cy230 \wedge Active: cy231;$ $cc4 = 1 \rightarrow Active: ch230;$ $plc4 = 1 \rightarrow Active: pl230 \wedge Active: pl231$	The hierarchical relations between feature clusters and feature variants

Term display for solve_desp@eclipse					Term display for solve_desp@eclipse				
	1	2	3	4		1	2	3	4
1	F1{[0, 1]}	F2{[0, 1]}	F3{[0, 1]}	F4{[0, 1]}	1	1	1	0	1
2	POC1{[0, 1]}	CC1{[0, 1]}	HC2{[0, 1]}	PLC2{[0, 1]}	2	1	1	1	1
3	HC3{[0, 1]}	HC4{[0, 1]}	PLC4{[0, 1]}	CC4{[0, 1]}	3	HC3{[0, 1]}	1	1	1
4	PO100{[0, 1]}	PO200{[0, 1]}	CH100{[0, 1]}	CH200{[0, 1]}	4	0	1	0	1
5	PL210{[0, 1]}	CY110{[0, 1]}	CY210{[0, 1]}	CY120{[0, 1]}	5	1	0	1	CY120{[0, 1]}
6	CY230{[0, 1]}	CY231{[0, 1]}	CH230{[0, 1]}	PL230{[0, 1]}	6	1	1	1	1
7	PL231{[0, 1]}	RM1{[0, 1]}	RM2{[0, 1]}	RM3{[0, 1]}	7	1	1	1	1
8	RM4{[0, 1]}	RM5{[0, 1]}	RM6{[0, 1]}	RM7{[0, 1]}	8	1	0	1	1
9	RM8{[0, 1]}	PR3{[0, 1]}	PR5{[0, 1]}	PR7{[0, 1]}	9	1	0	1	1
10	PR8{[0, 1]}	PR9{[0, 1]}	PR10{[0, 1]}	PR11{[0, 1]}	10	1	1	1	1
11	PR12{[0, 1]}	PR13{[0, 1]}	unused	unused	11	1	1	unused	unused

(1) Before the problem solving process

(2) After the problem solving process

Fig. 17. Configuration results for this case study.

- Precedence Relations: PR5, PR7, PR8, PR9, PR10, PR11, PR12, PR13.

The values of the variables before and after the configuration process are shown in Fig. 17. These configuration results are in accordance with the configuration of the oil pump body 2 shown in Fig. 12, and the derived process plan elements are used to build the feasible and optimal process plans for this part variant.

7.4. Discussion

This case study uses an oil pump body family to illustrate the proposed representation models and the generation methods for RMPP. The feature-based part variety model represents both the process-related information and the configuration-related information of two oil pump body variants in a well-structure and less-redundant way. The RMPP for this oil pump body family contains all the process plan elements for building all the feasible process plans for each oil pump body variant. Once there is a configuration requirement for a specific oil pump body variant, the configuration of this part variant and its process plan elements can be generated by integrated part configuration and RMPP configuration. The final process plans for a specific manufacturing scenario are then generated based on these process plan elements instead of generating it from scratch. In this way, reconfigurable machining process planning is able to flexibly response to any change requirement on the current machining process plan.

8. Conclusions

Conventional machining process planning approaches fail to embrace the modularity, commonality and reconfigurability among different part variants, which make them inefficient to handle current manufacturing scenario dominated by product families and dynamics. RPP is a new process planning approach conceived with modularity, platform-based and configuration-based techniques in mind. With a capability to adapt to new changes on product and manufacturing system, RPP has been well recognized as a key enabler for today's manufacturing paradigms.

This paper aims to extend the conventional approaches from targeting one single part to designing processes for a domain of part variants. This is fulfilled by further developing the concept of RPP to a concept of RMPP with the help of modularity, platform-based and configuration-based techniques. As a result, a solution framework for RMPP is proposed in this paper. In this framework, a feature-based part variety model is developed to represent the design specifications of a

part family from the perspective of part configuration as well as the perspective of process planning. Based on this part variety model, a concept of RMOP is then defined and a model for RMPP is defined as a combination of the RMOPs for all the feature clusters and the precedence relations between the interactive part variety components.

Two principles are used when developing these representation models: (1) modularly decompose the whole part family into correlative and configurable modules and use features as the atomic representation; (2) maximally represent the commonality among all the part variants in the part family so as to reduce datum redundancy. By following the principle (1), a variety decomposition network is developed to build the architecture of a part family and also propose the representation schemes for feature interactions and variety configuration constraints. Following the principle (2), the collective feature interactions and the collective precedence relations are represented at feature cluster level since they maintain the same in every part variant.

In the proposed solution framework, a RMOP generation method is developed to generate all the RMOPs in the RMPP for a part family; in addition, an approach called integrated part configuration and RMPP configuration is proposed to flexibly generate a specific part variant and its process plan elements. These process plan elements are then used to build the final process plan for a specific part variant in a specific manufacturing scenario.

In comparison with the conventional machining process planning approaches, the proposed RMPP approach reduces the process planning complexity for a part family. In order to simply show this comparison, we reasonably assume that there is a part family which has n part variants and we divide the process planning efforts for one part variant into two portions: (1) the efforts to analyse the information which are common in every part variant, such as generating the machining operation for a common feature; (2) the efforts to analyse the information which are not common in every part variant. The first portion is denoted as e^1 and the second one is denoted as e_k^2 , then we use E_1 and E_2 to denote the total efforts for the conventional approach and the proposed RMPP approach and we have:

- For the conventional approach, because they generate the process plans for one single part, $E_1 = ne^1 + \sum_{k=1}^n e_k^2$;
- For the RMPP approach, because of the commonalities in its models, $E_2 = e^1 + \sum_{k=1}^n e_k^2$;

It is obvious that $E_2 < E_1$. More importantly, it supports the process planning for a part family with infinite number of variants, because the feature-based part variety model uses domains to represent the attribute value of a feature cluster.

To verify the proposed models and approaches, they are applied on an oil pump body family. This paper focuses on the introduction and explanation of the concepts, the models and the solution framework relevant to RMPP. In the future work, the proposed models will be applied on a more complex industrial case and more manufacturing scenarios; moreover, an effective algorithm for generating a RMPP for a part family will be further investigated.

Acknowledgement

The authors gratefully acknowledge the financial support of China Scholarship Council (CSC) (Contract No. 201306050034). We also appreciate the anonymous referees for their valuable suggestions that have helped improve this paper substantially.

References

- Amaitik, S., & Kilic, S. (2005). Step-based feature modeller for computer-aided process planning. *International Journal of Production Research*, 43, 3087–3101.
- Azab, A., Perusi, G., ElMaraghy, H., & Urbanic, J. (2007). Semi-generative macro-process planning for reconfigurable manufacturing. *Digital enterprise technology* (pp. 251–258). Springer.
- Bonev, M., Hvam, L., Clarkson, J., & Maier, A. (2015). Formal computer-aided product family architecture design for mass customization. *Computers in Industry*, 74, 58–70.
- Burbidge, J. (1993). Comment on clustering methods for finding gt groups and families. *Journal of Manufacturing Systems*, 12, 428–429.
- Chaube, A., Benyoucef, L., & Tiwari, M. (2012). An adapted nsga-2 algorithm based dynamic process plan generation for a reconfigurable manufacturing system. *Journal of Intelligent Manufacturing*, 23, 1141–1155.
- Deb, S., Ghosh, K., & Paul, S. (2006). A neural network based methodology for machining operations selection in computer-aided process planning for rotationally symmetrical parts. *Journal of Intelligent Manufacturing*, 17, 557–569.
- Denkena, B., Shpitalni, M., Kowalski, P., Molcho, G., & Zipori, Y. (2007). Knowledge management in process planning. *CIRP Annals-Manufacturing Technology*, 56, 175–180.
- ElMaraghy, H. (2006). Reconfigurable process plans for reconfigurable manufacturing. In *Proceedings of the 3rd international CIRP sponsored conference on digital enterprise technology*.
- ElMaraghy, H. (2009). Changing and evolving products and systems—models and enablers. *Changeable and reconfigurable manufacturing systems* (pp. 25–45). Springer.
- ElMaraghy, W., ElMaraghy, H., Tomiyama, T., & Monostori, L. (2012). Complexity in engineering design and manufacturing. *CIRP Annals – Manufacturing Technology*, 61, 793–814.
- ElMaraghy, H., Schuh, G., ElMaraghy, W., Piller, F., Schönleben, P., Tseng, M., & Bernard, A. (2013). Product variety management. *CIRP Annals – Manufacturing Technology*, 62, 629–652. <http://dx.doi.org/10.1016/j.cirp.2013.05.007>.
- Etienne, A., Dantan, J. Y., Siadat, A., & Martin, P. (2006). An improved approach for automatic process plan generation of complex borings. *Computers in Industry*, 57, 663–675.
- Givehchi, M., Haghghi, A., & Wang, L. (2015). Generic machining process sequencing through a revised enriched machining feature concept. *Journal of Manufacturing Systems*, 37, 564–575.
- Haug, A., Hvam, L., & Mortensen, N. (2011). The impact of product configurators on lead times in engineering-oriented companies. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 25, 197–206.
- Hirshburg, K., & Siddique, Z. (2014). A proactive scaling platform design method using modularity for product variations. *Advances in Product Family and Product Platform Design* (pp. 201–219). Springer.
- Hu, S. J. (2013). Evolving paradigms of manufacturing: From mass production to mass customization and personalization. *Procedia CIRP*, 7, 3–8.
- Hvam, L., Haug, A., & Mortensen, N. (2010). Assessment of benefits from product configuration systems. In *13th Workshop on configuration, ECAI 2010, the 19th European conference on artificial intelligence*.
- Jiao, J., Zhang, L., & Pokharel, S. (2007). Process platform planning for variety coordination from design to production in mass customization manufacturing. *IEEE Transactions on Engineering Management*, 54, 112–129.
- Koren, Y., Hu, S., Gu, P., & Shpitalni, M. (2013). Open-architecture products. *CIRP Annals-Manufacturing Technology*, 62, 719–729.
- Markus, A., Vánca, J., & Horvath, M. (1997). Process planning by retrieval and adaptation. *Computers in Industry*, 33, 47–60.
- Pirmoradi, Z., Wang, G., & Simpson, T. (2014). A review of recent literature in product family design and platform-based product development. In T. Simpson, J. Jiao, Z. Siddique, & K. Hölttä-Otto (Eds.). *Advances in product family and product platform design* (pp. 1–46). (1st ed.). London: Springer chapter 1.
- Pitiot, P., Aldanondo, M., & Vareilles, E. (2014). Concurrent product configuration and process planning: Some optimization experimental results. *Computers in Industry*, 65, 610–621.
- Quintanilla, F. G., Cardin, O., L'Anton, A., & Castagna, P. (2016). A petri net-based methodology to increase flexibility in service-oriented Holonic manufacturing systems. *Computers in Industry*, 76, 53–68.
- Salvador, F., & Forza, C. (2004). Configuring products to address the customization-responsiveness squeeze: A survey of management issues and opportunities. *International Journal of Production Economics*, 91, 273–291.
- Salvador, F., Forza, C., & Rungtusanatham, M. (2002). Modularity, product variety, production volume, and component sourcing: theorizing beyond generic prescriptions. *Journal of Operations Management*, 20, 549–575.
- Tatikonda, M., & Wemmerlöv, U. (1992). Adoption and implementation of group technology classification and coding systems: insights from seven case studies. *The International Journal of Production Research*, 30, 2087–2110.
- Tolio, T., Ceglarek, D., ElMaraghy, H., Fischer, A., Hu, S., Laperrière, L., ... Vncza, J. (2010). Species-co-evolution of products, processes and production systems. *CIRP Annals – Manufacturing Technology*, 59, 672–693. [http://dx.doi.org/10.1016/0003-4916\(63\)90068-X](http://dx.doi.org/10.1016/0003-4916(63)90068-X).
- Vickery, S., Bolumole, Y., Castel, M., & Calantone, R. (2015). The effects of product modularity on launch speed. *International Journal of Production Research*, 53, 5369–5381.
- Wang, L. (2013). Machine availability monitoring and machining process planning towards cloud manufacturing. *CIRP Journal of Manufacturing Science and Technology*, 6, 263–273.
- Wang, L., Zhong, S., & Zhang, Y. (2015). Process configuration based on generative constraint satisfaction problem. *Journal of Intelligent Manufacturing*, 1–13.
- Xu, X., Klemm, P., Proctor, F., & Suh, S. (2006). Step-compliant process planning and manufacturing. *International Journal of Computer Integrated Manufacturing*, 19, 491–494.
- Xu, X., Wang, L., & Newman, S. (2011). Computer-aided process planning—a critical review of recent developments and future trends. *International Journal of Computer Integrated Manufacturing*, 24, 1–31.
- Yang, D., Dong, M., & Chang, X. (2012). A dynamic constraint satisfaction approach for configuring structural products under mass customization. *Engineering Applications of Artificial Intelligence*, 25, 1723–1737.
- Zhang, L. (2014). Product configuration: A review of the state-of-the-art and future research. *International Journal of Production Research*, 52, 6381–6398. <http://dx.doi.org/10.1080/00207543.2014.942012>.
- Zhang, L., Xu, Q., Yu, Y., & Jiao, R. (2012). Domain-based production configuration with constraint satisfaction. *International Journal of Production Research*, 50, 7149–7166.