J Intell Manuf (2008) 19:625–641
DOI 10.1007/s10845-008-0128-y

Paradigm shift: unified and associative feature-based concurrent and collaborative engineering

Y.-S. Ma · G. Chen · G. Thimm

Abstract With widely used concurrent and collaborative engineering technologies, the validity and consistency of product information become important. In order to establish the state of the art, this paper reviews emerging concurrent and collaborative engineering approaches and emphasizes on the integration of different application systems across product life cycle management (PLM) stages. It is revealed that checking product information validity is difficult for the current computer-aided systems because engineering intent is at best partially represented in product models. It is also not easy to maintain the consistency among related product models because information associations are not established. The purpose of this review is to identify and analyze research issues with respect to information integration and sharing for future concurrent and collaborative engineering. A new paradigm of research from the angle of feature unification and association for product modeling and manufacturing is subsequently proposed.

Keywords Concurrent and collaborative engineering · Feature-based design and manufacturing · Product life cycle modeling · Information validity and consistency

Introduction

As an advanced methodology for product development and manufacturing, since 1980s, concurrent engineering has been well studied and widely implemented in industry to compress time to market and cost (Prasad 1996). The basic concept is to achieve concurrency and close-loop evolution in product development and manufacturing processes via a tight integration of applications and parallel engineering. Concurrent engineering considers all life cycle issues simultaneously. The traditional sequential mode of product development has been changed into an iterative and evolutionary mode. On the one hand, to realize the overall required product functions, many aspects need to be considered, such as the spatial, stability, and esthetic concerns in building design (Rosenman and Gero 1996). On the other hand, to fulfill product life cycle requirements, downstream stages, such as machining and assembly for mechanical products, need to be considered in the early stages, such as conceptual design, to optimize the product and its related processes (Xue et al. 1999; Xue and Yang 2004; Roucoules et al. 2003; Feng and Song 2000). These publications reveal the complexity of the inherent relations across product life cycle stages. Traditionally, concurrent engineering is adopted in single or closely partnered companies, such as Boeing, General Motors, Toyota and their major suppliers. In the development of this technology, feature-based engineering has played a major role as one of the corner stones in effectively implementing knowledge embedment and modular information support within many computer aided systems, such CAD, CAM, CAPP applications (Shah and Mantyla 1995; Otto 2001). Since the introduction of feature technology two decades ago, it was proven to be a useful tool to model engineering semantics as well as to maintain associations among geometric entities.

Collaborative engineering is a technological approach that supports distributed, multi-disciplinary, and multi-organizational teams during the product development and manufacturing processes. This approach is motivated by the globalization of economy and boosted by the developments

---

Y.-S. Ma (✉)
Department of Mechanical Engineering, Faculty of Engineering, University of Alberta, Edmonton, Canada
e-mail: yongsheng.ma@ualberta.ca

G. Chen · G. Thimm
School of MAE, Nanyang Technological University, Singapore, Singapore

of supply chain (Chopra and Meindl 2007) and the Internet (Fuh and Li 2005; Yang and Xue 2003; Wang et al. 2002).

Today, customer requirements on product quality, time-to-market and manufacturing cost have become more and more stringent. Global product design and manufacturing has been pushing the adoption of a combined concurrent and collaborative engineering approach. This approach requires the integration of business and engineering application systems in an effective and efficient manner and involves the management of product data, process data, and engineering knowledge. Consequently, the importance of the validity and consistency of product and process information is emphasized; however, existing computer-aided systems have difficulties to fulfill this requirement. Therefore, a review of the state of art in this aspect of the concurrent and collaborative engineering is a timely effort in order to consolidate the potential new directions of research in the field. So far, there is not an updated and thorough review available from this angle.

This review starts with a review of feature-based technologies for product modeling and manufacturing (see section "Feature-based product modeling"). Several fundamental aspects of feature technologies, which include feature definitions, geometric representation schemes, feature relations, validity checking, and feature-based application integration, are discussed. Then, in Section "Developments and requirements of feature-based concurrent and collaborative engineering", the current development and requirements of concurrent and collaborative engineering are studied. Key technological aspects are covered, such as application integration, enterprise resources planning, knowledge-based engineering, etc. Existing shortcomings of concurrent and collaborative engineering are identified. Section "Research issues" analyzes research issues to highlight the requirements for a new approach. It is believed by the authors that the existing feature technologies has to be significantly further-developed by a unification approach to meet the modeling requirements of engineering informatics. Consequently, based on the review findings, Section "The authors' view of a new paradigm" proposes a new research approach, i.e. a unified and associative feature technology from the angle of supporting concurrent and collaborative engineering. The "Conclusions" section concludes this paper.

Feature-based product modeling

In this section, past research works on feature definitions, feature-based application, and feature-based integration methods are briefly reviewed.

Historical definitions of features

The concept of features is flexible and can be used in many aspects of mechanical engineering (Shah and Mantyla 1995). Early representative definitions are more specifically associating geometries of a part with certain engineering meanings; they include *Regions* of interest in a part model (Wilson and Pratt 1988); *Geometric forms* or *entities* that are used in reasoning in one or more design or manufacturing activities (Cunningham and Dixon 1988); *Generic shapes* associated to certain properties or attributes as well as knowledge useful in reasoning about the product (Shah 1991); or *Regions of an object* that are meaningful for a specific activity or application (Vandenbrande and Requicha 1993). More recently, feature definitions are more expanded to cover the applications in different stages of product life cycles. For example, Martino et al. define a feature as a set of form elements with functional meaning in a given application context that allows an association between shape and functionality (Martino et al. 1998) while Bidarra and Bronsvoort generalize a feature as a representation of shape aspects of a product that are partially mapped onto a generic shape and are functionally significant for some product life cycle phases (Bidarra and Bronsvoort 2000). These definitions reveal that features have two fundamental characteristics: (1) Features relate to product geometry on a level higher than geometric and topological entities; (2) they represent engineering intent.

The fundamentals of feature technology

As flexible and useful building blocks of product models, three fundamental aspects of features have to be addressed: geometric representation, data relation management, and model validation. The relevant research is reviewed in this sub-section.

Geometric representation schemes

The mainstream geometric representation schemes used in feature-based modeling systems are boundary representation (B-rep) and constructive solid geometry (CSG) (Shah and Mantyla 1995). B-rep explicitly stores part boundary, such as faces, edges, and vertexes. Low level information is hence available but the data structure is complex for specification and modification. In contrast, CSG uses primitive solids and set operations to construct part geometry. Low level information is not available but the data structure is compact and easily used during specification (Requicha 1980; Hoffman 1989). Usually, these two schemes are combined (Gossard et al. 1988; Roy and Liu 1998; Venkataraman et al. 2001).

Relations in a feature-based application model

A feature combines geometric and non-geometric entities. Therefore, compared with geometric models, relations are more complex in feature models. Managing these relations, especially the non-geometric ones, is essential for the validity of a product model (Otto 2001). Relations in a feature-based product model can be classified into two categories, *geometric relations* and *non-geometric relations* (see also Table 1).

Geometric relations refer to those constrained associations among topological or geometrical entities, i.e. solids, faces, edges, vertices, surfaces, curves, points, etc. Many publications focus on geometric relations in a feature model (Shah and Rogers 1993; Brunetti et al. 1995). These relations are explicitly declared and represented as geometric constraints, which allows maintaining the geometric integrity of features. Sometimes, unintentional feature interactions may affect validity of features (Karinthi and Nau 1992; Bidarra et al. 1997; Hounsell and Case 1999). These interactions usually cannot be prevented by geometric or algebraic constraints. Such geometric feature interactions can only be managed through associations between a feature model and a geometric model.

Non-geometric relations refer to any dependency associations involving entities representing non-geometric properties. For example, in process planning, the clamping faces or accessing faces are required to machine a feature. Interferences may occur if a wrong machining sequence is used (Faheem et al. 1998). Then the constraints applied to define interference checking entities are non-geometric relations. Taking another example, two features, which may not overlap spatially and even belong to different product life cycle stages, may interact with each other (Regli and Pratt 1996). Such interactions defined are also non-geometric relations. In addition, non-geometric relations also exist between features and non-geometric entities. For example, at the design stage, functional-form matrixes, bipartite function-feature graphs, design flow chains, design key characteristics, and mapping hierarchies can be used to link abstracted concept features to product functions (Feng et al. 1996; Mukherjee and Liu 1997; Whitney et al. 1999; Brunetti and Golob 2000; Brunetti and Grimm 2005). At the process planning stage, machining features are also related to non-geometric entities, such as machines, cutting tools, and machining processes (Khoshnevis et al. 1999; Sormaz and Khoshnevis 2003). Currently, it is still a challenge of research to completely represent non-geometric relations and to validate product models using non-geometric relations.

Using features to maintain a product model's validity

A product model must have a sound mechanism to check its validity. Compared to the strict validity maintenance mechanisms of B-rep or CSG, current feature-based modeling schemes are weak in this aspect. Rossignac (1990) suggested that validity of features should be defined in terms of the referenced geometric entities as well as their existence, shape, and relations to other geometric elements of the model. However, the validity of features must be checked more widely. The authors believe that a feature model is valid only when five conditions are satisfied. Firstly, the geometric and algebraic constraints specified on features are satisfied. Secondly, the feature model is consistent with the geometric model (Laakko and Mantyla 1993; Geelink et al. 1995). This consistency should also be used to validate engineering intent (Martino et al. 1994; Vieira 1995; Mandorli et al. 1997). Note that in the aforementioned publications, engineering intent has to be embedded into geometric, algebraic, or preliminary semantic constraints such as the boundary or interaction constraints (Bidarra and Bronsvoort 2000) without explicit definitions. Therefore, during transformation along engineering processes, engineering intent is not maintained and very often lost in downstream processes. Thirdly, non-geometric

Table 1 Summary of research on relations in feature-based applications

Relation nature	Related entities	Representation	Source
Geometric relations	Between geometric entities	Geometric constraints	Shah and Rogers 1993; Brunetti et al. 1995; van Holland and Bronsvoort 2000
	Between features	Interaction constraints	Karinthi and Nau 1992; Vieira 1995; Bidarra et al. 1997; Hounsell and Case 1999
Non-geometric relations	Between features and the corresponding geometric entities	Features refer to the corresponding geometric entities	Vandenbrande and Requicha 1993; Bidarra and Bronsvoort 2000; Laakko and Mantyla 1993; Mandorli et al. 1997
	Between features	Not mentioned	Regli and Pratt 1996; Faheem et al. 1998; Sharma and Hayes 2001
	Between features and other non-geometric entities, such as functions, behaviors, assembly methods, machines, cutting tools	Tables, graph, rules, etc.	Schulte et al. 1993; Feng et al. 1996; Anderl and Mendgen 1996; Mukherjee and Liu 1997; Whitney et al. 1999; Khoshnevis et al. 1999; Stage et al. 1999; Brunetti and Golob 2000; Park 2003

constraints specified among features are satisfied. For example, different machining sequences may influence the presence, shape, volume, and validity of machining features (Regli and Pratt 1996; Faheem et al. 1998; Sharma and Hayes 2001). So far, in most of the commercial CAx systems, methods for the representation, validation, and maintenance of inter-feature non-geometric constraints are insufficient. Fourthly, features are consistent with the related non-geometric entities. For example, the presence of features or the values of feature parameters may be determined by functional requirements or machining conditions (Anderl and Mendgen 1996; Oral and Cakir 2004). Lastly, the feature model is consistent with the engineering intent.

The first aspect is the traditional constraint satisfaction problem, which is mostly solved. Some researchers considered the second aspect but hardly touched on the other three aspects although they are equally important for product model validity.

Traditional feature-based applications

This subsection reviews definitions, semantics, and usage of features in specific applications. Table 2 gives a summary of the semantics and geometric representations of application features.

Conceptual design features (also called functional features)

Activities during the conceptual design stage concentrate on determining basic principles and critical specifications of a product. The engineering intent in this stage is on the product's functions, structure, material, and other critical properties. Due to the lack of product details, only a few researchers consider features in this stage. For example, functional features (Mcginnis and Ullman 1992; Brown 2002), which are non-geometric in nature, are only used to explain the purpose of individual design objects. In the past, functional features were used to associate product geometry with functions because they are used in the reasoning form features' usage (Wood and Ullman 1996); form features were also connected to product functions (Mukherjee and Liu 1997). The interesting use of functional features was to represent interactions between components based on physical effects, which are crucial for function realization (Schulte et al. 1993). Functional features associate surfaces, which interact with each other, with the specific spatial arrangement and relative motions of faces. This definition preliminarily links the realization of product functions to the corresponding geometry and topology. However, it does not fulfill two basic requirements, i.e. (1) representing functional requirements and realization rules in features; and (2) supporting incomplete, inexact, yet critical product geometries.

Detailed design features

In the detailed design stage, the main task is specifying the detailed product geometry, topology, and other properties for production. The design intent at this stage has to be consistent with the guidelines specified in the conceptual design, such as required functions or design patterns (Lee 2005; Ma and Tong 2003). For example, Stefano et al. (2004) used particular geometric characteristics in detailed designs to represent product functionality, but how to connect these geometric characteristics to the conceptual design for design validation was not mentioned.

Feature-based approach is also widely used for DFX considerations (Fazio et al. 1999; Boothroyd et al. 2002). Most research on manufacturability or assemblibility analysis is in this category, such as machining feasibility, complexity analysis, interference checking, or moldability evaluation (Ong and Chew 2000; Chen et al. 2002; Li 2005; Lockett and Guenov 2005). However, usually in detailed design stage, only form features are used for product geometry construction. The purpose of these features is not well specified and retained. A typical definition of a form feature is "a shape macro that is constructed for convenience, with little connection to function or manufacturing" (Han and Requicha 1998). Form features hide the tedious geometric modeling and editing processes from designers. Most current feature technologies, such as feature relations and validity control, deal with form features only. Since the purposes of traditional form features are not well specified, using them directly in application reasoning is difficult.

CAE features

In CAE analysis stage, the engineering intent is to select parts of a detailed design that are significant in a specific type of analysis. For example, the suppressibility feature attributes can be specified for idealization purpose (Deng et al. 2002). Lee used idealization features to integrate design and CAE analysis (Lee 2005). These idealization features contain the necessary information for the construction of models in different detail or abstraction levels. The idealization features keep the consistency among these representations. Only the geometric and topological specifications of the idealization features are mentioned.

Assembly design features

The main design intent in this stage is achieving the required product functions through specifying relations between mating parts. Assembly features were traditionally used to represent geometric relations between mating parts (Shah and Rogers 1993; Anantha et al. 1996; Case and Harun 2000; Chan and Tan 2003). The design intent of these features and

Table 2 Summary of application features

Application	Semantics	Geometry	Current limitations	Source
Conceptual design	Representing the interacting parts. The interactions result in the state transitions of the parts, which are part behaviors that are used to realize product functions	Solid, surface, wireframe	Geometric representations need to be further studied; Representation of functions and behaviors in features are incomplete	Mcginnis and Ullman 1992; Schulte et al. 1993; Wood and Ullman 1996; Mukherjee and Liu 1997; Brown 2002
Detailed design	Portions of a single part that materialize the conceptual design or are added for DFX purposes	Solid	Engineering intent of the used form features are not well specified	Han and Requicha 1998; Ong and Chew 2000; Chen et al. 2002; Stefano et al. 2004; Li 2005; Lockett and Guenov 2005
CAE analysis	Portions of a component, which are significant in a particular type of analysis	Solid, surface, wireframe	Not included in this review	Deng et al. 2002; Lee 2005
Assembly design	Portions of mating parts that achieve the product functions or are created for modularization or enhancing the assemblability	Solid, surface	Functional relations across features are not well specified	Shah and Rogers 1993; Anantha et al. 1996; Deneux 1999; Whitney et al. 1999; Case and Harun 2000; Shyamsundar and Gadh 2001; Csabai et al. 2002; Chan and Tan 2003; Ma et al. 2004a, 2007
Process planning	Geometry created or used in manufacturing processes, which depends on manufacturing constraints or process planning preferences	Solid, surface	Machining features are not well related to their engineering intent	Kumar et al. 1992; Vandenbrande and Requicha 1993; Joshi and Chang 1988; Tseng and Joshi 1994; Gupta and Nau 1995; Chu and Gadh 1996; Stefano 1997; ISO 1999; Gaines and Hayes 1999; Stage et al. 1999; Raman and Marefat 2004
Assembly planning	Portions of mating parts that affect the assembly processes, sequence, stability, etc.	Solid, surface	Not included in this review	van Holland and Bronsvoort 2000; Kim et al. 2004, 2006

the corresponding geometric relations is not explicitly specified. In addition, these definitions are only suitable when form features in mating parts have already been created, i.e. a bottom-up design approach is followed.

However, for top-down design processes, relations between parts may need to be specified while the detailed part geometry is still incomplete (Ma et al. 2007). These relations can be defined as assembly features and used to assign responsibilities in collaborative design (Shyamsundar and Gadh 2001; Csabai et al. 2002). The possibility of using assembly features to represent design intent is mentioned in Deneux (1999). However, no details are given. Whitney suggested transforming design intent into associations among key characteristics, corresponding datum flow chains, and assembly features (Whitney et al. 1999). This approach is more appropriate for representing design intent than purely using geometric or algebraic constraints.

Machining features

The origin of the feature technology probably lies in the search for a high level geometric representation to support process planning and CNC machining. Initially, only geometric aspects of machining features were considered. It was

Table 3 Summary of research on feature-based application connection or integration

Approach		Connection mechanism	Connected or shared entities	Current limitations	Source
Feature conversion		Direct conversion between feature models	Feature volumes	Feasibility needs to be proved	Cunningham and Dixon 1988; Anderson and Chang 1990; Bronsvoort and Jansen 1993; Subramani and Gurumoorthy 2005
Feature recognition		Usually none, except for the JTMS used in (Han 1996)	Geometric entities	No connection after recognition	Henderson 1984; Joshi and Chang 1988; Tseng and Joshi 1994; Kim 1994; Regli 1994; Han 1996
Common feature definitions	Design by feature	Application features are used for product design	Application features	Limitations on designers	Shah and Mantyla 1995
	Neutral features	All applications support neutral features	Neutral features	Lack of non-geometric associations	ISO 1999; Pratt and Srinivasan 2005
Multiple-view feature modeling	Pre-defined central model	NMT model or feature model tree	Geometric entities	Lack of non-geometric associations	Jha and Gurumoorthy 2000; Lee 2005
	Incremental updating the central model	Cellular model, intermediate model or master model	Geometric entities; or non-geometric entities that an application shares	Lack of non-geometric associations	Rosenman and Gero 1996; Dohmen et al. 1996; de Kraker et al. 1997; Suh and Wozny 1997; Martino et al. 1998; Bidarra et al. 1998; Hoffman and Joan-Arinyo 1998; Bronsvoort and Noort 2004

assumed that the purpose of machining features is implicitly embedded in the product model. For example, machining features are traditionally defined as volumes of material removed in machining operations (ISO 1999). Formalisms, such as attributed adjacency graph (Joshi and Chang 1988) or cell decomposition (Tseng and Joshi 1994), were proposed to represent machining features. Gradually, it was found that pure geometric methods may result in features, which are geometrically valid but invalid in the view of machining. Some researchers proposed using additional methods, such as the validity test in Vandenbrande and Requicha (1993) to prune infeasible machining features. Features mentioned in this paragraph are only machining feature candidates given to the process planning application for further selection.

To generate feasible and optimal machining feature interpretations of a design, accessibility test and manufacturability analysis are used (Gupta and Nau 1995; Chu and Gadh 1996). In this way, although features are still geometrically defined, engineering intent is used to control the generation of feature instances (Marefat and Britanik 1997).

Features are also used in other process planning activities, such as fixturability analysis (Kumar et al. 1992; Chu and Gadh 1996), as well as in other manufacturing processes, such as casting (Stefano 1997).

Assembly planning features

Holland and Bronsvoort (van Holland and Bronsvoort 2000) used handling features to represent feeding methods, feeding direction, fixturing methods, etc. Connection features are used to represent the insertion position, insertion path, tolerances, contact area, etc. Features are generated by assembly planning activities, such as fixture planning, feeding planning, stability analysis, etc. Kim et al. studied the assemblies created by welding and riveting (Kim et al. 2004, 2006). They used the associations among form features, geometric constraints, and joining methods to represent engineering intent. Non-geometric information is included in features and used to check whether the design specifications, such as degree of freedom, are satisfied by the selected joining methods. These two feature definitions link engineering knowledge, non-geometric entities, features, and product geometry explicitly.

Commonalities of application features

The reviews in this sub-section illustrate that application features are commonly regarded as constrained associations among a group of geometric and non-geometric entities. Application features must be tightly related to their purposes.

Pure geometric definitions are insufficient. In detail, the commonalities of application features include:

- Features use parameters and attributes to describe geometric and non-geometric feature properties.
- Features refer to a specific set of geometric entities.
- Algebraic and geometric constraints as well as the relations between features and geometric entities have to be maintained in order for the geometric integrity and validity of a feature model to persist.
- Features correspond to particular engineering processes and intent.

These commonalities provide a basis for the generic definition of distinct application features.

Developments and requirements of feature-based concurrent and collaborative engineering

Application integration

Traditional application integration approaches focus on the geometric data sharing. For example, integrations between design systems and reverse engineering (Varady et al. 1997; Benko et al. 2001), rapid prototyping (Starly et al. 2005), coordinate measurement (Kramer et al. 2001), mesh generation (Rezayat 1996), virtual reality (Ma et al. 2004b), process planning (Fuh and Chang 1996), and assembly systems (Noort et al. 2002) have been widely studied. In these publications, the application integration is based on geometric neutral data formats, such as the Initial Graphics Exchange Specification (IGES) or the STandard for the Exchange of Product model data (STEP) (ISO 2000). To support a comprehensive integration of applications, a more advanced data sharing mechanism is needed than those provided by the existing IGES or STEP standards.

Knowledge-based engineering

The product development process can be regarded as a sequence of decision making processes. Knowledge-based engineering (KBE) approaches are used in many CAx systems to support decision making, such as functional design, geometric constraint solving, parametric design, assembly oriented design, feature recognition, determining tooling components or layout, and process planning (Tor et al. 2002; Lee and Kim 1996; Myung and Han 2001; Zhang and Xue 2002; Zha et al. 2001a,b; Henderson 1984; Lee et al. 1997; Mok et al. 2001; Sormaz and Khoshnevis 1997; Park 2003; Chen et al. 2006). KBE approaches provide a better way than pure geometric modeling to embed engineering intent during product development processes.

Product model consistency

Collaborative engineering imposes conditions on the development of product modeling systems, i.e. model consistency among distributed and related applications must be controlled. In general, the consistency among distinct applications can be controlled via agents (Rosenman and Wang 1999), CORBA-based distributed programming (Pahng et al. 1998; Chan et al. 2000), or common databases (Kung et al. 1999). In particular, among designers working on the same product, such as a component, the geometric consistency can be controlled by specifically developed naming mechanisms (Lee et al. 1999; Bidarra et al. 2001; Li et al. 2004). For designers working on different components of an assembly, assembly features can be used to control the geometric compatibility between mating components (Shyamsundar and Gadh 2002; Ma et al. 2004a). These requirements need to be considered when developing a product modeling scheme.

Enterprise resource planning

Developed from the initial inventory control, subsequently material requirements planning (MRP) and manufacturing resources planning (MRP II), enterprise resource planning (ERP) are used to integrate separate enterprise information systems, such as design, manufacturing, material planning, finance, etc. (Umble et al. 2003; Jacobs and Bendoly 2003; Giachetti 2004). The main purpose is to increase the agility of manufacturing and to realize effective management of virtual enterprises. The importance of ERP is widely recognized. Many aspects of ERP approach have been studied, such as the process integration in an ERP system, relations with the electronic commerce, supply chain, and product data management (PDM) systems (Akkermans et al. 2003; Park and Kusiak 2005; Wang et al. 2005; Ou-Yang and Chang 2006).

Recently, the development of ERP and PDM systems has led effectively to a close-loop, integrated and systematic business approach, product life cycle management (PLM). It is expected that PLM comprehensively and consistently manages all stages of a product's life cycle (Thimm et al. 2006). The management involves all product data (in different forms, views, or granularity) and business processes, which commence from the market requirements to the final product disposal. The utmost objective is to realize a lean, sustainable, and profitable product development process through improved communication and negotiation among players from all product life cycle stages (Saaksvuori and Immonen 2005; Stark 2005). However, the engineering data integration with process models in PLM systems is still not well studied.

Application problems

Many problems persist in the implementation of concurrent and collaborative engineering. They can be categories into two groups. The first group is related to representation, processing, and management methods for engineering intent. Many decisions made in the product development and manufacturing processes are supported by engineering principles, concepts, and rules. Product models are the reflection of engineering intent and hence need to be verified based on such engineering intent. However, engineering intent is mainly represented as 'know-how' by individual engineers, or is only implicitly embedded in product data relations. The lack of intent representation affects the product validation processes. The second group is related to data mapping and change management. Concurrent and collaborative engineering uses separate but related applications. Theoretically, all CAx applications operate on their specific features mapped to a common set of data so that the product engineering and management is efficient with respect to changes. However, the available technologies have difficulties in maintaining globally consistent and comprehensive product models.

Existing industrial solutions

Many partial solutions to these problems were developed. For example, Unigraphics software streamlines commands to propagate changes of shared models among users within a collaboration environment (UGS 2006). However, such a data sharing mechanism is only supported when collaborators use the same software. To support real time solid model sharing and modifications among diverse CAx systems, 'OneSpace' collaboration software uses IGES or STEP as the common neutral format (CoCreate 2006) for all users. Product models with other data formats must be converted into a neutral format. In addition, only geometric data is shared. As analyzed in (Chen et al. 2006), sharing of non-geometric data, such as parameters, constraints, and features among different CAx systems is still an unresolved problem due to the lack of a commonly accepted representation scheme (Pratt et al. 2005). The widely used feature technology has the potential to solve these problems.

It should be noted that the nature of Product development stages is inter-related and mutually constraining. Different application models represent different aspects of the same product. When an application model is changed, the changes must be propagated to other related applications for checking and updating (Park and Khoshnevis 1993; Dohmen 1997; Ma and Tong 2003). Therefore, application models must be connected or integrated. Ideally, this task can be achieved by using feature technology. Many approaches have been proposed as shown in Table. These approaches are discussed in Sections "Application problems", "Existing industrial solutions", "Feature conversion" and "Feature recognition".

Feature conversion

Some researchers suggested using feature conversions to directly convert one feature model into another feature model (Bronsvoort and Jansen 1993; Gao 2004). This approach is supposed to maintain the consistency between feature models through direct connections. It can also use non-geometric information stored in the existing feature model to derive new feature models (Bronsvoort and Noort 2004). For example, Anderson and Chang use geometric reasoning to re-group design features into machining features (Anderson and Chang 1990). However, they assumed that part designs are created using only subtractive features, which can be regarded as primitive machining features. To convert protruding feature models to negative feature models, Gurumoorthy et al. used a clipping and classification method (Subramani et al. 2004; Subramani and Gurumoorthy 2005). They consider only geometric relations between feature models.

Feature conversions directly link feature models. However, the most common and also the most difficult task of feature conversion is to develop a consistent scheme accommodating different feature representations and variations even though they could be based on the same set of geometric entities (Shah 1988). The many-to-many relations between feature elements make direct feature conversion almost infeasible. The non-geometric information carried by the feature model must be used together with the geometric information to derive a new feature model. Otherwise, feature level operations, such as change propagation under a hybrid approach can be prohibitively troublesome.

Feature recognition

The feature recognition approach recognizes features from solid models. Many researchers focus on this issue (Henderson 1984; Joshi and Chang 1988; Regli 1994; Kim 1994; Tseng and Joshi 1994; Pal 2005). The multiple interpretations and feature interactions are main hurdles that hamper the use of this approach. Some researchers proposed to use engineering intent to solve these two problems (Vandenbrande and Requicha 1993; Gupta and Nau 1995; Gaines and Hayes 1999; Stage et al. 1999; Li et al. 2002; Raman and Marefat 2004). In other words, instead of pure geometric reasoning, features are recognized within a particular engineering context.

In addition, traditional feature recognition tools do not establish relations between the solid model and the recognized feature model. Any change in the solid model invalidates the whole recognized feature model. To solve this problem, Han proposed to use volumetric interference checking and a Justification-based Truth Maintenance System (JTMS) to manage geometric relations between design features and machining features (Han 1996). Whenever design features are modified, the machining feature model is updated accordingly. However, non-geometric relations between design and machining features should be recorded and maintained too. For example, without any geometric modifications, just a change to the tolerance specification of a design feature may invalidate the corresponding machining features (Zhou et al. 2002).

Common feature definition approach

Due to the difficulties of feature conversion and recognition approaches, it was proposed to use the same set of features for multiple applications as an alternative solution to application integration. This would save the effort of recognizing features and can also be used to support consistency control. The 'design-by-feature' approach can be regarded as belonging to this category. The initial purpose of the design-by-feature approach is to consider the downstream requirements in the design stage. When a common set of features are used for application integration, the maintenance of the consistency between design and application models is simplified. However, this approach imposes unreasonable limitations onto designers and is too rigid to deal with application integration. Another approach uses a set of neutral features, which are supported by all applications as for example STEP (ISO 1999). However, such hard-coded features can never be comprehensive and complete due to the varieties of applications. Currently, STEP standards mainly focus on geometric representations of machining features.

Multiple-view feature modeling

Feature conversion, feature recognition, and design-by-feature approaches create sequential relations between feature models. However, in concurrent engineering, multiple views need to be maintained simultaneously to keep them consistent. To meet this need, the multiple-view feature modeling approach has been proposed. In a multiple-view feature modeling approach, each feature model, which corresponds to a particular development stage, is a view of the whole product model. Two main methods to realize multiple-view feature modeling exist:

- *Pre-defined central model*: Jha and Gurumoorthy proposed a feature tree, which includes all the feature explanations of a product model, to achieve the application integration (Jha and Gurumoorthy 2000). Lee used a non-manifold topology to integrate design and analysis models (Lee 2005). Multiple levels of abstraction of each solid primitive are predefined in the corresponding feature classes and stored in a multi-dimensional model when features are generated. First, these two methods are geometric in nature, engineering intent is not represented. Second, the extensibility and applicability of these methods have to be proven since all feature interpretations or abstractions need to be predefined.
- *Incrementally updating a central model*: Bronsvoort et al. developed multiple-view feature models that encompass several product development stages. These feature models are integrated on the basis of a common cellular model (Bidarra et al. 1998). The cellular model is a non-manifold geometric model, which represents the combined product geometry from all stages (Dohmen et al. 1996). The relations among these views are established on the linking faces of features (de Kraker et al. 1997). Suh and Wozny proposed a set of fundamental features, which are actually faces, edges, and vertexes, as a common layer to support multiple applications (Suh and Wozny 1997). Application features are generated by analyzing and re-grouping these fundamental features. Martino et al. suggested using an intermediate model to integrate applications (Martino et al. 1998). The intermediate model consists of a set of common faces shared by features from different applications. Common faces are used to propagate modifications. These approaches consider only geometric relations between feature models.

It can be concluded that due to its unique nature, the feature concept is capable of linking knowledge to geometry as well as integrating applications. However, current feature technology is insufficient for these two purposes. In the next section, these insufficiencies are identified and listed as research issues.

Research issues

To fulfill the requirements of concurrent and collaborative engineering approaches, the shortcomings of the current feature technology must be addressed. These shortcomings are here presented as research issues.

Feature interoperability

Diverse feature definitions make the transfer or sharing of feature data among applications difficult. Traditional neutral feature definitions, such as (ISO 1999), focus on the representations of feature geometries. Neutral representations of parameters and constraints are still immature. In addition, engineering intent cannot be completely represented by parameters and numerical constraints. Recent development of STEP tries to extend those feature contents which can be neutrally represented (Pratt and Srinivasan 2005). However, whether to include all information into a single feature definition is desirable or not, is still uncertain (Bronsvoort and Noort 2004).

Feature association can be used to represent intent. An associative feature concept was proposed to represent relations between different forms of geometric entities depending on specific applications (Ma et al. 2003; Ma and Tong 2003, 2004). Associative features also model the evolvement of features at different stages of product development. The modifications made in one stage may affect the validity of the model in another stage. This associative feature concept can be extended to non-geometric associations. Furthermore, Ma et al. use assembly design features to represent essential relations between geometric entities, which may be assemblies, components, features, or faces (Ma et al. 2004a, 2007). These relations are design patterns generated in the conceptual design stage. Assembly design features can be regrouped and derived from specific viewpoints, such as functional design, assembly planning, or manufacturability analysis.

For process planning, the resource adaptive feature concept was proposed to represent associations between machining volumes and cutting tools, machines, and setups (Gaines and Hayes 1999; Stage et al. 1999; Raman and Marefat 2004). These resource adaptive feature definitions are not purely geometric. The machining specific entities, e.g. machines or cutting tools, are explicitly defined in feature classes as attributes or constraints. These associations are generated by a reasoning process, such as the machining sequence determination, setup determination, or tool selection process. These associations participate in the representation of process planners' intent.

More research efforts are needed to develop a neutral and complete representation of intent. Another issue is that the traditional neutral feature definitions are usually rigid.

Engineering intent representation and management

Numerical constraints (geometric or algebraic) are commonly used to represent design intent (Shah et al. 1994) and mostly with respect to detailed geometry. However, numerical constraints cannot represent design intent flexibly and completely. There are two reasons. The first is that numerical constraints represent only a subset of low level possible solutions in view of realizing design intent. Other solutions, working with high level collective templates, can embed engineering knowledge at high levels, should be investigated. The second reason is that certain relations cannot be represented by numerical constraints. For instance, two design features may be related because they are used together to realize a product function. Such inter-feature dependency relations are used to justify the presence or determine the properties values of features, but are too complicated to be expressed with numerical constraints.

Generally, so far, with today's technology engineering intent cannot be represented in a complete and explicit manner is still not available. In the real world, majority of the decisions are made in the conceptual design stage. The integration of conceptual design into the whole product model is a prerequisite for a comprehensive and explicit design intent representation (Chandrasekaran et al. 1993; Henderson 1993; Gui and Mantyla 1994). Two aspects of benefits can be expected. Firstly, downstream life cycle stages benefit from the concise and precise functional description of the product. Secondly, the time and effort for design modification or reuse can be reduced because alternative solutions are stored in the product model explicitly.

To represent the relations between product functions and physical structures, Kusiak et al. used a rule-based system to decompose customer requirements and functions, to map requirements to functions, and to record alternative solutions (Kusiak et al. 1991); other researchers identified the necessity of using part behavior to bridge the gap between abstract functions and physical objects (Welch and Dixon 1992; Umeda et al. 1996; Qian and Gero 1996). Their researches indicate that product functions represent by part behaviors model deeper design knowledge than geometry, topology, parameters, and constraints. In turn, each part behavior can be represented by its state transitions and is driven by part interactions.

However, the vagueness of the product geometry in the conceptual design stage creates an unresolved issue although it may be addressed by non-manifold geometric modelers and configuration spaces (Wong and Sriram 1993; Guan et al. 1997). Referring to Section "Conceptual design features (also called functional features)" about definitions of conceptual design features, it is clear that more research is needed in using feature technology to support conceptual design; linking conceptual design to other stages for sustainable engineering intent representation is far from complete.

In fact, DFX also expresses a kind of engineering intent. Some design specifications are intended to ease activities in downstream stages. For example, Thimm et al. (2004a,b) proposed a graph-based method for automatic machining sequence generation and tolerance analysis for rotational parts. A design dimension tree is used to generate ideal datum hierarchy trees and to measure real process plan efficiency. The structure of the datum hierarchy tree underlying a process plan is used to generate heuristics for more efficient design dimensioning schemes. Such heuristics serve the purpose of specifying better design dimension schemes for process planning and should be recorded in the product model.

Multiple-view geometric representation

Since different application features are associated and yet differently defined, their feature geometries may overlap. In addition, their representations may obey different geometric modeling requirements, such as using solid, surface, or wireframe representation. In a multiple-view feature-based product modeling scheme, incorporating these multi-faceted and diverse representations into a single geometric model and keeping them consistent are unresolved issues. The multidimensional, non-manifold cellular topology may lead to a solution. Following the pioneer work of Weiler (1988a,b), in which the radial edge structure was proposed to represent non-manifold geometries, the use of non-manifold geometric model to store canonical forms of the original objects, even if they are not on the final boundary, were discussed in Crocker and Reinke (1991); Masuda (1993). Non-manifold geometric models can be used to integrate applications (Sriram et al. 1995; Lee 2005). However, these publications did not fully apply the multi-dimensional, non-manifold topology to feature-based product modeling processes. Bidarra et al. proposed a cellular model to support multiple-view feature-based product modeling process (Bidarra et al. 1998 and 2005). Traditional B-rep and CSG usually represent only two-manifold solids. It is not easy to represent overlapping and multi-dimensional geometries in these schemes. This limitation makes them unsuitable to support application integration. The cellular topology, due to its unique data structure, overcomes afore mentioned limitations in geometry modeling; hence it is more suitable to serve the integration purpose. However, at present, the cellular topology has only been applied to 3D features. The cellular topology is also used for collaborative design (Wu and Sarma 2001; Lee et al. 2004) and efficient feature recognition (Woo 2003).

The multi-dimensional cellular topology has the capability to realize the geometric integration of feature-based applications. However, this has not been fully realized yet. In addition, the propagation of geometric modifications and in turn the maintenance of geometric consistency among different dimensional feature models are still research issues need to be addressed.

Persistent representation for non-geometric associations

The review in Section "Relations in a feature-based application model" shows that, for a single stage, non-geometric relations exist and are important. However, they are usually not well formulated and maintained for the reason that they often cannot be fully represented as numerical constraints. Furthermore, it is the engineering intent that actually controls the generation of a product model (Stage et al. 1999; Park 2003). The associations between engineering intent and the corresponding features have to be established and managed in order to allow product validation. Dependency networks provide a general solution. Kusiak and Wang (1995) used dependency relations to represent constraints on design variables. Four types of constraints are mentioned: equation, qualitative constraints, computer-based procedures, and influence rules. Park and Cutkosky (1999) used precedence, constraints, and abstraction links to model dependency relations during the collaborative engineering processes. Eastman (1996) analyzed how to use the dependency network to determine the influence scope of a design modification.

Non-geometric relations exist across the boundaries of application models. For example, associations between functional requirements in the conceptual design and geometric constraints or tolerance specifications in the detailed design must be maintained. Gorti and Sriram (1996) discussed the mapping of the functional relations among components to the corresponding spatial relations. In such a way, alternative spatial relations for the same functional relations can be found. Ranta et al. (1996) suggested generic ontologies to connect product development stages and discussed common function based associations among geometric entities as well as the possible mapping of abstract functional requirements to geometric constraints. Roy et al. explained that part specifications usually depend on product functional requirements (Roy et al. 2001; Roy and Bharadwaj 2002). Spatial relations cannot represent product functions completely. Energy transfer relations between part faces are essential in describing product functions and determining product specifications, such as fit types, tolerances, and surface finishes. Bronsvoort and Noort identified some non-geometric relations among feature models in a multiple view modeling system (Bronsvoort and Noort 2004). However, they use only geometric relations to connect detailed designs with assembly planning and manufacturing planning views. In addition, the connection mechanism between non-geometric data is unclear. In these publications, inter-stage non-geometric relations are usually established as direct links among specific entities. A more systematic and scalable method for sharing non-geometric data needs to be developed.

Hoffman and Arinyo proposed a generic architecture to establish and maintain associations between application models (Hoffman and Joan-Arinyo 1998, 2000). In this architecture, each client view deposits a part of its data into a master model and associates its private data to the public data. In the case of a modification, the master model notifies the associated clients, which are responsible for maintaining the consistency. However, no implementation details are given.

Unifying different application features

Generally, features have two parts: a geometric representation and a non-geometric representation. The geometric representation can be unified based on the cellular topology based modeling schemes. The unification of representations of engineering intent is more difficult because it is much more application specific. A generic, complete, and flexible feature definition, which can represent the commonalities of different application features, is needed.

Integrating knowledge-based methods with CAx tools

A full integration of KBE with CAx systems has not materialized yet (Hoffman and Joan-Arinyo 2000; Ma and Tong 2003; Roller and Kreuz 2003). Problems encountered include (1) Representing engineering intent using KBE; (2) Using engineering knowledge to drive product modeling or process planning; (3) Associating engineering knowledge with product designs or process plans; and (4) One-way control mechanism only from KBE systems to CAx systems, e.g. from knowledge to decisions, such as product configurations or machining methods; using engineering knowledge to verify product designs or process plans is not fully studied.

Establishing and maintaining non-geometric relations

The first challenge is to identify the major intra- and inter non-geometric relations in feature-based, multiple-view product models. The second challenge is to find suitable methods to represent and manage these non-geometric relations for the purpose of maintaining the validity and consistency of product models. With the addition of non-geometric relations, the efficiency with which modifications are propagated under a multiple-view product modeling approach becomes a challenge.

The authors' view of a new paradigm

Feature technology has been widely accepted as an effective means to implement engineering patterns to interface users and computer models in CAx systems. In CAD/CAM systems, feature technology is capable to bridge engineering semantics and geometry. However, from the viewpoint of maintaining the validity and consistency of product models, current feature technology still has severe shortcomings:

- Different feature definitions and their data structures make transferring and sharing data between applications difficult. A generic and flexible data structure is necessary.
- Engineering intent is not uniformly represented and maintained.
    - (i) The engineering intent proper to each application needs to be identified.
    - (ii) A method to represent and maintain engineering intent in product models is needed.
- Keeping the associated geometric representations of feature models consistent is difficult due to the different characteristics of feature geometry in distinct CAx applications.
- Non-geometric relations within and among application models are not generically represented and exploited.
- The lack of methods that determine the influence scope of any modification, especially across development stages. This shortcoming makes change propagation inefficient.

Based on the literature review, the authors believe that a feature-based unification approach is most suitable to bridge engineering semantics and product geometry and eventually integrate multiple applications. The key concept of the proposed approach is to establish a common theoretical frame work based on a generic feature definition as the parent class based on the object-oriented philosophy and develop the infrastructure that can deal with different features and abstract-levels of information in a coherent manner. This frame work can guide the implementation of effective interfaces among different applications via necessary feature level mappings to achieve feature level interoperability automatically. Ideally, a unified systematic CAx integration approach supporting applications across product life cycle stages. The feature-based interoperability scheme and mechanisms connect levels information of different granularity. Information association and unification are the strategies in this approach. With complete and explicit data associations, tedious, error-prone, and usually manual engineering change management in CAx systems require can be avoided. Unification is necessary because diverse CAx systems coexist and have different data structures.

Extending feature definition to support engineering intent embedment and data sharing among applications

Traditional features are specifically defined for certain computer applications, and used mainly to represent parameterized geometric shapes. This is insufficient. First, a generic, flexible, and scalable feature definition is needed such that a common object class data structure can be used for different applications (Ma 2005). Second, besides geometry, non-geometric feature data needs to be generically defined. The authors propose that a generic feature definition should be used as an intermediate information layer to associate product geometry and engineering knowledge consistently and explicitly. Here, 'generic' emphasize the 'abstracted' class definition that support polymorphic definitions of features such that systematic processing methods, including repository services, validating checking routines, and change propagation, can be shared and automated. This idea is supported by some research works related to associative or resource adaptive features (Raman and Marefat 2004; Ma and Tong 2003; Ma et al. 2007; Kim et al. 2004), and extended feature types and ontology schemes reported (Pratt et al. 2005;

Brunetti and Grimm 2005). However, none seems to have reached maturity.

Engineering intent representation and management in product models

Engineering intent representation was researched on for some time already but a concrete foundation has not been established yet: methods to link conceptual designs to detailed designs (Ranta et al. 1996; Brunetti and Golob 2000, Bronsvoort and Noort 2004) were preliminarily explored while some conceptual frameworks were proposed for the integration mechanisms between knowledge engineering methods and computer aided product design tools (Penoyer et al. 2000; Roller and Kreuz 2003).

Inter-stage non-geometric relations

Non-geometric relations among stages are barely addressed. These relations need to be identified and managed because they are crucial for consistency control among product models. Feature-level associations can be used to represent these non-geometric relations.

Multi-dimensional geometric modeling

To accommodate different types of features (e.g. non-manifold and of varying dimensional degrees) multi-dimensional geometric models must be integrated. As discussed in the earlier section of "Extending feature definition to support engineering intent embedment and data sharing among applications", the multi-dimensional cellular model can accommodate different geometric representation and modeling requirements uniformly. It can represent intermediate or overlapping geometries. It can also be used to keep the original forms of features regardless whether they are on the final boundary of the part or not. The feasibility of this approach has been studied in (Lee 2005; Bidarra et al. 2005).

Representation of dependency relations

Complete dependency relations must be maintained for effective and efficient management of engineering changes. A dependency network can be used to record and manage the dependency relations in product modeling processes. A truth maintenance system approach (Han 1996) can accommodate different types of dependency relations uniformly, such as numerical constraints, antecedent to consequent relations in engineering rules, and feature objects dependency relations in feature recognition or conversion processes.

Conclusions

This paper has provided a review of the existing research works on the integration of systems across different product life cycle stages. It has also identifies research issues for solving the existing problems. It identifies features and related technology as the likely best approach to solve many of the problems incurring during the integration of CAD and other tools used in the development, manufacturing, and other product life cycle stages. Literature related to geometric and non-geometric features, their use in applications, the maintenance of their consistency within and across applications is reviewed and therefore the 'state of the art' in concurrent and collaborative engineering is established. Finally, unresolved issues and research challenges are highlighted and a new paradigm for product modeling and manufacturing, feature-based concurrent and collaborative engineering, has been proposed. The proposed approach intends to extend the traditional feature concept to a flexible and enriched data type, which can be used to support the validity maintenance of product models. This approach is able to support data associations and propagation of modifications across product development life cycle stages. Ideally, it can be used to improve the feature level interoperability in future virtual enterprises and collaborative engineering.

References

Anantha, R., Kramer, G. A., & Crawford, R. H. (1996). Assembly modeling by geometric constraint satisfaction. *Computer-Aided Design, 28*(9), 707–722.

Anderl, R., & Mendgen, R. (1996). Modelling with constraints: Theoretical foundation and application. *Computer-Aided Design, 28*(3), 301–313.

Anderson, D. C., & Chang, T. C. (1990). Geometric reasoning in feature-based design and process planning. *Computers & Graphics, 14*(2), 225–235.

Akkermans, H. A., Bogerd, P., Yücesan, E., & van Wassenhove, L. N. (2003). The impact of ERP on supply chain management: Exploratory findings from a European Delphi study. *European Journal of Operational Research, 146*(2), 284–301.

Benko, P., Martin, R. R., & Varady, T. (2001). Algorithms for reverse engineering boundary representation models. *Computer-Aided Design, 33*(11), 839–851.

Bidarra, R., & Bronsvoort, W. F. (2000). Semantic feature modeling. *Computer-Aided Design, 32*(3), 201–225.

Bidarra, R., de Kraker, K. J., & Bronsvoort, W. F. (1998). Representation and management of feature information in a cellular model. *Computer-Aided Design, 30*(4), 301–313.

Bidarra, R., Dohmen, M., & Bronsvoort, W. F. (1997). Automatic detection of interactions in feature models. In *Proceedings of the 1997 ASME design engineering technical conferences*.

Bidarra, R., Madeira, J., Neels, W. J., & Bronsvoort, W. F. (2005). Efficiency of boundary evaluation for a cellular model. *Computer-Aided Design, 37*(12), 1266–1284.

Bidarra, R., van den Berg, E., & Bronsvoort, W. F. (2001). Collaborative modeling with features. In *Proceedings of the 2001 ASME design engineering technical conferences*.

Boothroyd, G., Dewhurst, P., & Knight, W. (2002). *Product design for manufacture and assembly* (2nd ed.). Marcel Dekker Inc.

Bronsvoort, W. F., & Jansen, F. W. (1993). Feature modelling and conversion–Key concepts to concurrent engineering. *Computers in Industry, 21*(1), 61–86.

Bronsvoort, W. F., & Noort, A. (2004). Multiple-view feature modeling for integral product development. *Computer-Aided Design, 36*(10), 929–946.

Brown, D. C. (2002). Functional, behavioral and structural features. In *Proceedings of KIC5, 5th IFIP WG5.2 Workshop on Knowledge Intensive CAD*.

Brunetti, G., & Golob, B. (2000). A feature-based approach towards an integrated product model including conceptual design information. *Computer-Aided Design, 32*(14), 877–887.

Brunetti, G., & Grimm, S. (2005). Feature ontologies for the explicit representation of shape semantics. *International Journal of Computer Applications in Technology, 23*(2/3/4), 192–202.

Brunetti, G., Martino, T. D., Falcidieno, B., & HaBinger, S. (1995). A relational model for interactive manipulation of form features based on algebraic geometry. In *Proceedings of the Third ACM Symposium on Solid Modeling and Applications*.

Case, K., & Harun, W. A. W. (2000). Feature-based representation for manufacturing planning. *International Journal of Production Research, 38*, 4285–4300.

Chan, C. K., & Tan, S. T. (2003). Generating assembly features onto split solid models. *Computer-Aided Design, 35*(14), 1315–1336.

Chan, F. T. S., Zhang, J., Lau, H. C. W., & Ning, A. (2000). Information integration platform for CIMS. In *Proceedings of the 2000 IEEE International Conference on Management of Innovation and Technology*.

Chandrasekaran, B., Goel, A. K., & Iwasaki, Y. (1993). Functional representation as design rationale. *Computer, 26*(1), 48–56.

Chen, G., Ma, Y. S., Thimm, G., & Tang, S. H. (2006). Associations in a unified modeling scheme. *ASME Transactions, Journal of Computing and Information Science in Engineering, 6*(2), 114–126.

Chen, L., Pu, J., & Wang, X. K. (2002). A general model for machinable features and its application to machinability evaluation of mechanical parts. *Computer-Aided Design, 34*(3), 239–249.

Chopra, S., & Meindl, P. (2007). *Supply chain management* (3rd ed.). Prentice Hall.

Chu, C. C. P., & Gadh, R. (1996). Feature-based approach for set-up minimization of process design from product design. *Computer-Aided Design, 28*(5), 321–332.

CoCreate Software Incorporation. (2006). http://www.cocreate.com.

Crocker, G. A., & Reinke, W. F. (1991). An editable nonmanifold boundary representation. *IEEE Computer Graphics & Applications, 11*(2), 39–51.

Csabai, A., Stroud, I., & Xirouchakis, P. C. (2002). Container spaces and functional features for top–down 3D layout design. *Computer-Aided Design, 34*(13), 1011–1035.

Cunningham, J. J., & Dixon, J. R. (1988). Designing with features: The origin of features. In *Proceedings of ASME Computers in Engineering Conference*.

de Kraker, K. J., Dohmen, M., & Bronsvoort, W. F. (1997). Maintaining multiple views in feature modeling. In *The 4th Symposium on Solid Modeling and Applications*.

Deneux, D. (1999). Introduction to assembly features: An illustrated synthesis methodology. *Journal of Intelligent Manufacturing, 10*(1), 29–39.

Deng, Y. M., Britton, G. A., Lam, Y. C., Tor, S. B., & Ma, Y. S. (2002). Feature-based CAD-CAE integration model for injection-moulded product design. *International Journal of Production Research, 40*(15), 3737–3750.

Dohmen, M. (1997). Constraint-based feature validation, Ph.D. Thesis, Delft University of Technology.

Dohmen, M., de Kraker, K. J., & Bronsvoort, W. F. (1996). Feature validation in a multiple-view modeling system. In *Proceedings of the 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference*.

Eastman, C. M. (1996). Managing integrity in design information flows. *Computer-Aided Design, 28*(6/7), 551–565.

Faheem, W., Hayes, C. C., Castano, J. F., & Gaines, D. M. (1998). What is a manufacturing interaction? In *Proceedings of the 1998 ASME Design Engineering Technical Conferences*.

Fazio, T. L. D., Rhee, S. J., & Whitney, D. E. (1999). Design-specific approach to design for assembly (DFA) for complex mechanical assemblies. *IEEE Transactions on Robotics and Automation, 15*(5), 869–881.

Feng, C. X., Huang, C. C., Kusiak, A., & Li, P. G. (1996). Representation of functions and features in detail design. *Computer-Aided Design, 28*(12), 961–971.

Feng, S. C., & Song, E. Y. (2000). Information modeling of conceptual process planning integrated with conceptual design. In *The 2000 ASME Design Engineering Technical Conferences*.

Fuh, J. Y. H., & Chang, C. H., & Melkanoff, M. A. (1996). The development of an integrated and intelligent CAD/CAPP/CAFP environment using logic-based reasoning. *Computer-Aided Design, 28*(3), 217–232.

Fuh, J. Y. H., & Li, W. D. (2005). Advances in collaborative CAD: The-state-of-the art. *Computer-Aided Design, 37*(5), 571–581.

Gaines, D. M., & Hayes, C. C. (1999). CUSTOM-CUT: A customizable feature recognizer. *Computer-Aided Design, 31*(2), 85–100.

Gao, J., Zheng, D. T., & Gindy, N. (2004). Mathematical representation of feature conversion for CAD/CAM system integration. *Robotics & Computer-Integrated Manufacturing, 20*(5), 457–467.

Geelink, R., Salomons, O. W, van Slooten, F., van Houten, F. J. A. M., & Kals, H. J. J. (1995). Unified feature definition for feature based design and feature based manufacturing. In *Proceedings of the ASME International Computers in Engineering Conference*.

Giachetti, R. E. (2004). A framework to review the information integration of the enterprise. *International Journal of Production Research, 42*(6), 1147–1166.

Gorti, S. R., & Sriram, R. D. (1996). From symbol to form: A framework for conceptual design. *Computer-Aided Design, 28*(11), 853–870.

Gossard, D. C., Zuffante, R. R., & Sakurai, H. (1988). Representing dimensions, tolerances, and features in MCAE systems. *IEEE Computer Graphics and Applications, 8*(2), 51–59.

Guan, X., Duffy, A. H. B., & Maccallum, K. J. (1997). Prototype system for supporting the incremental modelling of vague geometric configurations. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 11*(4), 287–310.

Gui, J. K., & Mantyla, M. (1994). Functional understanding of assembly modeling. *Computer-Aided Design, 26*(6), 435–451.

Gupta, S. K., & Nau, D. S. (1995). Systematic approach to analyzing the manufacturability of machined parts. *Computer-Aided Design, 27*(5), 323–342.

Han, J. H. (1996). 3D geometric reasoning algorithms for feature recognition, Ph.D. Thesis, University of Southern California.

Han, J. H., & Requicha, A. A. G. (1998). Feature recognition from CAD models. *IEEE Computer Graphics and Applications, 18*(2), 80–94.

Henderson, M. R. (1984). Extraction of Feature Information from Three-Dimensional CAD Data, Ph.D. Thesis, Purdue University.

Henderson, M. R. (1993). Representing functionality and design intent in product models. In *Proceedings of the Second ACM Symposium on Solid Modeling and Applications*.

Hoffman, C. M. (1989). *Geometric and solid modeling: An introduction*. Morgan Kaufmann.

Hoffman, C. M., & Joan-Arinyo, R. (1998). CAD and the product master model. *Computer-Aided Design, 30*(11), 905–918.

Hoffman, C. M., & Joan-Arinyo, R. (2000). Distributed maintenance of multiple product views. *Computer-Aided Design, 32*(7), 421–431.

Hounsell, M. S., & Case, K. (1999). Feature-based interaction: An identification and classification methodology. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 213*(4), 369–380.

ISO. (1999). Industrial automation systems and integration—product data representation and exchange—part 224: Application protocol: Mechanical product definition for process planning using machining features, ISO 10303-224:1999. Geneva, Switzerland: International Organization for Standardization (ISO).

ISO. (2000). Industrial automation systems and integration—product data representation and exchange—part 42: Integrated generic resource: Geometric and topological representation, ISO 10303-42:2000. Geneva, Switzerland: International Organization for Standardization (ISO).

Jacobs, F. R., & Bendoly, E. (2003). Enterprise resource planning: Developments and directions for operations management research. *European Journal of Operational Research, 146*(2), 233–240.

Jha, K., & Gurumoorthy, B. (2000). Automatic propagation of feature modification across domains. *Computer-Aided Design, 32*(12), 691–706.

Joshi, S., & Chang, T. C. (1988). Graph-based heuristics for recognition of machined features from a 3D solid model. *Computer-Aided Design, 20*(2), 58–66.

Karinthi, R. R., & Nau, D. (1992). An algebraic approach to feature interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 14*(4), 469–484.

Khoshnevis, B., Sormaz, D. N., & Park, J. Y. (1999). An integrated process planning system using feature reasoning and space search-based optimization. *IIE Transactions, 31*(7), 597–616.

Kim, Y. S. (1994). Volumetric feature recognition using convex decomposition. In J. J. Shah, M. Mantyla, & D. S. Nau (Eds.), *Advances in feature based manufacturing, manufacturing research and technology* (pp. 39–63). Elsevier Science.

Kim, K. Y., Manley, D. G., & Yang, H. J. (2006). Ontology-based assembly design and information sharing for collaborative product development. *Computer-Aided Design, 38*(12), 1233–1250.

Kim, K. Y., Wang, Y., Muogboh, O. S., & Nnaji, B. O. (2004). Design formalism for collaborative assembly design. *Computer-Aided Design, 36*(9), 849–871.

Kramer, T. R., Huang, H., Messina, E., Proctor, F. M., & Scott, H. (2001). A feature-based inspection and machining system. *Computer-Aided Design, 33*(9), 653–669.

Kumar, A. S., Nee, A. Y. C., & Prombanpong, S. (1992). Expert fixture-design system for an automated manufacturing environment. *Computer-Aided Design, 24*(6), 316–326.

Kung, H. K., Du, T. C. T., & Weng, C. H. (1999). Applying object-oriented database technologies in concurrent design processes. *International Journal of Computer Integrated Manufacturing, 12*(3), 251–264.

Kusiak, A., Szczerbicki, E., & Park, K. (1991). A novel approach to decomposition of design specifications and search for solutions. *International Journal of Production Research, 29*(7), 1391–1406.

Kusiak, A., & Wang, J. (1995). Dependency analysis in constraint negotiation. *IEEE Transactions on Systems, Man, and Cybernetics, 25*(9), 1301–1313.

Laakko, T., & Mantyla, M. (1993). Feature modeling by incremental feature recognition. *Computer-Aided Design, 25*(8), 479–492.

Lee, S. H. (2005). A CAD-CAE integration approach using feature-based multi-resolution and multi-abstraction modeler techniques. *Computer-Aided Design, 37*(9), 941–955.

Lee, R. S., Chen, Y. M., & Lee, C. Z. (1997). Development of a concurrent mold design system: A knowledge-based approach. *Computer Integrated Manufacturing Systems, 10*(4), 287–307.

Lee, J. Y., Han, S. B., Kim, H., & Park, S. B. (1999). Network-centric feature-based modeling. In *Proceedings of the Seventh Pacific Conference on Computer Graphics and Applications*.

Lee, J. Y., & Kim, K. (1996). Geometric reasoning for knowledge-based parametric design using graph representation. *Computer-Aided Design, 28*(10), 831–841.

Lee, J. Y., Lee, J. H., Kim, H., & Kim, H. S. (2004). A cellular topology-based approach to generating progressive solid models from feature-centric models. *Computer-Aided Design, 36*(3), 217–229.

Li, C. L. (2005). Automatic layout design of plastic injection mould cooling system. *Computer-Aided Design, 37*(7), 645–662.

Li, W. D., Ong, S. K., Fuh, J. Y. H., Wong, Y. S., Lu, Y. Q., & Nee, A. Y. C. (2004). Feature-based design in a distributed and collaborative environment. *Computer-Aided Design, 36*(9), 775–797.

Li, W. D., Ong, S. K., & Nee, A. Y, C. (2002). Recognizing manufacturing features from a design-by-feature model. *Computer-Aided Design, 34*(11), 849–868.

Lockett, H. L., & Guenov, M. D. (2005). Graph-based feature recognition for injection moulding based on a mid-surface approach. *Computer-Aided Design, 37*(2), 251–262.

Ma, Y. S. (2005). A case study on non-parametric design method in ODM collaborative product development. *International Journal of Product Development, 2*(4), 411–434.

Ma, Y. S., Britton, G. A., Tor, S. B., & Jin, L. Y. (2007). Associative assembly design features: Concept, implementation and application. *International Journal of Advanced Manufacturing Technology, 32*(3), 434–444.

Ma, Y. S., Britton, G. A., Tor, S. B., Jin, L. Y., Chen, G., & Tang, S. H. (2004a). Design of a feature-object-based mechanical assembly library. *Computer-Aided Design & Applications, 1*(1–4), 397–404.

Ma, Y. S., & Tong, T. (2003). Associative feature modeling for concurrent engineering integration. *Computers in Industry, 51*(1), 51–71.

Ma, Y. S., & Tong, T. (2004). An object-oriented design tool for associative cooling channels in plastic-injection moulds. *International Journal of Advanced Manufacturing Technology, 23*(1–2), 79–86.

Ma, Y. S., Tor, S. B., & Britton, G. A. (2003). The development of a standard component library for plastic injection mould design using an object-oriented approach. *International Journal of Advanced Manufacturing Technology, 22*(9–10), 611–618.

Ma, W. Y., Zhong, Y. M., Tso, S. K., & Zhou, T. X. (2004b). A hierarchically structured and constraint-based data model for intuitive and precise solid modeling in a virtual reality environment. *Computer-Aided Design, 36*(10), 903–928.

Mandorli, F., Cugini, U., Otto, H. E., & Kimura, F. (1997). Modeling with self validation features. In *Proceedings of ACM/IEEE Symposium on Solid Modeling and Applications'97*.

Marefat, M., & Britanik, J. (1997). Case-based process planning using an object-oriented model representation. *Robotics & Computer-Integrated Manufacturing, 13*(3), 229–251.

Martino, T. D., Falcidieno, B., Giannini, F., Hassinger, S., & Ovtcharova, J. (1994). Feature-based modeler by integrating design and recognition approaches. *Computer-Aided Design, 26*(8), 646–653.

Martino, T. D., Falcidieno, B., & Habinger, S. (1998). Design and engineering process integration through a multiple view intermediate modeler in a distributed object-oriented system environment. *Computer-Aided Design, 30*(6), 437–452.

Masuda, H. (1993). Topological operators and Boolean operations for complex-based nonmanifold geometric models. *Computer-Aided Design, 25*(2), 119–129.

Mcginnis, B. D., & Ullman, D. G. (1992). The evolution of commitments in the design of a component. *Journal of Mechanical Design, 114*(1), 1–7.

Mok, C. K., Chin, K. S., & Ho, J. K. L. (2001). An interactive knowledge-based CAD system for mould design in injection moulding processes. *International Journal of Advanced Manufacturing Technology 17*(1), 27–38.

Mukherjee, A., & Liu, C. R. (1997). Conceptual design, manufacturability evaluation and preliminary process planning using function-form relationships in stamped metal parts. *Robotics & Computer-Integrated Manufacturing, 13*(3), 253–270.

Myung, S., & Han, S. (2001). Knowledge-based parametric design of mechanical products based on configuration design method. *Expert Systems with Applications, 21*(2), 99–107.

Noort, A., Hoek, G. F. M., & Bronsvoort, W. F. (2002). Integrating part and assembly modeling. *Computer-Aided Design, 34*(12), 899–912.

Ong, S. K., & Chew, L. C. (2000). Evaluating the manufacturability of machined parts and their setup plans. *International Journal of Production Research, 38*(11), 2397–2415.

Oral, A., & Cakir, M. C. (2004). Automated cutting tool selection and cutting tool sequence optimization for rotational parts. *Robotics & Computer-Integrated Manufacturing, 20*(2), 127–141.

Otto, H. E. (2001). From concepts to consistent object specifications: Translation of a domain-oriented feature framework into practice. *Journal of Computer Science & Technology, 16*(3), 208–230.

Ou-Yang, C., & Chang, M. J. (2006). Developing an agent-based PDM/ERP collaboration system. *International Journal of Advanced Manufacturing Technology, 30*(3–4), 369–384.

Pahng, F., Senin, N., & Wallace, D. (1998). Distribution modeling and evaluation of product design problems. *Computer-Aided Design, 30*(6), 411–423.

Pal, P., Tigga, A. M., & Kumar, A. (2005). Feature extraction from large CAD databases using genetic algorithm. *Computer-Aided Design, 37*(5), 545–558.

Park, S. C. (2003). Knowledge capturing methodology in process planning. *Computer-Aided Design, 35*(12), 1109–1117.

Park, H., & Cutkosky, M. R. (1999). Framework for modeling dependencies in collaborative engineering processes. *Research in Engineering Design, 11*(2), 84–102.

Park, J. Y., & Khoshnevis, B. (1993). A real-time computer-aided process planning system as a support tool for economic product design. *Journal of Manufacturing Systems, 12*(2), 181–193.

Park, K., & Kusiak, A. (2005). Enterprise resource planning (ERP) operations support system for maintaining process integration. *International Journal of Production Research, 43*(19), 3959–3982.

Penoyer, J. A., Burnett, G., Fawcett, D. J., & Liou, S. Y. (2000). Knowledge based product life cycle systems: Principles of integration of KBE and C3P. *Computer-Aided Design, 32*(5–6), 311–320.

Prasad, B. (1996). Concurrent engineering fundamentals: Integrated product and process organization. Prentice Hall.

Pratt, M. J., Anderson, B. D., & Ranger, T. (2005). Towards the standardized exchange of parameterized feature-based CAD models. *Computer-Aided Design, 37*(12), 1251–1265.

Pratt, M. J., & Srinivasan, V. (2005). Towards a neutral specification of geometric features. *International Journal of Computer Applications in Technology, 23*(2/3/4), 203–218.

Qian, L., & Gero, J. S. (1996). Function-behavior-structure paths and their role in analogy-based design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing 10*(4), 289–312.

Raman, R., & Marefat, M. M. (2004). Integrated process planning using tool/process capabilities and heuristic search. *Journal of Intelligent Manufacturing, 15*(2), 141–174.

Ranta, M., Mantyla, M., Umeda, Y., & Tomiyama, T. (1996). Integration of functional and feature-based product modeling—the IMS/GNOSIS experience. *Computer-Aided Design, 28*(5), 371–381.

Regli, W. C. (1994). Geometric algorithms for recognition of features from solid models, Ph.D. Thesis, University of Maryland.

Regli, W. C., & Pratt, M. J. (1996). What are feature interactions? In *Proceedings of the 1996 ASME Design Engineering Technical Conference and Computers in Engineering Conference*.

Requicha, A. A. G. (1980). Representations for rigid solids: Theory, methods, and systems. *ACM Computing Surveys (CSUR), 12*(4), 437–464.

Rezayat, M. (1996). Midsurface abstraction from 3D solid models: General theory and applications. *Computer-Aided Design, 28*(11), 905–915.

Roller, D., & Kreuz, I. (2003). Selecting and parameterising components using knowledge based configuration and a heuristic that learns and forgets. *Computer-Aided Design, 35*(12), 1085–1098.

Rosenman, M. A., & Gero, J. S. (1996). Modelling multiple views of design objects in a collaborative CAD environment. *Computer-Aided Design, 28*(3), 193–205.

Rosenman, M., & Wang, F. J. (1999). CADOM: A component agent-based design-oriented model for collaborative design. *Research in Engineering Design, 11*(4), 193–205.

Rossignac, J. R. (1990). Issues on feature-based editing and interrogation of solid models. *Computers & Graphics, 14*(2), 149–172.

Roucoules, L., Salomons, O., & Paris, H. (2003). Process planning as an integration of knowledge in the detailed design phase. *International Journal of Computer Integrated Manufacturing, 16*(1), 25–37.

Roy, U., & Bharadwaj, B. (2002). Design with part behaviors: Behavior model, representation and applications. *Computer-Aided Design, 34*(9), 613–636.

Roy, U., & Liu, C. R. (1998). Feature-based representational scheme of a solid modeler for providing dimensioning and tolerancing information. *Robotics & Computer-Integrated Manufacturing, 4*(3/4), 335–345.

Roy, U., Pramanik, N., Sudarsan, R., Sriram, R. D., & Lyons, K. W. (2001). Function-to-form mapping: Model, representation and applications in design synthesis. *Computer-Aided Design, 33*(10), 699–719.

Saaksvuori, A., & Immonen, A. (2005). *Product lifecycle* management (2nd ed.). Springer.

Schulte, M., Weber, C., & Stark, R. (1993). Functional features for design in mechanical engineering. *Computers in Industry, 23*(1–2), 15–24.

Shah, J. J. (1988). Feature transformations between application-specific feature spaces. *Journal of Computer-Aided Engineering, 5*(6), 247–255.

Shah, J. J. (1991). Conceptual Development of Form Features and Feature Modelers. *Research in Engineering Design, 2*(2), 93–108.

Shah, J. J., Ali, A., & Rogers, M. T. (1994). Investigation of declarative feature modeling. In *Proceedings of the ASME'94 Computers in Engineering*.

Shah, J. J., & Mantyla, M. (1995). *Parametric and feature-based CAD/CAM: Concepts, techniques, and applications*. John Wiley and Sons, Inc.

Shah, J. J., & Rogers, M. T. (1993). Assembly modeling as an extension of feature-based design. *Research in Engineering Design, 5*(3–4), 218–237.

Sharma, V., & Hayes, C. C. (2001). Operation ordering principles and intra-setup planner: Combining human control with automation in process planning. In *Proceedings of the 2001 ASME Design Engineering Technical Conferences*.

Shyamsundar, N., & Gadh, R. (2001). Internet-based collaborative product design with assembly features and virtual design spaces. *Computer-Aided Design, 33*(9), 637–651.

Shyamsundar, N., & Gadh, R. (2002). Collaborative virtual prototyping of product assemblies over the Internet. *Computer-Aided Design, 34*(10), 755–768.

Sormaz, D. N., & Khoshnevis, B. (1997). Process planning knowledge representation using an object-oriented data model. *International Journal of Computer Integrated Manufacturing, 10*(1–4), 92–104.

Sormaz, D. N., & Khoshnevis, B. (2003). Generation of alternative process plans in integrated manufacturing systems. *Journal of Intelligent Manufacturing, 14*(6), 509–526.

Sriram, R. D., Wong, A., & He, L. X. (1995). GNOMES: An object-oriented nonmanifold geometric engine. *Computer-Aided Design, 27*(11), 853–868.

Stage, R., Roberts, C., & Henderson, M. (1999). Generating resource based flexible form manufacturing features through objective driven clustering. *Computer-Aided Design, 31*(2), 119–130.

Stark, J. (2005). *Product lifecycle management–21st century paradigm for product realization*. Springer.

Starly, B., Lau, A., Sun, W., Lau, W., & Bradbury, T. (2005). Direct slicing of STEP based NURBS models for layered manufacturing. *Computer-Aided Design, 37*(4), 387–397.

Stefano, P. D. (1997). Automatic extraction of form features for casting. *Computer-Aided Design, 29*(11), 761–770.

Stefano, P. D., Bianconi, F., & Angelo, L. D. (2004). An approach for feature semantics recognition in geometric models. *Computer-Aided Design, 36*(10), 993–1009.

Subramani, S., & Gurumoorthy, B. (2005). Maintaining associativity between form feature models. *Computer-Aided Design, 37*(13), 1319–1334.

Subramani, S., Nalluri, S. R. P. R., & Gurumoorthy, B. (2004). 3D clipping algorithm for feature mapping across domains. *Computer-Aided Design, 36*(8), 701–721.

Suh, Y. S., & Wozny, M. J. (1997). Interactive Feature Extraction for a Form Feature Conversion System. In *The 4th Symposium on Solid Modeling and Applications*.

Thimm, G., Britton, G. A., & Fok, S. C. (2004a). A graph theoretic approach linking design dimensioning and process planning, Part 1: Designing to process planning. *International Journal of Advanced Manufacturing Technology, 24*(3–4), 261–271.

Thimm, G., Britton, G. A., & Fok, S. C. (2004b). A graph theoretic approach linking design dimensioning and process planning, Part 2: Design heuristics for rotational parts. *International Journal of Advanced Manufacturing Technology, 24*(3–4), 272–278.

Thimm, G., Lee, S. G., & Ma, Y. S. (2006). Towards unified modeling of product life-cycles. *Computes in Industry, 57*(4), 331–341.

Tor, S. B., Britton, G. A., Zhang, W. Y., & Deng, Y. M. (2002). Guiding functional design of mechanical products through rule-based causal behavioural reasoning. *International Journal of Production Research, 40*(3), 667–682.

Tseng, Y. J., & Joshi, S. B. (1994). Recognizing multiple interpretations of interacting machining features. *Computer-Aided Design, 26*(9), 667–688.

UGS Corporation. (2006). http://www.ugs.com.

Umble, E. J., Haft, R. R., & Umble, M. M. (2003). Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operational Research, 146*(2), 241–257.

Umeda, Y., Ishii, M., Yoshioka, M., Shimomura, Y., & Tomiyama, T. (1996). Supporting conceptual design based on the function-behavior-state modeler. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 10*(4), 275–288.

van Holland, W., & Bronsvoort, W. F. (2000). Assembly features in modeling and planning. *Robotics & Computer Integrated Manufacturing, 16*(4), 277–294.

Vandenbrande, J. H., & Requicha, A. A. G. (1993). Spatial reasoning for the automatic recognition of machinable features in solid models. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 15*(12), 1269–1285.

Varady, T., Martin, R. R., & Cox, J. (1997). Reverse engineering of geometric models—an introduction. *Computer-Aided Design, 29*(4), 255–268.

Venkataraman, S., Shah, J. J., & Summers, J. D. (2001). An investigation of integrating design by features and feature recognition. In *IFIP conference*, FEATS.

Vieira, A. S. (1995). Consistency management in feature-based parametric design. In *Proceedings of the 1995 ASME Design Engineering Technical Conferences*.

Wang, C. B., Chen, T. Y., Chen, Y. M., & Chu, H. C. (2005). Design of a meta model for integrating enterprise systems. *Computers in Industry 56*(3), 305–322.

Wang, L. H., Shen, W. M., Xie, H., Neelamkavil, J., & Pardasani, A. (2002). Collaborative conceptual design—state of the art and future trends. *Computer-Aided Design, 34*(13), 981–996.

Weiler, K. (1988a). The radial edge structure: A topological representation for non-manifold geometric boundary modeling. In M. J. Wozny, H. W. McLaughlin, & J. L. Encarnacao (Eds.), *Geometric modeling for CAD applications* (pp. 3–36). North-Holland.

Weiler, K. (1988b). Boundary graph operators for non-manifold geometric modeling topology representations. In M. J. Wozny, H. W. McLaughlin, & J. L. Encarnacao (Eds.), *Geometric modeling for CAD applications* (pp. 37–66). North-Holland.

Welch, R. V., & Dixon, J. R. (1992). Representing function, behavior and structure during conceptual design. In *Design theory and methodology—DTM'92*.

Whitney, D. E., Mantripragada, R., Adams, J. D., & Rhee, S. J. (1999). Designing assemblies. *Research in Engineering Design, 11*(4), 229–253.

Wilson, P. R., & Pratt, M. J. (1988). A taxonomy of features for solid modeling. In M. J. Wozny, H. W. McLaughlin, & J. L. Encarnacao (Eds.), *Geometric modeling for CAD applications* (pp. 125–136). North-Holland.

Wong, A., & Sriram, D. (1993). SHARED: An information model for cooperative product development. *Research in Engineering Design, 5*(1), 21–39.

Woo, Y. (2003). Fast cell-based decomposition and applications to solid modeling. *Computer-Aided Design, 35*(11), 969–977.

Wood, S. L., & Ullman, D. G. (1996). The functions of plastic injection moulding features. *Design Studies, 17*(2), 201–213.

Wu, D., & Sarma, R. (2001). Dynamic segmentation and incremental editing of boundary representations in a collaborative design environment. In *Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications*.

Xue, D., & Yang, H. (2004). A concurrent engineering-oriented design database representation model. *Computer-Aided Design, 36*(10), 947–965.

Xue, D., Yadav, S., & Norrie, D. H. (1999). Knowledge base and database representation for intelligent concurrent design. *Computer-Aided Design, 31*(2), 131–145.

Yang, H., & Xue, D. (2003). Recent research on developing Web-based manufacturing systems: A review. *International Journal of Production Research, 41*(15), 3601–3629.

Zha, X. F., Du, H. J., & Qiu, J. H. (2001a). Knowledge-based approach and system for assembly oriented design, Part I: The approach. *Engineering Applications of Artificial Intelligence, 14*(1), 61–75.

Zha, X. F., Du, H. J., & Qiu, J. H. (2001b). Knowledge-based approach and system for assembly oriented design, Part II: The system implementation. *Engineering Applications of Artificial Intelligence, 14*(2), 239–254.

Zhang, F., & Xue, D. (2002). Distributed database and knowledge base modeling for concurrent design. *Computer-Aided Design, 34*(1), 27–40.

Zhou, F., Kuo, T. C., Huang, S. H., & Zhang, H. C. (2002). Form feature and tolerance transfer from a 3D model to a set-up planning system. *International Journal of Advanced Manufacturing Technology, 19*(2), 88–96.