Int. J. Computer Applications in Technology, Vol. X, No. Y, XXXX

Fine grain associative feature reasoning in collaborative engineering

Yong-Sheng Ma*

Department of Mechanical Engineering,
University of Alberta,
4-9 Mechanical Engineering Building,
Edmonton, Alberta T6G 2G8, Canada
Fax: + 1 780 492 2200
Email: yongsheng.ma@ualberta.ca
*Corresponding author

C.H. Bong

Singapore Precision Engineering and
Tooling Association,
114 Balestier Road 329679, Singapore
Fax: + 65 6292 4517
Email: chunhian@ualberta.ca

Abstract: This paper explores the vast domain of systematic collaborative engineering with reference to product lifecycle management approach from the angle of feature-level collaboration among partners. A new method of fine grain feature association modelling and reasoning is proposed. The original contribution is on the explicit modelling and reasoning of collaborative feature relations within a dynamic context. A case study has been carried out to illustrate the interweaving feature relations in collaborative oil-rig space management and the effective application of such relations modelled in design solution optimisation.

Keywords: fine-grain associations; feature-based collaboration; collaborative engineering; CADCAM; intelligent design.

Reference to this paper should be made as follows: Ma, Y-S. and Bong, C.H. (XXXX) 'Fine grain associative feature reasoning in collaborative engineering', *Int. J. Computer Applications in Technology*, Vol. X, No. Y, pp.xxx–xxx.

Biographical notes: Y-S. Ma has been a tenured Associate Professor at University of Alberta since 2007. He received his BEng from Tsinghua University in Beijing (1986), and both MSc and PhD degrees from UMIST, UK, in 1990 and 1994, respectively. Before 2007, he had been working in Singapore as an Associate Professor of Nanyang Technological University (NTU) (2000–2007), a Group Manager at Singapore Institute of Manufacturing Technology (1996–2000) and a Lecturer at Ngee Ann Polytechnic (1993–1996). His research areas include CADCAM, product lifecycle management, feature-based product and process modelling.

C.H. Bong is currently working at Singapore Precision Engineering and Tooling Association (SPETA) as a Project Engineer. He was a student under the supervision of Dr. Y-S. Ma from 2006 to 2007 at Nanyang Technological University (NTU). He conducted the preliminary study in piping space management in oil-rig design.

1 Introduction

Product lifecycle management has been an active research domain in recent years but its scientific framework and key supporting theories are still evolving with different schools of thoughts. Many researchers and developers extended design and manufacturing information systems to cover the full spectrum of product lifecycles. Such effort can be represented by some commercial software packages, such as Teamcenter (Siemens AG, 2007), Windchill (PTC.com, 2007), etc. Saaksvuori and Immonen (2005) have systematically described this approach. Although such systems and studies have advanced engineering system integration coverage to a broad business management domain supported with databases and network technologies, the key challenging problems, such as semantic interoperability, detailed engineering constraint management, effective and efficient change management, still left unresolved. This paper looks into these challenges from a totally different angle. Instead of building an engineering information system by integrating different application systems piece by piece, a complete and open product lifecycle repository system supporting multiple applications, systems and stages is proposed (Ma et al., 2007a). The key contribution of this method to the field of research is the unlimited information grain size of application integration. The characteristics created with such fine grain system integration are the full support of associative features within and across different applications or lifecycle stages; constraints can be flexibly created and managed. In turn, the product model can be seamlessly integrated with engineering process model in a smart modelling manner. Then detailed business and engineering intelligence could be embedded into the system's integrated product and process model. Eventually, pervasive and intelligent collaborative engineering can be enabled.

2 Literature review

Concurrent engineering promote engineering consideration about all lifecycle issues in parallel across different stages (Prasad, 1997). The traditional sequential mode of product development has been changed into an iterative and evolutionary mode. Product development becomes evolvement of design and manufacturing processes via tight application integration and parallel engineering. At the same time, collaborative engineering approach has grown significantly to support distributed, multi-discipline and multi-organisation teams during the product development processes. This approach is motivated by the globalisation of economy and boosted by the development of the internet (Wang et al., 2002; Fuh and Li, 2005; Yang and Zhang, 2006). In general, systematic product and process modelling methods developed are useful for both concurrent and collaborative engineering. In fact, these two business approaches have been blended into a global business management trend and create a more system oriented concurrent and collaborative approach. Concurrent and collaborative engineering deals with either separate or integrated applications via associations. To support such inter-weaving, integrated, and complex engineering systems, informatics modelling plays an essential role.

When studying individual engineering areas, such as design, manufacturing and management processes, Knowledge-Based Engineering (KBE) is a common practice in many CAx systems to support decision making, such as functional design and process planning, etc. (Zha et al., 2001; Tor et al., 2002; Park, 2003).

Fundamentally, system integration is a problem of information sharing and management. As to the contents of information, it can be largely classified into product-related data and process-related data. Within the domain of product-related data, two categories exist, i.e. geometric and non-geometric. Traditionally, application integration was based on the geometric data sharing. For example, integrations between product design and tooling design, reverse engineering, rapid prototyping, computerised numerical machining control, coordinate measuring machine, mesh generation for CAE, virtual reality systems, etc. have been widely studied (Kramer et al., 2001; Deng et al., 2002). STEP and IGES standards have been developed for this purpose. However, they are aimed to achieve the required interoperability mainly for geometric information. To allow early design anticipation and later change management across the product lifecycle stages, ever more close value chains are being formed in modern collaborative engineering, and then non-geometric information are shared and used while the intellectual properties must be surely warranted. Such industrial applications demand a systematic approach to enable the interoperability for managing not only product or process geometric entities but also their related constraints and semantics. The authors propose a unified associative feature-based approach.

3 Theoretical exploration

3.1 From feature templates to associative features

The concept of features is flexible and can be used in many aspects of mechanical engineering. There are various definitions about features for different application domains. Representatives of feature definitions include a region of interest in a part model (Wilson and Pratt, 1988), any geometric form or entity that is used in reasoning in one or more design or manufacturing activities (Cunningham et al., 1996), generic shapes associated to certain properties or attributes and knowledge useful in reasoning about the product (Shah, 1991), regions of an object that are meaningful for a specific activity or application (Vandenbrande and Requicha, 1993), a set of form elements with a functional meaning in a given application context that allows an association between shape and functionality (Martino et al., 1998), a representation of shape aspects of a product that can be mapped to a generic shape and are functionally significant for some product lifecycle phase (Bidarra and Bronsvoort, 2000), etc. These definitions reveal that features have two fundamental characteristics: being related to product information in a higher level than geometric and topological entities; and representing engineering semantics.

However, in the current CADCAM technology, features are generally parametric patterns of 'basic unit of knowledge'. Features are defined in a 'fixed' pattern that can be defined with a group of 'fixed' constraints. A new concept, named 'associative feature' was introduced (Ma and Tong, 2003) suggesting a new method of feature modelling where a type of continuously changing features can be defined in object-oriented manner, and the feature properties and behaviours evolve as the associative feature evolves. As stated in Ma and Tong (2003) that associative features should have the following key characteristics: (1) built-in associative links to its related geometric entities; (2) self-validation for the consistency of its entities, attributes, constraints, etc.; (3) methods available for constructing, storing, indexing, editing and destroying its instances; (4) methods that can be expanded to interface with query and execution mechanisms for high-level knowledge processes; (5) methods to interface with other engineering application tools. The conceptual representation of associative feature is very much related to the definitions of 'generic feature' and 'generic constraint' as described by Ma et al. (2007a).

It is a theoretical advancement to suggest that features are kept valid and associated throughout a product lifecycle instead of within just one stage or one application of the lifecycle, such as detailed product design. A well-defined feature object type (or a class) has to be developed to cover the various stages of lifecycles of different mechanical components in a very generic yet abstracted form. More recently, the concept of associative assembly features (Ma et al., 2007b) has extended the 'broadness' of *associative features* to cover assembly patterns with good scalability across components or assembly members. The successful implementation proofs that associative features can be applied to serve as the intermediate layer of information representation to interface the tedious CAD geometry and the related attribute creation and manipulation methods with high-level knowledge management and engineering. Associative feature approach can effectively enhance traditional feature-based technology in three aspects: design change management by features, semantic modelling for engineering rules and engineering models' reuse. Then, it is clear that the next natural research work of this technology is to develop the generic mechanisms of capturing, storing and retrieving engineering knowledge with the support of associative features because if the effort is successful, a full cluster of associative features can be *persistently modelled* to support product lifecycle management. To achieve this goal, more effective interoperability among computer systems is essential.

3.2 Existing standards supporting interoperability

Interoperability can be described as the exchangeability and usability of data types and related information between two or among more systems or components. Interoperability among engineering software tools is in high demand due to globalisation and value chain integration. In the past, interoperability has always been a 'problem' because data formats and their embedded data structures (schemas) become a means of protection of commercial interests. To date, interoperability has only been achieved at the data level. IGES and STEP-based interoperability depends upon geometry modelling standardisation. Mere geometric entities are converted from one format to another based on the common B-Rep and CSG frameworks. For example, STEP has obvious limitations on semantics and high-level entity types such as user-defined features and constraints. If a parametric part is created in Pro-E with features, such as counter-bore holes, chamfers, or boss cylinders, after being exported into a STEP file and imported into UGS NX, the model becomes a fixed solid block. The observation from the informatics point of view can reveal that the information grain size is at the 'file level' when data exchanges take place. In addition, the contents of the files are partially translated due to the inconsistent definitions of semantic entities.

3.3 Engineering intent representation and management

Theoretically, all CAx applications should operate on a common set of data so that the product engineering and management can be effective and efficient to manage changes required. However, as reviewed above, in reality, different applications have difficulties in sharing a consistent product model space because of their different semantic representations and derived variations.

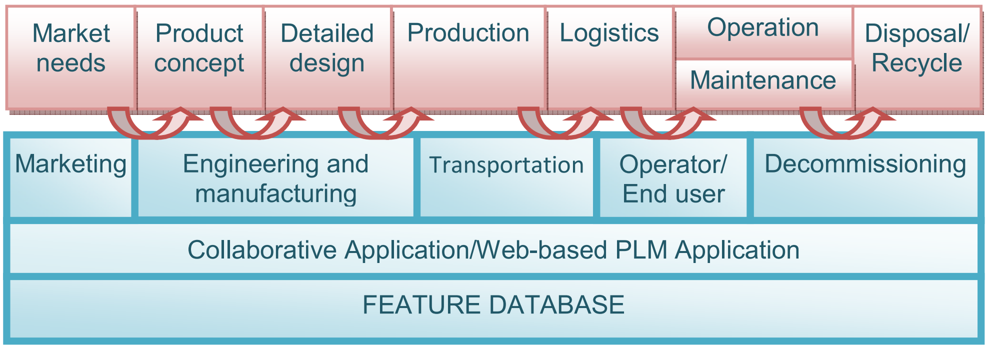
The representation and processing methods for engineering intent depends on the collective grouping of constraints and the reasoning or optimisation methods. Many decisions made in the product development and manufacturing processes are supported by engineering principles, concepts, and rules. However, engineering intent, such as 'know-how', has been only implicitly embedded in product data relations. So far, very limited works are done in constraint management that is associated to engineering intent. The lack of intent representation has affected the product validation processes. Developing a unified and associative collaborative engineering platform is a challenge in engineering informatics.

When engineering intent is represented explicitly by the collective groups of constraints and entities, due to the continuing evolvement and changes along the product lifecycle, such constraints have to be maintained systematically to keep its validity. This requirement involves data consistency and validation checking methods. It can be appreciated that for a product, there is a master product model. All other related models, such as analysis models, manufacturing models, tooling models, quality models, and even the MRP and ERP models, are either directly based on or associated to the master model. Figure 1 gives an information structure that shows the coverage of an ideal PLM system and the supporting sub-systems according to the authors. Engineering intent expressed by well-defined relations can be explicitly represented and managed via database technology. In general, there exists a continued flow of engineering or business intent throughout the stages of a product lifecycle. When the application scope is scaled from a small enterprise to a bigger one, or even to a full-scale OEM like Boeing and GM, the configurations of products and lifecycles to be managed could become tremendously important and very complicated. Then there is an issue of scalability too.

3.4 Fine grain interoperability and associations

In recent years, the concept of 'open' data formats or source codes is gaining acceptance due to customer demands for collaboration in the global arena. Data-oriented functions, such as creating a solid in CAD systems, have become a common expectation. Currently, the interoperability among different systems is confined by the accessible neutral-information grain size. In order to facilitate engineering collaboration, therefore, the research of interoperability at the level of semantic knowledge becomes imperative. This topic covers the informatics modelling for semantic information sharing, mapping, manipulation, conversion and knowledge-based reasoning and automation (Bronsvoort and Noort, 2004; Gao et al., 2004; Pratt and Srinivasan, 2005). To solve this problem, a scheme like 'the valuing system' of 'trading currencies' in a large market is needed. For higher level semantic information association and sharing, a common and flexible standardised scheme is required but not yet available (Kim et al., 2006; Ou-Yang and Chang, 2006; Ouertani and Gzara, 2007).

Figure 1 Information model structure for a fine grain feature-based collaborative engineering (see online version for colours)



The authors champion associative, fine-grain feature-based modelling approach. Fine grain associations refer to the relations created or used for certain engineering purpose among engineering entities without the limitation of access, even to entities below 3D solid or part level. It is tied closely to the associative feature concept that has been introduced by the authors in early publications (Ma and Tong, 2003). To generalise the effort, the engineering intent has to be explicitly modelled and 'materialised' in the form of engineering associative features which consist of lower level entities, constraints and reasoning methods. In other words, generic engineering intent can be modelled in the form of a set of *live objects* according to the principles of software informatics while the collective properties and behaviour can be dynamically created and managed by creating the associated objects across different grain sizes, keeping the relations persistently, managing their applied methods, and evolving their statuses or stages. Then many dynamic changing scenarios with the effective context support can be illustrated. By managing these scenarios, intelligent design and manufacturing can be achieved.

Towards to this direction, a preliminary case study on the space management of pipelines in oil-rig construction industry is demonstrated. The above theoretical points are further explained with the example.

4 Case study

This study focuses on 'pipeline feature' modelling supported with fine-grain 'context' extraction and optimisation of space management in oil-rig design. It is an extension to a student's research project of parametric computer-aided design with Solidworks. Due to the limitation of resources, the project did not involve multiple CAD systems. The prototype system can generate an oil-rig model with the programming toolkit. Associative context space has been modelled via C++ programming and dynamically derived via CAD API functions. The research element in this project is how space management can be automatically and intelligently optimised via the associative feature object methods and an algorithm interacting with the surrounding context. Then pipeline planning and preliminary 3D layout management are tested interactively with the support of a grid-based optimisation algorithm. The key research impact is on the method for associative design feature modelling, generation and management with constraints. Each instance of the associative pipeline feature can be automatically created, analysed and semi-automatically optimised in stages with the developed program. According to the authors' knowledge, no similar previous work has been reported.

Similar to the cooling channel modelling reported in early publication (Ma and Tong, 2003), the pipeline of an installed oil-rig system is modelled as a functioning object with the following characteristics: (1) The pipeline is modelled in a continuously evolving associative feature with flexible behaviours defined with well-defined constraints. The essential member entities are the connection segments and attached interfacing mechanical elements that form a connected 'path' from a start position to the destination position. This path is modelled as a set of connected line and arc segment in 3D space. Each segment has the starting point and vector, ending point and vector, as well as the connectivity constraints. Such constraints are added along the process of concept development by the designers interactively. When a pipeline is initially designed, a simplified pattern is automatically generated with the built-in constraints of segment connectivity, inlet and outlet ending flange geometry, and initial given raw material lengths. (2) Within a given space envelope interested, the existing parts or systems are searched and extracted for verification. (3) Some engineering rules are built into the design algorithm by specifying optimisation constraints, e.g. '*the minimum crossing space between pipelines has to be more than x meters*', '*the use of lower space is prioritised for the ease of maintenance*', '*reserving maximum space for operational use*', etc.

To represent the space occupation and for the ease of space analysis, a 3D space-grid method to model the interested space environment is developed. To simply explain the concept, a 2D concept is shown in Figure 2. The grid elements are represented by a binary array and the availability is simply represented by 'True' of 'False' Boolean value. The grid size can be adjusted according to the required resolution or the scale of the interested space. The grids are also very easy to be indexed and analysed to derive the available space clouds, with the detailed distribution and topological neighbourhood search. With a reference coordinate system, the minimum or maximum space locations can be easily determined via a few simple iterative functions. Then it can be built into the software to search for the possible paths that a pipeline can be installed with the minimum space due to engineering requirement. For illustration purpose, again, a 2D path via available grids is shown in Figure 3. Clearly, more often than not, there are many solutions, and optimisation based on the selection criteria is necessary. First, characteristic attributes or properties of the pipeline paths have to be modelled and analysed. Then the 'associative pipeline' feature object class can be developed. While different solutions determined by an algorithm automatically are feasible objects, or 'options' for the designers to select interactively, based on merits of different solutions, the contents of the pipeline are captured gradually such that the object instance becomes more and more materialised or 'solidified'.

Figure 2 2D schematic representation of space occupation and the path patterns (see online version for colours)

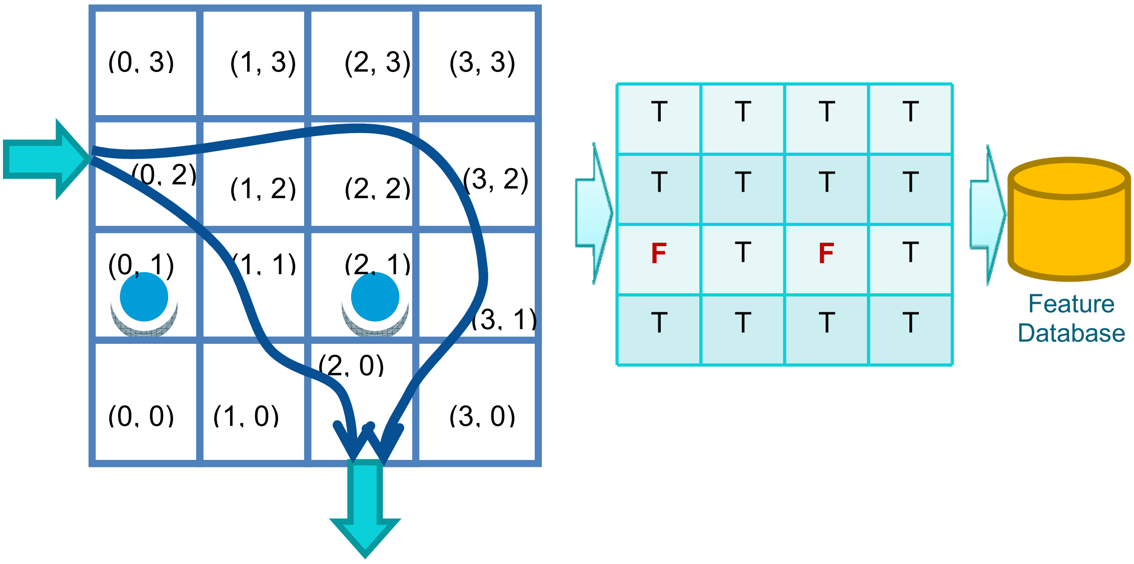
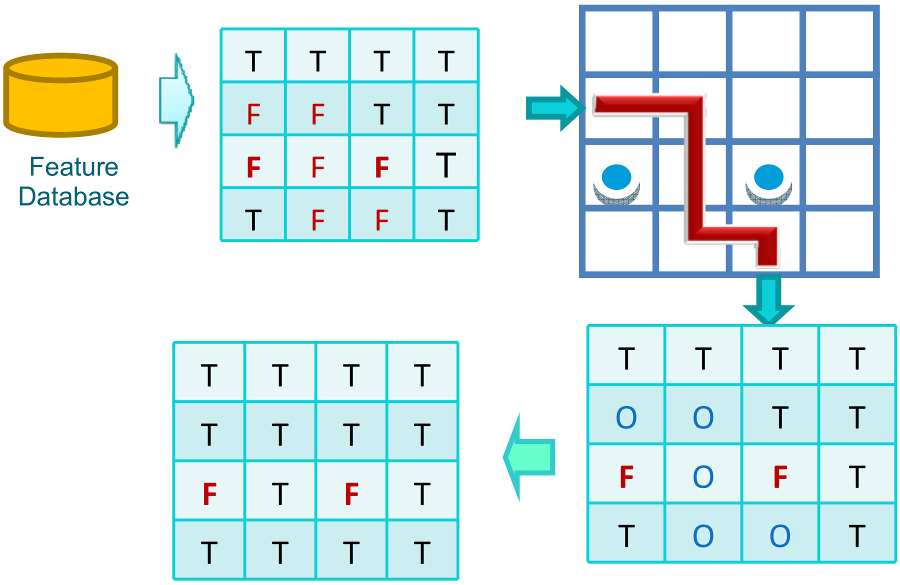


Figure 3 Simplified pipeline path searching method in 2D illustration (see online version for colours)



By adding one more dimension in the searching and optimisation algorithm, as shown in Figure 4, the concept of pipeline feature has been extended into 3D pattern with associated parameters, characteristic attributes, built in constraints and the intelligent generation and editing methods. As to the surrounding context environment, the design space can be initialised as shown in Figure 5.

The surrounding geometry elements, such as neighbouring faces of the interested space, are identified after cycling the entire product model with a few searching and analysis routines such that only the relevant entities are clustered into the neighbouring entity list. Then their volumetric portions which fall in the interested space scope are extracted. Note that *fine grain* access to context space geometric elements is required.

Like the 2D grids used to represent certain areas, 3D volumetric ‘grids’ are generated throughout the space of the context environment and classified as either ‘occupied’ or ‘available’ ones. By running the optimisation routines, a ‘best’ concept of pipeline layout can be determined and then the full solid representation in the form of solid pipes (including flanges and connection interfaces if necessary but not shown here) are generated automatically. In similar manner, a new piece of equipment can be evaluated to fit into a space predefined and its location and orientation are optimised if feasible. Such space management algorithm can be repeated again and again whenever a new pipeline or a piece of equipment is to be inserted according to some built-in space management strategies. Since most of the on-board equipment items are purchased and installed by the collaborating partners and subject to replacement by other competitive suppliers, this case study is representative in real collaborative engineering situations as the authors observed in Keppel Offshore & Marine Ltd in Singapore.

Figure 4 An example pipeline path in three orthogonal views (see online version for colours)

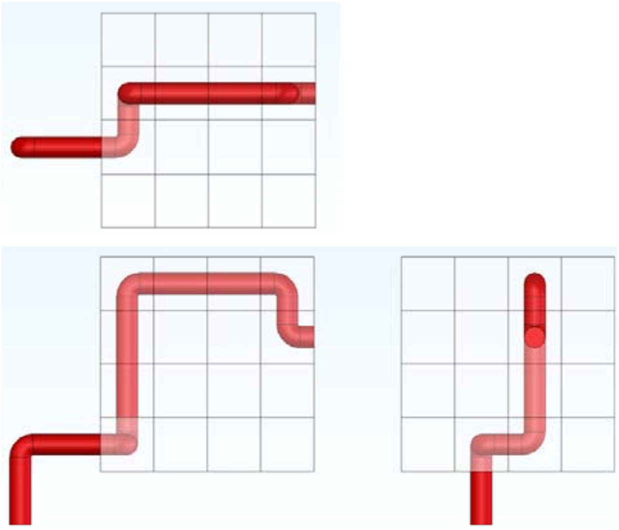
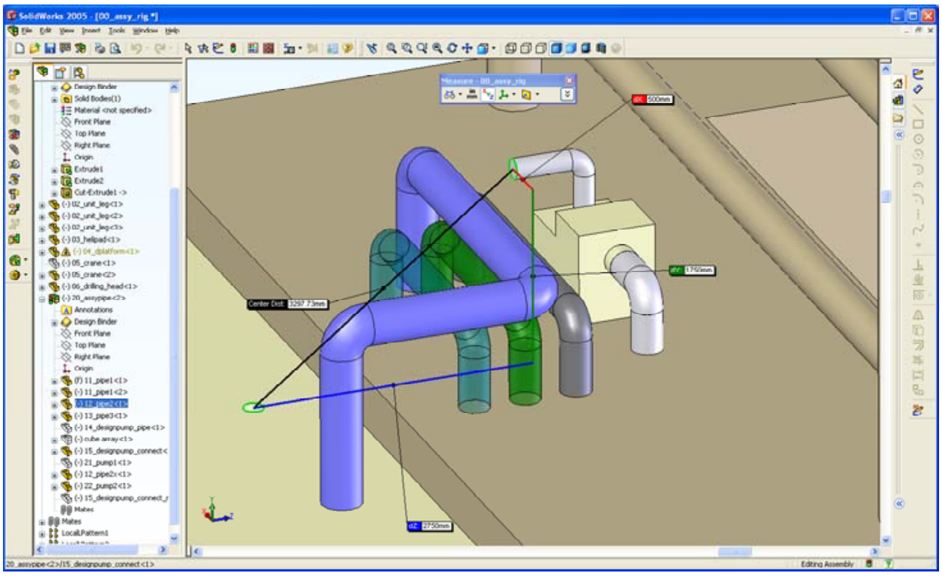


Figure 5 Initial space state for the design of pipeline path (see online version for colours)



There could be different space management searching conditions and the process can be interactively controlled by the designer or automated either partially or fully depending on the trade off of productivity and the flexibility of design practice. For example, the pipeline inlet and outlet positions can be both predefined, unlimited, or limited by some orientations or positional allowances. In Figure 6, a candidate solution is generated. However, the pipeline has been put *under* the existing pipelines and it is then rejected by the designer due to its difficulty of by-passing in construction or future maintenance. Figure 7 offers an accepted pipeline path design. The authors are aware of that there are commercial solutions available for the generation of pipelines or cables trays, etc. However, the main point of this paper is to highlight the context reasoning requirement and the algorithm working with associative features and fine grain access to the context geometries as shown in this case study.

Figure 6 A candidate path solution (see online version for colours)

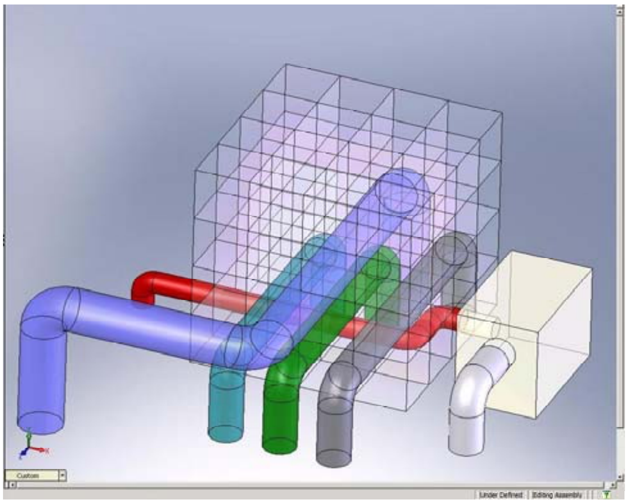
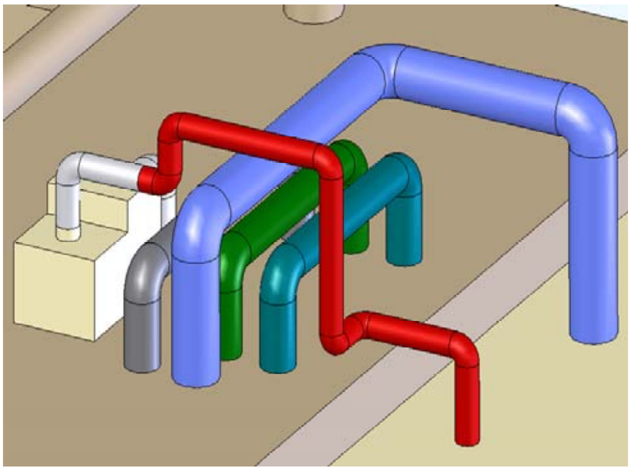


Figure 7 One of the acceptable pipeline paths generated (see online version for colours)



5 Conclusions

Engineering activities evolve gradually from conceptual level to detailed level; from abstract to complete forms; from unknown to known; and from prototypes to matured products and processes. Engineering activities are associated by shared engineering knowledge and methods with the support of reasoning and decision making from the user or its agents. Traditional engineering IT solutions with the legacy of 3D CADCAM systems have encountered explosive data volume and complicated data consistency problems for system integration. Although Product Data Management (PDM) systems have been developed to track engineering drawings and product model integrity, but their information grain size is at the 3D solid part level. Such information grain size does not address detailed semantic relations during the design and manufacturing processes and hence PDM systems end up into managing huge product databases without effective methods to manage intricate engineering knowledge and methods.

This paper emphasises the associative feature modeling in engineering informatics. Context-based and fine grain information access is essential for supporting intelligent and dynamic design and manufacturing processes. A fine grain associative feature application scenario has been illustrated by an example prototype program developed to design oil-rag models via certain automated API applications. Effective pipeline feature definition, generation and editing interfaces as well as the program algorithms are discussed. The novelty of this research is that the associative feature scheme is able to support fine-grain feature-level associations and propagation of modifications across product lifecycle stages with the precondition that a unified product modelling database is in place. Associative features provide an intermediate information layer to bridge the gap between engineering knowledge and product geometry. They are also used to maintain geometric and non-geometric relations across product models. The feasibility of the proposed scheme is demonstrated with the prototype system and a case study.

Acknowledgements

The author would like to acknowledge the contribution of Mr. Tan Siong Wei, who was a student of Nanyang Technological University, in the form of industrial attachment report and the background information obtained from Keppel Offshore & Marine Ltd and PPL Shipyard Pte Ltd of Singapore which motivated the case study used in this study.

References

Cunningham, T.W., Mantripragada, R., Lee, D., Thornton, A.C. and Whitney, D. (1996) 'Definition, analysis and planning of a flexible assembly process', *Proceedings of the ASMA Japan/USA Symposium on Flexible Automation*, Boston, MA.

Bidarra, R. and Bronsvoort, W.F. (2000) 'Semantic feature modeling', *Computer Aided Design*, Vol. 32, No. 3, pp.201–225.

Bronsvoort, W.F. and Noort, A. (2004) 'Multiple-view feature modeling for integral product development', *Computer-Aided Design*, Vol. 36, No. 10, pp.929–946.

Deng, Y-M., Britton, G.A., Lam, Y.C., Tor, S.B. and Ma, Y-S. (2002) 'Feature-based CAD-CAE integration model for injection-moulded product design', *International Journal of Production Research*, Vol. 40, No. 15, pp.3737–3750.

Fuh, J.Y.H. and Li, W.D. (2005) 'Advances in collaborative CAD: the-state-of-the art', *Computer Aided Design*, Vol. 37, No. 5, pp.571–581.

Gao, J., Zheng, D.T. and Gindy, N. (2004) 'Extraction of machining features for CAD/CAM integration', *International Journal of Advanced Manufacturing Technology*, Vol. 24, Nos 7–8, pp.573–581.

Kim, K.Y., Manley, D.G. and Yang, H.J. (2006) 'Ontology-based assembly design and information sharing for collaborative product development', *Computer-Aided Design*, Vol. 38, No. 12, pp.1233–1250.

Kramer, T.R., Huang, H., Messina, E., Proctor, F.M. and Scott, H. (2001) 'A feature-based inspection and machining system', *Computer-Aided Design*, Vol. 33, No. 9, pp.653–669.

Martino, T.D., Falcidieno, B. and Habinger, S. (1998) 'Design and engineering process integration through a multiple view intermediate modeler in a distributed object-oriented system environment', *Computer-Aided Design*, Vol. 30, No. 6, pp.437–452.

Ma, Y-S. and Tong, T. (2003) Associative feature modeling for concurrent engineering integration, *Computers in Industry*, Vol. 51, No. 1, pp.51–71.

Ma, Y-S., Tang, S-H. and Chen, G. (2007a) 'A fine-grain and feature-oriented product database for collaborative engineering', in Li, W.D., Ong, S.K., Nee, A.Y.C. and McMahon C. (Eds): *Collaborative Product Design and Manufacturing Methodologies and Applications*, Springer, London, pp.109–136.

Ma, Y-S., Britton, G.A., Tor, S.B. and Jin, L.Y. (2007b) 'Associative assembly design features: concept, implementation and application', *International Journal of Advanced Manufacturing Technology*, Vol. 32, Nos 5–6, pp.434–444.

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

Ouertani, M.Z. and Gzara, L. (2007) 'Tracking product specification dependencies in collaborative design for conflict management', *Computer-Aided Design*, Vol. 40, No. 7, pp.828–837.

Park, S.C. (2003) 'Knowledge capturing methodology in process planning', *Computer Aided Design*, Vol. 35, No. 12, pp.1109–1117.

Prasad, B. (1997) *Concurrent Engineering Fundamentals, Vol. II: Integrated Product Development*, Prentice Hall, USA, New York.

Pratt, M.J., and Srinivasan, V. (2005) 'Towards a neutral specification of geometric features', *International Journal of Computer Applications in Technology*, Vol. 23, Nos 2–4, pp.203–218.

PTC.com (2007) Available online at: http://www.ptc.com/appserver/mkt/products/home.jsp?k=37

Saaksvuori, A. and Immonen, A. (2005) *Product Lifecycle Management*, 2nd ed., Springer, Berlin.

Shah, J.J. (1991) 'Conceptual development of form features and feature modelers', *Research in Engineering Design*, Vol. 2, pp.93–108.

Siemens AG (2007) Available online at: http://www.plm.automation.siemens.com/en_us/products/teamcenter/

Tor, S.B., Britton, G.A., Zhang, W.Y. and Deng, Y-M. (2002) 'Guiding functional design of mechanical products through rule-based causal behavioural reasoning', *International Journal of Production Research*, Vol. 40, No. 3, pp.667–682.

Vandenbrande, J.H. and 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*, Vol. 15, No. 12, pp.1269–1285.

Wang, L.H., Shen, W.M., Xie, H., Neelamkavil, J. and Pardasani, A. (2002) 'Collaborative conceptual design – state of the art and future trends', *Computer-Aided Design*, Vol. 34, No. 13, pp.981–996.

Wilson, P. and Pratt, M. (1988) A taxonomy of features for solid modeling', in Wozny, M.J., McLaughlin, H.W. and Encarnacao, J.L. (Eds): *Geometric Modeling for CAD Applications*, Elsevier, North Holland, IFIP, pp.125–136.

Yang, Q.Z. and Zhang, Y. (2006) 'Semantic interoperability in building design: methods and tools', *Computer-Aided Design*, Vol. 38, No. 10, pp.1099–1112.

Zha, X.F., Du, H.J. and Qiu, J.H. (2001) 'Knowledge-based approach and system for assembly oriented design I: the approach', *Engineering Applications of Artificial Intelligence*, Vol. 14, No. 1, pp.61–75.