Collaborative Computer-Aided Design – Research and Development Status

W. D. Li[1], Y. Q. Lu[2], W. F. Lu[3], J. Y. H. Fuh[4] and Y. S. Wong[5]

[1]Singapore Institute of Manufacturing Technology, wdli@simtech.a-star.edu.sg
[2]Singapore Institute of Manufacturing Technology, yqlu@simtech.a-star.edu.sg
[3]Singapore Institute of Manufacturing Technology, wflu@simtech.a-star.edu.sg
[4]National University of Singapore, mpefuhyh@nus.edu.sg
[4]National University of Singapore, mpewys@nus.edu.sg

ABSTRACT

In order to facilitate product design and realization processes, presently, research is actively carried out for developing methodologies and technologies of collaborative computer-aided design systems to support design teams geographically dispersed based on the quickly evolving information technologies. In this paper, the developed collaborative systems, methodologies and technologies, which are organized as a horizontal or a hierarchical manner, are reviewed. Meanwhile, a 3D streaming technology, which can effectively transmit visualization information across networks for Web applications, is highlighted and the algorithms behind it are disclosed.

Keywords: Collaborative CAD, distribute system, 3D streaming

1. INTRODUCTION

During the past two decades, the mechanical computer-aided design (CAD) industry has experienced some major technological innovations and paradigm shifts. The most recent R&D cycle starting from the end of last century is to renovate CAD systems to be distributed and collaborative by using the quickly developed information technology (IT). It aims to meet the increasing demands of globally collaborative design and the outsourcing trends in manufacturing. In a collaborative CAD system, designers and engineers can share their work with globally distributed colleagues via the Internet/intranet. Furthermore, these collaborative systems also allow designers to work closely with suppliers, manufacturing partners, and customers across enterprises' firewalls to get valuable input into the design chain. With a broader vision, the collaborative CAD systems, computer-aided engineering (CAE), computer-aided manufacturing (CAM), enterprise resource planning (ERP) and product data management (PDM) systems can be integrated to form collaborative product commerce, which supports intra- and inter- enterprise applications for the whole product life-cycle.

A collaborative CAD system cannot be simply set-up through equipping a standalone CAD system with IT and communication facilities. Due to the complexity of collaborative design activities and the specific characteristics/requirements of CAD systems, it needs some innovations or even fundamental changes in many aspects of CAD systems, such as infrastructure design, communication algorithms, geometric computing algorithms, etc.

A collaborative CAD system needs two kinds of capabilities and facilities: distribution and collaboration. These two terms emphasize the different aspects of a system: physically, the former separates CAD systems as geographically dispersed and expands them to support remote design activities, and, functionally, the latter associates and co-ordinates individual systems to fulfil a global design target and objective. In the aspects of enabling technologies, distribution is more fuelled by the development of IT, such as Java, .Net, Web, XML and Web service technologies, and collaboration is more driven by the design and development of effective collaboration mechanisms to facilitate human-human/human-computer relationships. However, although having different focuses, they are closely inter-related and complementary. A collaboration mechanism needs the specific design of the distributed architecture of a system to meet the functional and performance requirement.

Recently, research has been actively conducted to develop prototype systems and methodologies to support collaborative CAD. At the same time, CAD vendors have realized the huge business opportunities in this area and launched some systems in markets in recent several years. According to the functions and roles of users participating in a design activity, a collaboration CAD can be organized as either a *horizontal* or a *hierarchical* manner. The horizontal collaboration emphasizes on collocating a design team from the same discipline to carry out a complex design task simultaneously. The hierarchical collaboration can establish an effective communication channel between upstream design and downstream manufacturing, and it can enrich principles and methodologies of concurrent engineering to link diversified engineering tools dynamically. In this paper, the research works and developed systems are surveyed according as these two collaboration manners, whilst in each category, their distributed architectures and mechanism are discussed.

2. HORIZONTAL COLLABORATIVE CAD

The horizontal collaborative CAD systems can be further divided into two types: visualization-based design systems, which provide a light-weight manner for users to assist collaborative design through visualizing, annotating and inspecting design models in a Web or a CAD environment, and simultaneous collaborative design systems, which support interactive and simultaneous co-modelling and co-modification functions in a teamwork.

2.1 Visualization-based design systems

Visualization-based CAD systems have used to support visualization, annotation and inspection of design models to provide assistance of collaborative design activities. The systems are either generically plugged in a Web browser or add-on viewers in some CAD systems. Some systems and their functions are listed in Tab. 1. Generally, these visualization systems are light-weight, easy-deployed and platform-independent, and they can facilitate an on-line team to take on design discussion, product review, design remark and customer survey to enhance collaborative new products and conceptual design. In order to suit the requirements of these systems in the Internet with limited bandwidth capability, research has been carried out to innovate light-weight 3D standards and 3D streaming communication (the 3D streaming communication will be discussed in Section 4). In order to deliver and manipulate interactive 3D objects effectively in the Web, some concise formats, such as VRML, X3D, W3D and MPEG-4, have been launched to represent the geometry of 3D CAD models as visualization-used triangular meshes and trimming lines [Roy and Kodkani, 1999; Tay and Roy, 2003]. VRML is fundamental for these standards to represent geometric elements and scenes, whilst X3D and MPEG-4 are extended to support XML-based representation and video/audio application in compressed binary formats, respectively. Some formats such as OpenHSF and ZGL are equivalent to the VRML standard in function whilst they define data for effective 3D streaming transmission through defining functions in data compression, mesh simplification and object prioritizing. The above formats are for generic usage and they are not suitable for representing complex CAD models since they lack feature and assembly structures to organize information. The research trend in this area is to support and provide complex engineering data and the attributes, advanced streaming and compression formats, strong interoperability and cross-platform capabilities.

Java Applet and MS ActiveX technologies are widely used for developing the Web-based visualization clients, and some services written in Java Servlet, MS COM/DCOM or CGI technologies are deployed in the server side to provide support and system maintenance [Lee, et al. 1999; Shyamsundar and Gadh, 2002; Chen, et al. 2003]. The Applet or ActiveX clients can communicate with the Servlet or .Net servers. Generally, both of the client side and server side programming capabilities and functions are essential and complementary for this application, since the former such as Applet can provide efficient handling of frequent interactive operations on models by users in clients and the latter such as Servlet can maintain the server effectively and establish the effective communication between the clients and the server.

2.2 Simultaneous collaborative design systems

Simultaneous collaborative CAD systems can effectively support co-modelling and co-modification functions among designers. Effective team organization, coordination and negotiation can ensure the success of a collaboration process, and it is important to propose an effective architecture based on the available IT infrastructures, such as client-server, peer-to-peer and Web service. Meanwhile, referring to the new distributed environment and requirement, it is imperative to design optimized feature and assembly representation schemes.

Distributed system architectures

The architectures for the developed collaborative CAD systems can be classified into three types:

- *"Thin server + strong client"*
- *"Strong server + thin client"*
- *"Peer-to-peer"*

In the first architecture, clients are equipped with whole CAD functions and some communication facilitators. A server plays as an information exchange to broadcast

CAD files or commands generated by a client to other clients during a collaborative design process. Some developed systems in this architecture include CollaCAD™, IX Design™, Tay and Roy [2003], etc. In the second architecture, the data structures in clients are light-weight and they primarily support visualization and manipulation functions (such as selection, transformation, changing visualization properties of displayed parts, etc.). The main modeling activities are carried out in a common workspace in the server side. A thin/strong representation in client/server respectively has been proposed to enhance the performance of the system effectively [Li, et al. 2004]. The developed systems include Alibre Design™, OneSpace™, van den Berg, et al. [2002], Li, et al. [2004], ect. The third architecture, including Begole, et al. [1997] and Inventor collaborative tool™, supports the sharing and manipulation of services or modules of a system by other systems. For example, for the Inventor collaborative tool™, an MS Netmeeting tool is embedded for the peer-to-peer communication and application sharing.

Considering the characteristics of CAD systems, the three architectures show potentials in different aspects. The implementation of the first architecture is the most straightforward comparing to the other two architectures. Through equipped with a communication facility, standalone CAD systems can be conveniently re-developed as design clients and linked together by the server with information exchange and collaboration coordination functionalities. This architecture can effectively meet the requirement of CAD design for real-time interactive operations since most of the geometric computing for modeling and modification is carried out in the clients locally. Meanwhile, it can support heterogeneous modelling systems in clients and a neutral information exchange format, for example, XML, can be designed for communication in the environment [Bianconi and Conti, 2003]. However, the adaptability of the architecture is not easily maintained. If a new user is added in the environment, a whole package of CAD system has to be added and configured. Meanwhile, such architecture is difficult to be migrated to the Web application. The second architecture is getting more popular since it brings a new kind of business model – application service provider (ASP). With such architecture, small and medium enterprises (SMEs) or even individual designers with specific domain knowledge can rent on-line high-end CAD systems, so they are able to participate and co-operate in the design process with large firms. The scalability of system can be enhanced since it is convenient to add new seats in the distributed system. The third architecture employed the peer-to-peer computing manner. Services of a peer with CAD function can be manipulated by another peer. The architecture is high-performed for point-to-point communication and collaboration. However, it is not suitable for a group of users to work together. Compared to the first architecture, the implementation difficulty of the last two is increased, whilst the scalability is enhanced.

Team management and design coordination

In Tab. 2, four simultaneous collaborative design systems are summarized. In them, team management and design coordination functions are the crucial components for establishing a well-organized team to conduct a collaborative design task.

A working session mechanism is effective for the team management. Each session can be used to organize a collaborative task, and designers in the same session can share the design information dynamically [Alibre Design™, Li, et al., 2002]. In a system, different design tasks can be carried out at the same time in different sessions. In a session, designers can play as different roles such as project leader, members and supporters. A project leader is responsible to manage a session and supervise the whole design process, and he/she is authorized to schedule the process and avoid deadlocks during design due to network problems. A design member can carry out the design collaboratively, and a supporter can provide comments for the design or resources required arising during the collaborative design. Messaging is an important assistance functions for collaborative design [Alibre Design™, OneSpace™, CollabCAD™, Tay and Roy, 2003]. Through messages in text, video or audio, designers in a session can communicate with each other to exchange design idea.

Several mechanisms are used for design coordination. A control token mechanism has been utilized in [Alibre Design™, Li, et al., 2002] to schedule a collaborative design activity. Each session has a control token, that is, at any one time, only the user who holds the control token is the active designer and can edit a part; whilst the other users in the same session only receive the updated information and are observers. The user who is carrying out the edition function can become an observer by transferring his control token to another user. The advantage of the control token mechanism is that design conflicts can be avoided during a simultaneous process, and the disadvantage is the low efficiency. Another mechanism is based on an agent system. Mori and Cutkosky [1998] proposed an agent-based system to coordinate design based on the theory of Pareto optimality. The agents are reactive and they can track and respond to changes in the state of the design when any designer changes his model and brings the conflicts. However, the communicating between designers through the agent system is simple and limited, and this architecture is not suitable some complex design activities. In a situation when designers are working for different sub-assemblies, Shyamsundar and Gadh [2002] and Chen, et al. [2004] defined a set of new assembly representations to constrain the design assemblies assigned to individual designer, further to form a whole collaboratively developed assembly. Meanwhile, methodologies to detect and manage conflicts arising from a collaborative activity are investigated [Case and Lu, 1996].

Optimized feature and assembly-based representation

A significant problem for the above systems is that communication efficiencies are still quite far from satisfactory when large-size feature- and assembly-based models are designed collaboratively. In order to address this problem, some works have been appeared recently to optimize or simplify geometric entities of distributed design feature- or assembly feature-based models to accelerate the communication. Wu and Sarma [2001] developed an algorithm to incrementally update the B-Rep of a design model based on a cellular representation in a distributed environment. Based on the cells from the segmented B-Rep of a design model, the algorithm can identify and extract those regions that have been modified by a designer, and dynamically transmit and embed the modified regions into a B-Rep at another site. Lee [1999] proposed a network-centric virtual prototyping system in a distributed computing architecture, in which a shape abstracting mechanism was developed to provide a light-weight Abstracted Attributed B-rep (AAB) in clients to represent a feature-based model stored and maintained in a server for concise and transparent communication between the server and the clients over the network. A naming consistency paradigm was established to maintain the interoperability and identification between geometric entities of the server and the clients during a concurrent design process. Li et al. [2002] developed a distributed feature mechanism to filter the varied information of a working part during a co-design activity to avoid unnecessary re-transferring of the complete large-size CAD files each time when any interactive operation is imposed on the model by a client, so as to enhance the effectiveness of the information communication for co-design activities. In order to support collaborative assembly design activities effectively, Shyamsundar and Gadh [2002] developed a new geometric representation named as AREP and a collaborative prototyping system based on the representation to perform real-time geometric modification for components/sub-assemblies in an assembly model. In AREP, an envelope mechanism was designed to simplify the some internal geometric structures and entities, which are irrelevant to assembly constraints, of components designed separately and collaborated around the assembly constraints. Points are kept in envelopes to refer to corresponding detailed entities for further query and retrieve. Chen et al. [2004] proposed an assembly representation for collaborative design. Their functional modules include a master assembly model (MAM) and a slave assembly model (SAM). The MAM is a complete representation stored in the server, and SAM is a simplified version of MAM used for visualization-based manipulation in the client. However, it does not address the real-time design modification in a collaborative design environment. The research direction is towards supporting optimized traffic and real-time feature and assembly design.

3. HIERARCHICAL COLLABORATIVE CAD

The hierarchical collaboration uses an asynchronous way to organize a design activity. Different from the traditional “sequential engineering”, which is a “throwing-over-wall” approach, this collaboration is still focuses on bidirectional communications and interactions among designers, and it can avoid conflicts happened in a simultaneously collaborative design activity and unnecessary waiting when the design is organized across different time zones. Storing, accessing and maintaining CAD models in repositories securely and conveniently is crucial to support this kind of collaboration function.

Multi-representation for features

In order to support feature-based applications in a collaborative design environment, Gadh and Sonthi [1998] developed a four-level representation scheme for features to address different applications effectively. The representation consists of boundary representation, aggregate geometric abstraction representation, domain independent geometric abstraction representation and domain dependent features. The motivation of this representation is to provide several layers of geometric abstractions and aggregations in a server to response to different manufacturing applications efficiently. Han and Requicha [1998] and De Martino et al. [1998] separately developed a distributed system consisting of a design-by-feature client and a downstream manufacturing feature recognition client connected by a geometric server. The functions of the geometric server are twofold: first, it is a repository to store features generated by these two clients; second, it transfers design features in the design-by-feature client to the feature recognition client. The distinction of these two works is in their feature recognition algorithms. The former used a hint-based reasoning method depending upon the augmented design features as hints, whilst the latter developed a graph-based reasoning method to work on the geometric models converted from the design feature models. In the above works, changes made in the design-by-feature client can be propagated to the feature recognition client automatically to achieve data completeness and consistency. However, this information flow is unidirectional. If a modification of a design part is required by the manufacturing feature recognition client, it should be made in the design-by-feature client, which process forces a user to think in a way that is not natural for him or her and blurs the functional differences among design and manufacturing. Hoffmann and Joan-Arinyo [2000] proposed a master model scenario to store shared design information and a multi-way communication mechanism among design and manufacturing clients. However, the features supported in this work are still limited to some simple types and the work is still far from practical applications. This problem can be effectively solved through developing a generic and robust integration strategy of design-by-feature and feature recognition algorithms to support multiple views of a design model, which is actively investigated [De Kraker, et al., 1997; Jha and Gurumoorthy, 2000].

Integration mechanisms for distributed systems

Several significant infrastructures have been reported. Liu [2000] proposed a COM interface-based framework to wrap and expose API functions of CAD kernels/systems and process planning modules for remote invocations. The concept of developing standard interface specifications, namely the common core interfaces, was proposed to encapsulate specific feature functions of different CAD kernels/systems to provide a generic and neutral application layer according to some international standards for features. The advantages of these two works include the straight-forwardness of calling wrapped feature functions and the neutrality of CAD kernels/systems for different applications. However, considering the complexity and variation of features, the programming effort for implementation is quite huge and the add-on wrapping structures make the system quite heavy. In Jacquel and Salmon's system [2000], features from an ACIS modelling kernel are wrapped as services for remote design and manufacturing analysis. Gerhard et al. [2001] proposed an event-based and agential framework to communicate design and manufacturing information through agent channels based on the Java RMI technology, and manufacturing analysis functions are enveloped as agents to support the establishment of an open and plug-in environment. Compared to the discussed former infrastructure, the agent-based mechanism can provide a more flexible and lighter-weight working manner for communication and collaboration. Some developed systems are surveyed in Tab. 3.

4. 3D STREAMING TECHNOLOGY

A streaming technology was initially developed and applied in the video and audio industries to provide an effective way for transferring large-volume images and sounds over the Internet. Through an incremental process, huge video and audio files downloaded from a server can reach clients for gradual display. Similarly, in order to effectively transmit 3D CAD information across networks with the limited bandwidth and to enhance the visualization performance, a 3D streaming technology for CAD visualization information, typically triangular meshes, have recently emerged to dispatch the meshes from a simplified level to a complex level gradually. Different from video and audio which data structures are natural to be re-arranged as frames for streaming, meshes for CAD models are difficult to be de-coupled as continuous streams and a set of geometric algorithms need to be developed.

Two algorithms are crucial for establishing the 3D streaming technology: mesh simplification algorithm and mesh refinement algorithm. The mesh simplification algorithm can reduce the original mesh model to a smaller-size model and shorten clients' waiting time to obtain the first sight of the model across the network with limited bandwidth capability. The mesh refinement algorithm, which is a reverse process of the mesh simplification, provides clients with a gradually refined model through a smooth transition from the coarser model to the original one. Hence, 3D streaming is actually the incremental refinement process through progressive transmission over the Internet.

Mesh simplification

Mesh simplification algorithms are used to decouple a mesh-based CAD visualization model as a simplified model whilst keeping some shape features to achieve acceptable approximations to the original shape. The developed simplification algorithms can be generally categorized into three classes:

- *Vertex decimation*
- *Iterative edge contraction*
- *Vertex clustering*

Vertex decimation, which was initially proposed by Schroeder et al. [1992], is used to reduce numbers of meshes through removing vertices and re-triangulating of the meshes. Many improved algorithms have been developed following Schroeder's decimation idea [Soucy and Laurendeau, 1996; Garthwaite and Reposa, 2000]. However, these approaches cannot handle some associated attributes such as color and texture when performing decimation and re-triangulation.

Iterative edge collapse is another promising simplifying approach that preserves volume and other geometric properties better than vertex decimation.

Some of the common steps used by iterative edge collapse consist of selection of vertex pairs (either edge or non-edge type), determination of target point placement and reconstruction. In Fig. 3, it can be observed that an edge collapse can reduce two triangles (shaded) and one vertex (v_1, $v_2 \to \bar{v}$). In [Hoppe, 1996; Melax, 1998], some edge collapse algorithms have been developed. Hoppe [1996] developed a level-of-details (LOD) structure based on the edge collapse method. By simplifying the initial mesh representation of $\widehat{M}$ into $\{M_n = \widehat{M}, M_{n-1}, \ldots, M_1, M_0\}$ along the coarser direction, this LOD structure is able to render smooth visual transition along the sequence of $M_0 \to M_1 \to \ldots \to M_{n-1} \to M_n = \widehat{M}$, which shows different levels of detail of the original mesh model at different time.

Vertex clustering was first introduced by Rossignac and Borrel [1993] to process arbitrary polygonal input of mesh representation. Regardless of the original shape, it places a bounding box around the original model and divides it into a grid. All the vertices inside a certain cell will be treated as one vertex so that the original mesh model will be simplified cell by cell. This process can be implemented very fast but dramatically alters the model's topology. Moreover, the quality of simplification is hard to control since it depends on the size and number of grid cells, which cannot ensure a good approximation with topological loyalty.

Mesh refinement

In the 3D streaming algorithm, the simplification process is followed immediately by mesh refinement. There are two refinement methods available: the progressive forest split (PFS) method to refine the mesh model from low-detail to high-resolution, and the multi-resolution mesh method to generate a series of LODs of the original model. In the PFS proposed by Taubin et al. [1998], features an adaptive refinement operation, which arbitrarily adds more vertices and connections within the forest of edges to incrementally make the mesh smoother. The shortcoming of this method is that it not support the path from the simplified version to its original, which means that it is a smoothing process based on the current level of mesh regardless of whether this refinement can or can not restore the mesh to its original form.

The multi-resolution mesh method, initially proposed by Funkhouser and Sequin [1993] and improved by Hoppe [1996], can preserver design models and facilitate 3D streaming effectively through transmitting the series of LODs. The shortcoming of this method is that too many data are needed during streaming. The research trend is to combine the PFS and the multi-resolution methods to achieve good performance in terms of the accuracy of shape preservation and optimized traffic.

5. FUTURE TRENDS AND CHALLENGES

(1) _Integration of horizontal and hierarchical collaboration_

Horizontal and hierarchical collaborations are complementary in functions. It is important to establish a vertically seamless linkage between the upstream design and the downstream manufacturing processes through the creation of intelligent strategies for effective information interchange, and the horizontally interpersonal linkage of group work in the upstream design phases. The integral system can support interrelated activities and share domain knowledge between designers and systems to improve design quality and efficiency. Modules for the hierarchical collaboration should be wrapped as services for remote revoking. Within the integral system, scheduling and coordination is becoming crucial and challenging, and distributed intelligent algorithms and technologies such as agent-based systems can be used to enhance the integrated system.

(2) _New feature-based formats and enhanced streaming technology for Web applications_

The primary geometric data in a VRML model are triangular patches and boundary trimming lines between faces, and the information for the high-level features can not be preserved. In order to organize the visualization data as a feature-based format to support some feature-based manipulations in the Web-based visualization module, such as highlighting or hiding a feature in a part, dynamically retrieving some important parameters and attributes of a feature, or evaluating the creation history of the part, a new visualization format based on features and VRML needs to be developed.

For the 3D streaming technology, such as Hoops streaming toolkit™ and the RealityWave vizStream™, not all components or details of the entire component might be required for viewing at each instant, and different users might have interests in different portions of a model. The 3D streaming technology should be enhanced to provide the selective visualization function and adaptive multi-resolution representation according to users' definitions and requirements.

(3) _Security and Interoperability of collaborative CAD systems_

As customers and suppliers move to Internet-based collaboration, security must be considered carefully. Whilst much of the security solutions offered by the current collaborative CAD systems will be handled at the transmission level, they can also benefit by incorporating additional security features into their data models.

Interoperability between collaborative CAD and PDM systems must be achieved. IGES is currently the de facto standard for SMEs and suppliers. So at a minimum, collaborative CAD solutions must be able to successfully handle IGES importing and exporting between the major CAD application. Ultimately, the goal for collaborative CAD solutions must include the ability to access and manipulates legacy CAD data in its native file format.

6. REFERENCES

Begole J, Struble CA and Shaffer CA, 1997, Leveraging Java applets: toward collaboration transparency in Java. IEEE Internet Computing, 1-2, pp.57-64.

Case MP and Lu SCY, Discourse model for collaborative design. Computer-Aided Design, 28, pp.333-345.

Chan FTS, Zhang J and Li P, 2003, Agent- and CORBA-based application integration platform for an agile manufacturing environment. International Journal of Advanced Manufacturing Technology, 21(6), pp.460-468.

Chen L, Song ZJ and Feng L, 2004, Internet-based real-time collaborative assembly modeling via an e-assembly system: status and promise. Computer-Aided Design (in press).

De Kraker KJ, Dohmen M and Bronsvoort WF, 1997, Maintaining multiple views in feature modelling. Proc. Solid Modelling'95, New York, USA, pp.123-130.

De Martino T, Falcidieno B and Habinger S, 1998, Design and engineering process integration through a multiple view intermediate modeller in a distributed object-oriented system environment. Computer-Aided Design, 30, pp.437-452.

Funkhouser T and Sequin C, 1993, Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments. Computer Graphics (Proc. ACM SIGGRAPH'93), 27, pp.247-254.

Gadh R and Sonthi R, 1998, Geometric shape abstractions for internet-based virtual prototyping. Computer-Aided Design, 30, pp.473-486.

Garthwaite T and Reposa J, 2000. Mesh Decimation. Major Qualifying Project Report, MOW-2733, Worcester Polytechnic Institute.

Gerhard JF, Rosen D, Allen JK and Mistree F, 2001, A distributed product realization environment for design and manufacturing. ASME Journal of Computing and Information Science in Engineering, 1(3), pp.235-244.

Han J and Requicha AAG, 1998, Modeller independent feature recognition in a distributed environment. Computer-Aided Design, 30, pp.453-463.

Hoffmann CM and Joan-Arinyo R, 2000, Distributed maintenance of multiple product views. Computer-Aided Design, 32, pp.421-431.

Hoppe H, 1996, Progressive meshes. Computer Graphics (Proc. ACM SIGGRAPH'96), 30.

Huang GQ, 2002, Web-based support for collaborative product design review. Computers in Industry, 48, pp.71-88.

Jacquel D and Salmon J, 2000, Design for manufacturability: a feature-based agent-driven approach. Proc. of the Institution of Mechanical Engineers Part B, 214(10), pp.865-880.

Jha K and Gurumoorthy B, 2000, Multiple feature interpretations across domains. Computers in Industry, 42, pp.421-431.

Lee JY, Kim H, Han SB and Park SB, 1999, Network-centric feature-based modelling. Proc. Pacific Graphics'99, South Korea, pp.280-289.

Li WD, Ong SK, Fuh JYH, Wong YS, Lu YQ and Nee AYC, 2004, Feature-based design in a collaborative and distributed environment. Computer-Aided Design (in press).

Liu XD, 2000, CFACA: component framework for feature-based design and process planning. Computer-Aided Design, 32, pp.397-408.

Melax S, 1998, A simple, fast and effective polygon reduction algorithm. Game Developer magazine, pp.44-49, (http://www.melax.com/polychop/gdmag.pdf).

Mori T and Cutkosky MR, 1998, Agent-based collaborative design of parts in assembly. Proc. 1998 ASME Design Engineering Technical Conferences, Atlanta, Georgia, USA, DETC98/CIE-5697.

Rossignac J and Borrel P, 1993, Multi-resolution 3D approximation for rendering complex scenes. Modeling in Computer Graphics: Methods and Applications, pp.455-465.

Roy U and Kodkani SS, 1999, Product modeling within the framework of the World Wide Web. IIE Transactions. 31, 667-677.

Schroeder WJ, Zarge JA, Lorensen, E, 1992 Decimation of triangle meshes. Computer Graphics (Proc. SIGGRAPH '92), 26, pp.65-70.

Shen WM, Maturana F and Norrie DH, 2000, MetaMorph II: an agent-based architecture for distributed intelligent design and manufacturing. Journal of Intelligent Manufacturing, 11, pp.237-251.

Shyamsundar N and Gadh R, 2002, Collaborative virtual prototyping of product assemblies over the Internet. Computer-Aided Design, 34, pp.755-768.

Soucy M and Laurendeau D, 1996, Multiresolution surface modeling based on hierarchical triangulation. Computer Vision and Image Understanding. 63(1), pp.1-14.

Sun J, Zhang YF and Nee AYC, 2001, A distributed multi-agent environment for product design and manufacturing planning. International Journal of Production Research, 39(4), pp.625-645.

Taubin G, Gueziec A, Horn W and Lazarus F, 1998, Progressive forest split compression. Computer Graphics (Proc. SIGGRAPH'98), 32, pp.123-132.

Tay FEH and Roy A, 2003, CyberCAD: a collaborative approach in 3D-CAD technology in a multimedia-supported environment. Computers in Industry, 52, 127-145.

van den Berg E, Bidarra R and Bronsvoort WF, 2000, Web-based interaction on feature models. Proc. Seventh IFIP WG 5.2 Workshop on Geometric Modelling: Fundamentals and Applications, Parma, Italy, pp.319-320.

Wu D and Sarma R, 2001, Dynamic segmentation and incremental editing of boundary representations in a collaborative design environment. Proc. 6th ACM Solid Modeling'01, Michigan, USA, pp.289-300.

Zhou SQ, Chin KS, Xie YB and Yarlagadda P, 2003, Internet-based distributive knowledge integrated system for product design. Computers in Industry, 50, pp.195-205.

Products		Characteristics and functions	Data distributed methods
Cimmetry Systems Autovue™	(1) (2)	A viewer for part and assembly models. View, mark-up, measure, explode, cross-section, etc.	3D streaming
InFlow ConceptWorks™	(1) (2)	An add-on viewer to SolidWorks. (2) View and mark-up.	3D streaming
Actify SpinFire™	(1) (2)	A viewer for part models. View, cross-section, measure, grid and ruler.	Download
SolidWorks eDrawing™	(1) (2)	A viewer for native or simplified SolidWorks files. View, mark-up, measure, 3D pointer, animation.	Download
Adaptive Media Envision3D™	(1) (2)	A viewer for part models. View, mark-up, redline, chat.	3D streaming
Centric Software Pivotal Studio™	(1) (2)	A platform to provide a manager, a project organizer and a viewer. View, mark-up, video/audio conferencing, chat.	Download/ 3D streaming
Hoops Streaming Toolkit™	(1) (2)	A toolkit to provide 3D streaming APIs. BaseStream class library, advanced compression, attribute support, etc.	3D steaming
RealityWave ConceptStation™	(1) (2)	A VizStream platform, which consists a server and a client. View, mark-up, message.	3D streaming
Autodesk Streamline™	(1) (2)	A platform based on the VizStream. View, measure, bill of materials.	3D streaming

Tab. 1. Visualization-based CAD systems.

Systems	Collaborative mechanisms	Functions & information distribution
Alibre Design	*Team design sessions* A design session can be used to organize a virtual team to design 2D and 3D models simultaneously. *Repository* Through repository, users' models can be securely shared and accessed. *Message Centre* It can support message sharing among users.	• 2D and 3D Modelling • Mark-up • Annotation • View • Text and voice chat • Directly transferring CAD models • Peer-to-peer communication
CoCreate OneSpace	*3D personal collaboration* Through this service, up to two other users can be invited to an online meeting. Meeting users can view and mark-up 2D or 3D models. *Model manager* It can store and share users' models through a database, and specify who has permission to read and modify design work. *Project data manager* It can organize 2D or 3D project files in a database and helps track of document version and history.	• 2D and 3D Modelling • Mark-up • View • Netmeeting • Integration with PDM • Directly transferring CAD models • Client/Server communication
Autodesk Inventor Collabor-ative Tool	*Application sharing* MS netmeeting tool is embedded into Inventor systems to organise co-design activities. An Inventor that has the "control baton" can control and manipulate another remote Inventor system to design, and the controlled system is an observer. The "control baton" can be acquired and exchanged.	• 2D and 3D Modelling • Netmeeting, • Whiteboard • Chat • Directly transferr-ing CAD models • Netmeeting T.120 communication
Collab-CAD	*Design team* Members in the team can simultaneously design and share 2D and 3D models. *Repository* It can store and share users' models through a database.	• 3D Modelling • Text or voice chat • Directly transferring CAD models • Client/Server communication

Tab. 2. Simultaneous collaborative CAD systems.

R&D work	Key characteristics	Communications and infrastructures
Liu [2000]	A generic component framework for distributed feature-based design and process planning.	• Exchange data: STEP files • Client/server based on MS COM/DCOM
Jacquel and Salmon [2000]	A manufacturing analysis agent system supporting feature-based design and manufacturability evaluation.	• Autonomous agent organisation (based on Swarm multi-agent system) • ACIS 3D kernel to support design feature agents
Shen, et al., [2000]	A MetaMorph agent architecture ensuring the coordination among design parts and resource agents to support distributed design and manufacturing activities.	• Mediator-centric hybrid agent organisation • AutoCAD with AME 2.0 to support product design • TCP/IP protocol to support high-level KQML communication among agents
Sung, et al., [2001]	A CyberCut system integrating product design and process planning, including several modules: (1) a Web-based design tool; (2) a new geometric representation for information exchange between the design and process planning modules, and (3) an automated process planning system.	• Exchange data: a designed geometric representation called SIF-DSG • Web client/server
Sun, et al., [2001]	An agent architecture integrating design, manufacturability analysis, process planning and scheduling.	• Exchange data: proprietary CAD files • Multi-agent organisation (based on JATLite multi-agent system) • TCP/IP protocol to support high-level KQML communication among agents
Cheng, et al., [2001]	A Web-based design and manufacturing support system, including seven functional modules: electronic catalogue, intelligent selection, mounting details, sealing devices, lubrication, manufacturing database, and design module.	• Exchange data: CAD drawing files • Three-tier client/server based on Java RMI and JDBC
Huang [2002]	A Web-based system to support designers and management to make product design review collaboratively.	• Exchange data: Design documents • Web client/server
Kong, et al. [2002]	An Internet-base collaborative system for a press-die design process for automobile manufacturers.	• Exchange data: CAD files • Three-tier client/server based on CORBA, Java, Java3D and relational database
Chan, et al., [2003]	An integrated system to support an agile manufacturing, in which an agent technology is used to wrap intelligent manufacturing functional modules and CORBA is for the establishment of the communication infrastructure.	• Multi-agent organisation (based on JATLite multi-agent system) • TCP/IP protocol to support high-level KQML communication among agents
Zhou, et al., [2003]	An Internet-based system for designers to look for and retrieve distributive design knowledge, which is represented according to STEP standards and an ANN is used for knowledge search engine.	• Exchange data: STEP files • Three-tier client/server based on ASP and OCBC

Tab. 3. Related work of distributed concurrent engineering for design and manufacturing.