ASSEMBLY OPERATION TOOLS FOR *e*-PRODUCT DESIGN AND REALIZATION

by

Kyoung-Yun Kim

Bachelor of Science, in Industrial Engineering, Chonbuk National University, South Korea, 1995

Master of Science, in Industrial Engineering, Chonbuk National University, South Korea, 1998

Submitted to the Graduate Faculty of

the School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2003

UNIVERSITY OF PITTSBURGH

SCHOOL OF ENGINEERING

This dissertation was presented

by

Kyoung-Yun Kim

It was defended on

June 24th, 2003

and approved by

Dr. Bopaya Bidanda, Professor, Department of Industrial Engineering

Dr. Michael R. Lovell, Associate Professor, Department of Mechanical Engineering

Dr. Bryan A. Norman, Associate Professor, Department of Industrial Engineering

Dr. Anne M. Robertson, Associate Professor, Department of Mechanical Engineering

Dissertation Director: Dr. Bartholomew O. Nnaji, Professor, Department of Industrial Engineering

ABSTRACT

ASSEMBLY OPERATION TOOLS FOR *e*-PRODUCT DESIGN AND REALIZATION

Kyoung-Yun Kim, PhD

University of Pittsburgh, 2003

True competitive advantage can only result from the ability to bring highly customized quality products to the market at lower cost and in less time. Many customers are demanding customization and rapid delivery of innovative products. Industries now realize that the best way to reduce life cycle costs is to evolve a more effective product development paradigm using the Internet and web-based technologies. Yet there remains a gap between these market demands and current product development paradigms.

Assembly plays a very important role in manufacturing industries, given that joints on a structure are inevitable because of the limitations on component geometric configurations and material properties along with various engineering requirements. Appropriate joints should be determined by considering mechanical and mathematical implications and assembly/joining knowledge. Currently, the effects of joining are analyzed upon completion of assembly modeling. This sequential process is arduous and time-consuming and is eliminated with the tools developed in this work. The existing CAD systems require that a product developer possess all the design and analysis tools in-house making it impractical to employ all the needed and newest tools. Existing assembly design methodologies have limitations on capturing the non-geometric aspects of a designer's intent and the physical effects of joining in an Internet-based product development environment.

In this work, new assembly design (AsD) frameworks and assembly operation tools (AOT) are developed to integrate AsD, *virtual analysis*, and decision making for *e*-product design and realization. The AOT include the *assembly design (AsD)*, *assembly implication (AsI)*, and *assembly advisory (AsA) engines*. The AsD formalism, which is the base of the AsD engine, represents the assembly/joining relations symbolically for computer interpretation, and the automatically generated AsD model is used for inferring mathematical/physical implications, as well as lean AsD information exchange. A new *virtual assembly analysis* concept is introduced to transparently predict the various effects of joining and is implemented in a *service-oriented architecture*. The AsA engine employs *hierarchical semantic net* to support an AsD decision by capturing AsD information and assembly/manufacturing knowledge. The concepts and AOT are validated using a case study of realistic mechanical assemblies.

DESCRIPTORS

AHP
Assembly Design Decision Making
Assembly Operation Tools
CAE
Concurrent Engineering
e-Design and Realization
Hierarchical Semantic Net
Joint Design and Analysis
Life-Cycle Engineering
Service-Oriented Architecture
Assembly Design
Assembly Operation Analysis
CAD/CAM
Collaborative Assembly Design
Design Formalism
e-Tools
Joining Process
Knowledge-Based Design Decision Making
Semantic Network
Virtual Assembly Analysis

ACKNOWLEDGEMENTS

In acknowledging my appreciation and thanks, I am most grateful to the Almighty God for His divine guidance and blessings, with which His has brought many wonderful people during the course of my research.

I received a lot of support from many people during the course of this research. I owe special thanks to my academic advisor, Prof. Bart. O. Nnaji, for his advice, inspiration, and support in this work. My gratitude also goes to the entire faculty and staff of the Department of Industrial Engineering at the University of Pittsburgh, for the support and kindness they have shown me over the years. I thank Drs. Bopaya Bidanda, Michael Lovell, Bryan Norman, and Anne Robertson, for serving on my dissertation committee, and for providing invaluable advice and constructive criticism for my work. Special appreciation is extended to Mr. David L. Conover, Corporate Fellow of ANSYS, Inc. for supporting this work with software and training, and for providing the test-site for tools that were developed. I also would like to thank Prof. Dong-Won Kim at Chonbuk National University, South Korea, for his encouragement and intellectual advise on my graduate work.

I appreciate the efforts of all the team members of Pegasus *e*-Designer project, whose constructive criticism and discussions have lent a great impetus to the success of this project. My heartfelt thanks go to all of the Automation and Robotics laboratory members for their support and suggestions in this work. I acknowledge the help of Yan Wang and David Manley in providing useful suggestions and assistance with the formatting of this report and in proofreading parts of this dissertation.

Finally, I wish to thank to my parents who have always provided endless love throughout my life with their continuous prayer. My deepest appreciation has been reserved for my wife Mi-Jeong for all her love, unwavering support, and continued prayers. Last but not the least, I would like to thank the little ones, Hanna and our expecting baby. Your smile has always inspired me, Hanna. Thank you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

The mechanical product industry, as one of the leading industry sectors in the United States economy, generates over $1 trillion in annual revenue (US Department of Commerce 2002). Two areas of this sector, the automotive and aerospace industries, alone generate over $700 billion in annual revenues. The entire industry sector requires a high level of performance in productivity and quality to maintain a competitive edge in the global economy. Assembly plays a very important role in this sector, especially in the automotive and aerospace industries. Approximately 40 percent of General Motors' manufacturing facilities are designated for assembly (Bandyopadhyay *et al.* 2001). There have been many cases reporting the significant contribution of improvement of assembly and assembly design on overall cost reduction. For example, Broothroyd reported cases suggesting the replacement of assembly operations and assembly design with alternatives was able to reduce total cost by 50 percent on average. As an another illustration, Jame Cnossen, Ford manager of manufacturing systems and operations research reported that savings of over 1 billion dollars as a result of improving assembly design in products to the Taurus line of cars (Boothroyd *et al*. 1994, Molloy *et al*. 1998).

Joints on a structure are inevitable because of the limitations on component geometric configuration and material property and the requirements of inspection, accessibility, repair, and portability (Messler 1993). The problem of joining components is therefore a key issue in the design process. Joining components often provides a way of realizing simpler forms of the individual components of products, which can make it easier and cheaper to manufacture each component. However, joints frequently cause problems of various considerations. First, from a mechanical or chemical viewpoint, many failures of fatigue or of corrosion occur at welded joints. Special treatment or non-destructive testing may have to be employed to prevent the potential problems in consideration of the characteristics of joining methods (LeBacq *et al*. 2002). As another illustration, physical effects of joining sometimes lead to local and global weakening of the mechanical properties of the material of the components (e.g., the heat affected zone of a weld and deformation effects of welding). Second, from an efficiency viewpoint, certain joining methods need some extra material to be added to the structure (i.e., screws, bolts, rivets, or welding filler material). Third, from a overall manufacturing and assembly cost viewpoint, the increase of the number of joints can increase overall manufacturing and assembly cost. The number of joints must be optimized in order to decrease the overall cost while maintaining engineering requirements. Designing the assembly while keeping in mind potential joining problems, is an important aspect of efficient product design. Recent trends toward recycling may lead the designer to consider disassembling as well as assembling components. In this context, the importance of assembly design considering joining has been highlighted in manufacturing industries to a greater extent (Shyamsundar *et al.* 1998, Srinivasan *et al*. 1999).

In order to achieve high performance during a product's life-cycle, an intelligent assembly design system should be able to assist a designer during a product's assembly and joint design processes. This can be realized by predicting expected assembly design problems, providing alternative suggestions, and eventually solving assembly and joining problems. Such an ideal intelligent assembly design system should have the capability of employing *spatial relationships* and joining protocols that result in the physical realization of an assembly. Traditional solid assembly modeling systems, while adequate for visualization purposes, cannot support downstream activities, such as joining analysis, manufacturing analysis, and product design intent analysis (Sriraman 1999).

True competitive advantage can only result from the ability to bring highly customized quality products to the market at lower cost and in less time. Product development has become a very complicated process. Discrete product manufacturers are under pressure from customers (and the market) to move away from the traditional *make-to-stock* production model to a *build-to-demand* model. Many customers are no longer satisfied with mass-produced goods. They are demanding customization and rapid delivery of innovative products (FIPER 2001, ISIGHT 2002). Industries now realize that the best way to reduce life cycle costs is to evolve a more effective product development paradigm using the Internet and web-based technologies. Yet, there remains a gap between these current market demands and current product development paradigms. The existing CAD systems require that a product developer possess all the design analysis tools in-house, making it impractical to employ all the necessary and newest tools.

1.1 Current Assembly Design

In order to achieve high performance in a product's life-cycle, an intelligent assembly design system should be able to assist a designer during the product assembly and joint design process. An ideal intelligent assembly design system should have the capability of employing spatial relationships and joining protocols that result in the physical realization of an assembly. Existing designer systems have limitations on capturing the non-geometric aspects of designer intent on an assembly with joints. The result is that the designer in a CAD environment cannot completely specify joining relationships on an assembly design. Therefore, the development of an assembly formalism to specify joining relationships symbolically is a prerequisite for an intelligent assembly modeling system.

Collaborative assembly design is just starting to emerge as a viable alternative to the traditional assembly design process, in which an assembly design can be developed via an iterative process between designers, manufacturers, marketing people, and ultimately customers in remote locations. This emergence can be linked to the recent outburst of growth in the development of the Internet and associated technologies. There are some research efforts that are investigating the assembly methodologies and protocols necessary for distributed assembly design. However, it is not still fully clear how assembly and joint design should be implemented in collaborative design environments. None of the existing research has developed an assembly formalism that accommodates joining processes. Thus, there is a strong need to develop an assembly design formalism and framework to capture general assembly relationships and joining relationships of an assembly in a collaborative design environment.

Joints on a structure are inevitable because of various engineering requirements and products are very rarely monolithic. The trial and error procedure is generally used in assembly design processes, because current assembly modeling systems have no means of checking various effects of joining during assembly design. The current design practice and analysis for verifying an assembly design concept is usually performed after selecting a final design concept. For example, a welding operation can generate thermal expansion and distortion of a structure, which will affect the joint and finally the entire structure. If a welded structure is distorted, then precision assembly cannot be achieved. Therefore, the weld distortion should be minimized by optimizing the welding operation or by the use of an alternate joining method, such as joining with cast nodes. In another illustration, the rivet joints of an aircraft body frame should be capable of sustaining the prescribed load or mechanical forces in physically holding the assembly components together. If analysis indicates that stress level is not well balanced, the number of rivet joints could be optimized or an alternate joining method, such as welding, could be considered. Presently, the effects of joining, as described in previous examples, are analyzed after finishing assembly modeling. If the analyses indicate that certain modification is required, then another iteration of modeling is needed. This process can be arduous and time-consuming.

Instead of the current trial and error procedure for verifying an assembly design concept, a more efficient process is introduced in this work to predict the various effects of joining. The joining analysis process is embedded into the assembly design process and it can guide designers to make appropriate design decisions. This integrated process generates an assembly design for joining and eliminates the time-consuming feedback processes between assembly design and assembly analysis processes. Previous research has largely focused on assembly modeling and process planning without considering joining processes. This justifies that there is a strong need to develop an assembly modeling system which can provide assembly operation tools as a prelude to generating mechanical and physical implications of a joining process.

1.2 Research Objectives, Tasks, and Approaches

In this research, a set of engineering tools is developed to accomplish the following in product assembly design processes: 1) to describe joining relations in assemblies; 2) to capture the spatial relationship implications and physical effects of joining; 3) to assist a designer to make a right decision on assembly design; and 4) to improve assembly design efficiency by supporting concurrent assembly design and joining analysis. These objectives are realized in this work by the following research tasks.

- **Development of an assembly design formalism and an associated design engine to capture joining relations:** This assembly design formalism allows the specification of the joining relations symbolically, which computer tools can interpret, and it has mathematically solvable implications.
- **Extension of the spatial relationship kernel to embody joining relations:** Various joining processes are specified by using an appropriate protocol developed in this research. The assembly design engine integrates the extended spatial relationship kernels and captures interaction of geometric elements within assembly.
- **Development of an assembly implication engine to capture spatial relationship implications and the physical effects of joining:** The assembly implication engine extracts various implication information from the assembly model (e.g., the spatial relationship implications and physical effects before/after joining). The obtained information provides for an understanding of the feasibility of the specified joining process within the geometric constraints during the assembly design process. A virtual assembly analysis tool is developed

as a subset of the assembly implication engine to predict the physical effect of joining processes in an Internet-based collaborative assembly design environment.

- **Development of an assembly advisory engine to capture joint design information (implication) and to support assembly design decisions:** This assembly advisory engine will propose joining alternatives to a designer by considering the assembly design information, obtained from the assembly design engine and the assembly implication engine, and assembly/joining knowledge.

In this work, a service-oriented architecture for Internet-based collaborative assembly design is developed. This architecture provides a scalable, flexible, and efficient collaborative assembly design platform, which enables different stakeholders of assembly design to work on an assembly product development concurrently. *Service* is envisioned as the core for collaborative assembly design within this platform. Various computational engineering tools make certain services available to other design participants in a network-based distributed environment. Based upon this service architecture, a Virtual Assembly Analysis (VAA) tool is developed to enable transparent and remote joining analysis and its competence for Internet-based collaborative product design environment is discussed. In addition, the VAA processes are validated with physical experiments.

This work introduces an assembly design formalism considering joining relations and implements an Assembly Design (AsD) engine utilizing this assembly design formalism. This formalism provides mathematically solvable implications. The capturing of joining relations to preserve design intent was accomplished by using a spatial relationship kernel developed by Liu and Nnaji (1991). A Spatial Relationship Implication (SRI) tool developed in this research interprets designer's joining intent captured by the assembly design formalism and it provides for an understanding of whether a particular assembly process satisfies the designer's intent. In addition to the VAA tool, this SRI tool serves as an important sub-tool of an Assembly Implication (AsI) Engine.

Finally, a *hierarchical semantic net*-based Assembly Advisory (AsA) engine interprets the captured physical effects and mathematical implications of assembly operations. This engine manages the interaction between nominal geometry and an assembly process. While the AsD engine and the AsI engine result in the realization of design for assemblability, the Assembly Advisory (AsA) engine supports designers' decisions on joining. In this work, a new Assembly Design Decision Making (ADDM) framework is developed to propose assembly alternatives to the designer by considering assembly design information, obtained from the AsD and AsI engines, and assembly/joining knowledge base.

1.3 Research Organization

In this documentation, Chapter 2 provides a background and literature review of relevant research areas and important aspects of this research. Chapter 3 explores how the developed AsD formalism captures assembly and joining relations. The developed AsD formalism is implemented on a relevant AsD engine. Chapter 4 discusses how an Assembly Implication (AsI) engine can capture spatial relationship implications and physical effects of joining in service-oriented collaborative assembly design. Chapter 5 explains how the AsA engine supports assembly design decision-making. In Chapter 6, the developed concepts and frameworks, and the performance of the assembly operation tools (AOT) are tested and validated using a case study. Chapter 7 concludes this dissertation with the contributions and areas of future research.

2.0 BACKGROUND AND LITERATURE REVIEW

2.1 Assembly Operation and Joining

An assembly is a collection of manufactured parts, brought together by assembly operations to perform one or more of several primary functions. An *assembly operation* is defined as the process or series of acts involved in actual realization of assembly. Joining finalizes the assembly operation and generates joints. Messler (1993) divided the primary functions of the assembly into three categories: structural, mechanical, and electrical. The primary function of structural assemblies is to carry static and/or dynamic loads, such as body frames of automobiles. For mechanical assemblies, the primary function, while often seemingly structural, is to create, enable, or permit some desired motion or series of motions through the interaction of the component parts. Examples can be found in automotive engines, robot arms and manipulators, actuators, etc. For others, the primary objective may be to permanently join two or more components, such as an automobile welded space frame. Such assemblies must be capable of carrying loads and, so they must be structurally sound. The loads being carried are another important consideration for the purpose of creating or permitting motion. Finally, the primary purpose of electrical assemblies is to create, transmit, or process some desired electromagnetic signal to perform some function, such as microelectronic packages and printed circuit boards.

Usually, assemblies perform multiple functions, with some function being primary and the others secondary; thus, the joints in an assembly also perform multiple functions. For instance, the solder joints on a printed circuit board have the primary function of providing electrical connectivity, but they also sustain mechanical forces by physically holding the assembly of electrical components together in proper arrangement under acceleration or differential thermal expansion and contraction. The primary function of the joints in an automobile frame is to provide a structural connectivity. These joints may also have a secondary function of allowing certain movement corresponding to vibration of the structure. To achieve a function, diverse material properties and multiple parts are often employed. In the case of an automobile frame, joints must be created between those different components and different materials. The joining of different materials to achieve function is often a challenging aspect of joining: for example, the joining of transparent and brittle glass with a tough structural metal frame.

To enable material and structural optimization, an appropriate joint design is critical and can provide additional benefits in terms of damage tolerance by changing properties along a potential crack path, thus, disrupting and arresting crack propagation. Local joints should be compatible to the overall structure design. If a deformation effect of a weld joint on a metal frame is propagated onto a windshield area, it can result in a fitting distortion problem between the window and the metal frame (Nnaji *et al.* 2003-a). Moon and Na (1997) and Tarng *et al.* (1999) developed mathematical models and applications to optimize welding processes, but they did not present a methodology to connect their process optimization tools and assembly modeling tools. Figure 2.1 shows various joining processes. Appropriate joining processes should be selected considering various constraints, such as material, manufacturing, assembly, and geometric constraints, as well as their physical effects. Through this work, an assembly modeling method has been developed to enable a designer to select a particular joint, satisfying required functions. A set of joining methods including riveting and welding was developed as a case study in this work.

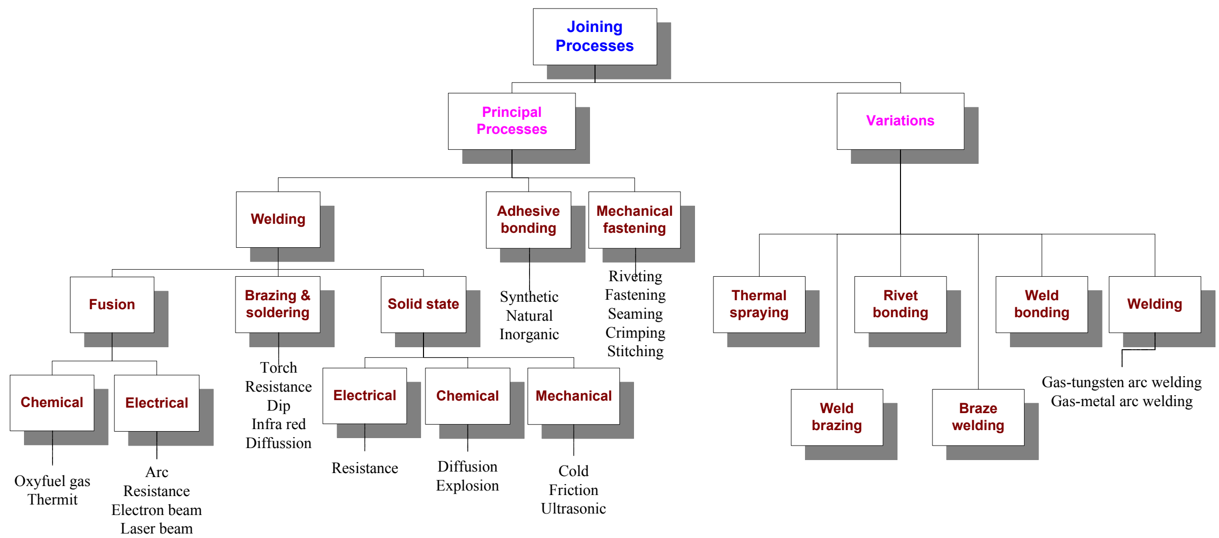


Figure 2-1 Joining processes, adapted from Kalpakjian (1995)

2.1.1 Arc Welding Operation

Arc welding is a joining process by which workpieces are joined with an airtight seal between their surfaces. These welding processes involve partial melting and fusion of the joint between two members, so the thermal energy required for these welding operations is usually supplied by chemical or electrical means. Filler metals, which are metals added to the weld area during welding of the joint, may or may not be used. Fusion welds made without the addition of filler metals are known as autogenous welds. During welding processes using electrical means, there is an ongoing electric discharge generating sparks between the electrode and the workpiece.

The resulting high temperature of more than 3300 °C (6000 °F) melts the metal in the vicinity of the arc and the molten material from the electrode is added to supplement the welding seam. Whereas spot welding is performed with an alternating current, arc welding is performed with direct current, usually at 100 - 200 A at 10 - 30 V (Kalpakjian 1995, Kim 1994).

Originally, arc welding used carbon rods for electrodes, but these did not add material to the weld, so metal filler rods were added. Modern methods have essentially replaced carbon arc welding by providing quality solutions to the welding requirements. In some methods, to prevent oxidation of the molten metal, electrodes are coated with flux material that melts during the welding process. An inert gas such as helium or argon also serves to prevent oxidation (Kim 1994).

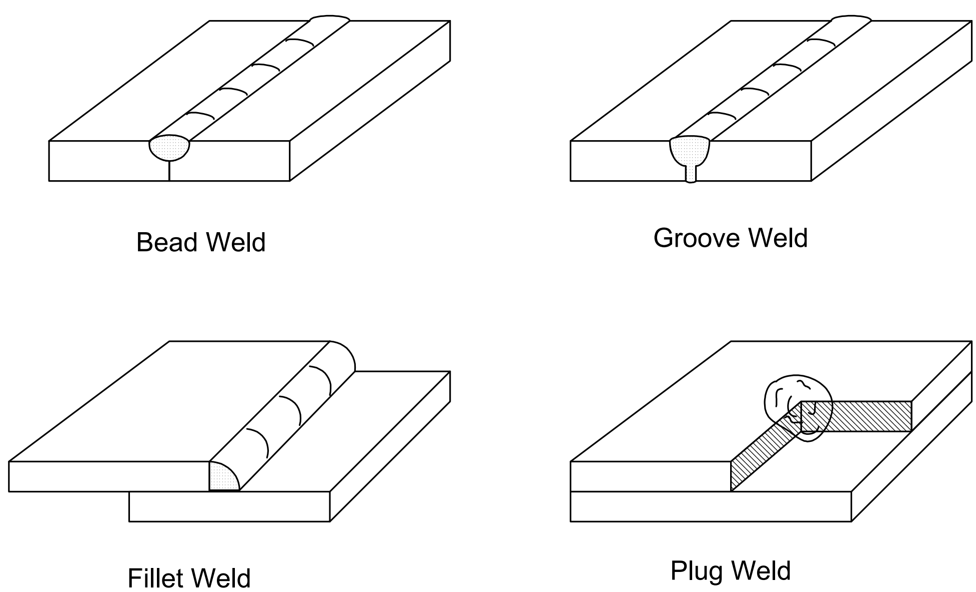


Figure 2-2 Four basic types of fusion welds (Degarmo 1984)

There are four basic types of fusion welds, as illustrated in Figure 2-2. Bead welds require no edge preparation. Because the weld is made on a flat surface and the penetration is limited, they are suitable only for joining thin sheets of metal, for building up surfaces, or for applying hard facing metals.

Groove welds are used where full-thickness strength is desired on thicker materials. These require some type of edge preparation to make a groove between the abutting edges. V, double V, U, and J configurations are the most common, usually produced by oxyacetylene frame cutting. The type of groove configuration depends primarily on the thickness and material property of the workpiece, the welding process to be employed, and the position of the work. Special consumable insert rings or strips are often used to assist in obtaining proper spacing between the mating edges and to aid in assuring proper quality in the root pass. These are especially useful in pipeline welding, particularly under field conditions and where the welding must be done from only one side of the workpiece.

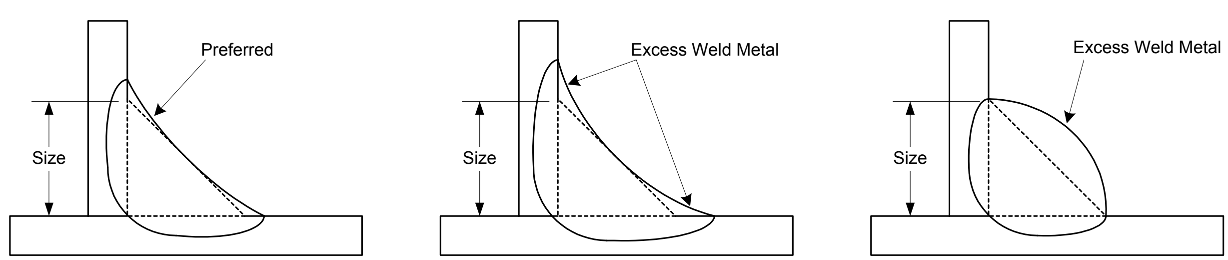


Figure 2-3 Preferred shape of fillet welds (Degarmo 84)

Fillet welds are used for tee, lap, and corner joints. The size of fillet welds is measured by the leg of the largest 45° right triangle that can be inscribed within the contour of the weld cross section. This is shown in Figure 2-3, which also indicates the proper shape for fillet welds to avoid excess metal and to reduce stress concentration. Fillet welds require no special edge preparation. They may be continuous or made intermittently, with spaces being left between short lengths of weld.

Plug welds are used to attach one part on top of another, replacing rivets or bolts. A hole is made in the top plate, and welding is started at the bottom of this hole. These welds can offer substantial savings in weight as compared with riveting or bolting.

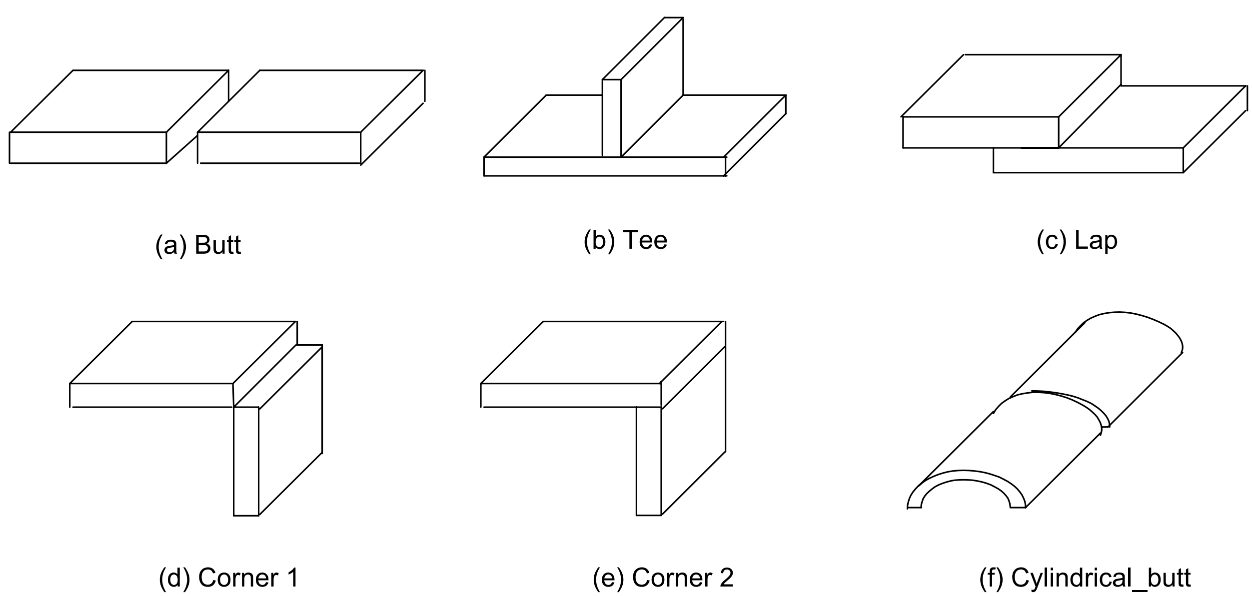
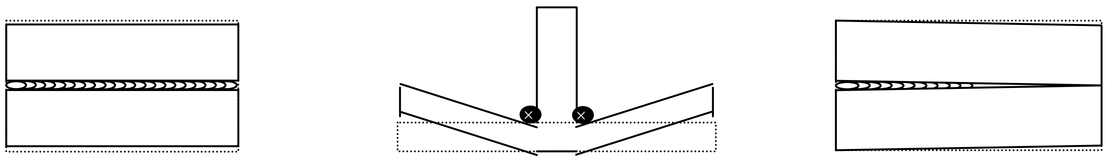


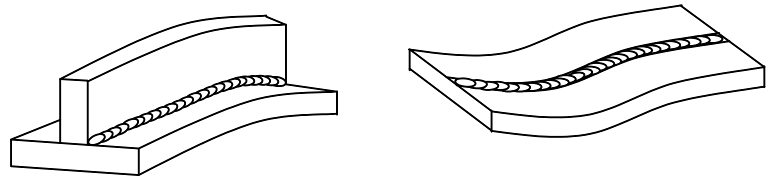
Figure 2-4 Basic types of weld joints

Figure 2-4 shows the basic types of joints that can be made through the use of bead, groove, and fillet welds. In selecting the type of weld joint to be used, the primary consideration should be the type of loading that will be applied. Too frequently, this basic fact is neglected and a large proportion of what are erroneously called "welding failures" are the result of such oversights (Kim 1994). Secondary factors in joint selection are cost and accessibility for welding. Cost is affected by the required edge preparation, the amount of weld metal that must be deposited, the type of process and equipment that must be used, and the speed and ease with which the welding can be accomplished. The combination of a joint design and welding loads generates various physical effects. Thermal distortion is an important physical effect of welding and an indispensable consideration to achieve quality welding. In this work, those considerations are deliberated to evaluate assembly design alternatives and to support an assembly design decision.

2.1.1.1 Welding Distortion. Due to the highly localized transient heat input from arc welding, considerable residual stresses and deformations, such as welding distortion, welding shrinkage, and welding warpage occur during heating and cooling in the welding cycle. In contrast to load stresses, residual stresses are internal forces occurring without external forces. Plastic upsetting generated during heating is concomitant with strains. The stresses resulting from the strains incorporate and react to generate the internal forces, which then can cause deformations (i.e., bending, buckling, and rotation). Figure 2-5 shows typical deformation shapes classified by their aspect (Masubuchi 1980).



a) Transverse shrinkage b) Angular change c) Rotational distortion



d) Longitudinal shrinkage e) Longitudinal bending distortion f) Buckling distortion

Figure 2-5 Various types of weld distortion (Masubuchi 1980)

Figure 2-6 shows the shape of a heat-affected zone, which is like a dumbbell because welding time on starting and ending crater points takes longer than on the midpoints of weldlines (Masubuchi 1980). In order to improve weld robustness, multi-layer welding is used in a thick-walled construction, as well as in an aluminum alloy welded construction. By using multi-layer welding, tensile residual stress and brittle fracture resistance can be reduced. Generally, in multi-layer welding as compared with single layer welding, longitudinal residual stress and shrinkage are mitigated. In fact, transverse residual stress and transverse and angular distortion increase in multi-layer welding (Radaj 1992). The superimposition of multiple heatings has a certain temperature cycle. Figure 2-7, which is adapted from Radaj (1992), represents the temperature cycle of a short length weld seam. In Figure 2-7-a, the temperature increases quickly at the start of the second layer, and then begins to fluctuate while tapering off as a result from the subsequent welding layers. Similarly, the final layer has been heated due to the remaining heat flux present in previous welding (Figure 2-7-b). Distortion can be minimized through controlling cooling time between layers and appropriate weld sequencing. Controlled cooling time allows heat to remain beneath the current weld bead. For example, preventing second welding layer from cooling too fast gives the effect of preheating before the next weld pass is made.

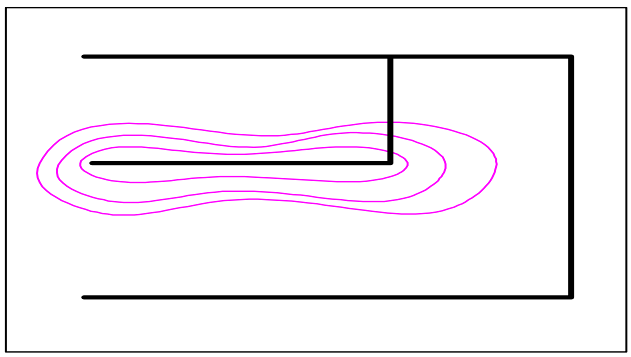
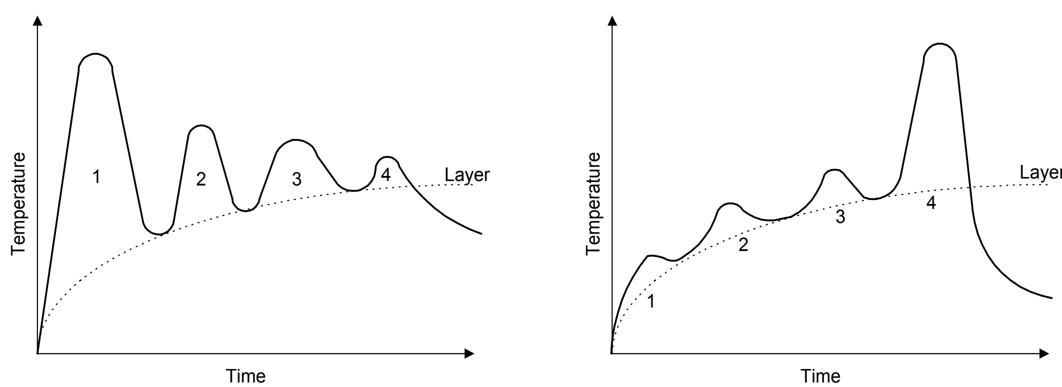


Figure 2-6 Heat-affected zone around a weld seam



a. First welding layer

b. Final welding layer

Figure 2-7 Temperature cycle in the multi-layer welding of a short weldline, adapted from Radaj (1992)

There has been considerable research to investigate weld joint design and analyze welding physical effects. Tsai *et al.* (2001) investigated the effects of welding parameters and joint geometry on the magnitude and distribution of residual stresses on thick-section butt joints. Milewski and Barbe (1999) modeled and analyzed laser melting within a narrow groove weld joint. Their model was developed to design and optimize laser weld joints and used to predict spatial and intensity effects of the joint. Chang *et al.* (1999) developed a three-dimensional numerical analysis model for weld-bonded joints. Normal stress and shear stress distributed at the edges of a spot weld and in the lap region were computed for weld-bonded joints, which were made with adhesives of different elastic modulus or thickness. Weaver (1999) developed a shell element model to determine weld loads and throat. Weld sizes were determined based on throat shear against the electrode.

Jeong and Cho (1997) presented an analytical solution to predict the transient temperature distribution in fillet arc welding, including the effect of the molten metal generated from the electrode. Moon and Na (1997) proposed mathematical models and neural networks to optimize welding process variables necessary to obtain the desired weld bead shape. Tarng *et al.* (1999) developed an application of neural networks and simulated annealing algorithms to model and optimize gas tungsten arc welding.

2.1.2 Riveting Operation

A common method of permanent or semi-permanent mechanical joining is riveting (Kalpakjian 1995). Thousands of rivets may be used in the construction and assembly of many structures, such as airplanes, ships, automobiles, etc. Installing a rivet consists of placing a rivet in a hole and deforming the end of its shank by upsetting (heading). Sufficient compressive elastic energy must be stored in the components to ensure that the rivet is placed in tension by stress relaxation when the compressive forging pressure is released. Figure 2.8 shows the riveting process. The quality of the riveted joint depends on the preparation of the hole and the control of the punch pressure cycle. The rivet design should be determined by considering the required strength of the assembled joint, the required ductility of the rivet material, and the control of the forging process.

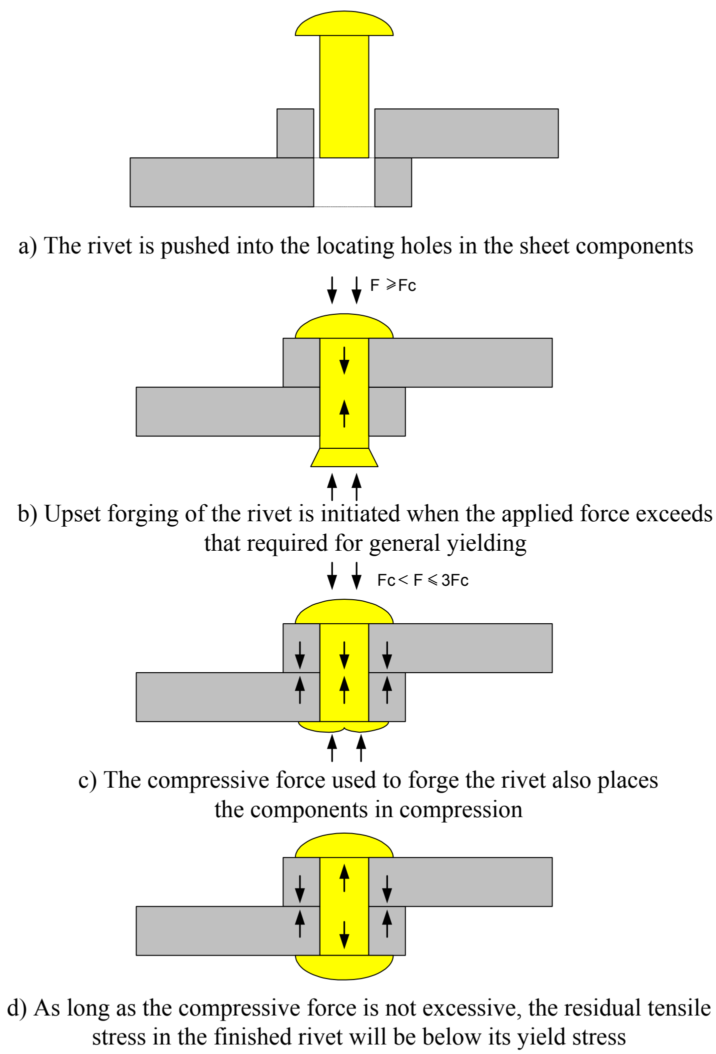


Figure 2-8 Forming a riveted joint (Brandon and Kaplan 1997)

Assuming the rivet to be elastic to the yield stress, to have the same compliance as the material being joined, and no work-hardening beyond the yield stress, then the compressive load required to forge a rivet of radius r is: $F_c = \pi r^2 \sigma_y$, where σ_y is the yield stress (Brandon and Kaplan 1997). If the compressive load is increased beyond this point, the elastic energy is stored in the joint assembly (Figure 2-8). Brandon and Kaplan (1997) analyzed the situation approximately. As the compression load is released, the sign of the stress in the rivet is reversed, placing it in tension. If the forging force exceeds $3F_c$, then the relaxation process will place the rivet under a tensile stress, which exceeds its yield stress, and reverse plastic flow will occur. The residual tensile stress in the rivet increases linearly from zero to σ_y as the forging force increases from F_c to $3F_c$. The tensile strength perpendicular to the riveted joint will be a maximum when the forging force is the minimum required for general yielding, $F=F_c$. In that case, the rivet experiences no prestress, but the tensile strength parallel to the joint is improved by the tensile residual stress in the rivet. It is because the frictional force at the interface reduces the stress concentration at the rivet by assisting load transfer to the components (Figure 2-9). The optimum upsetting conditions thus depend on the expected stresses in service, and will be somewhere in the range $F_c < F \leq 3F_c$.

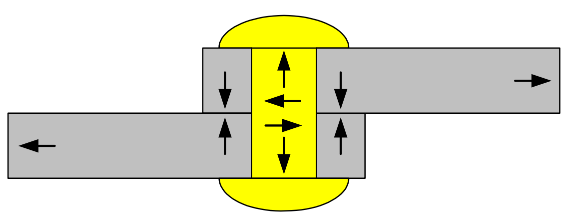


Figure 2-9 Residual stress in a rivet (Brandon and Kaplan 1997)

Recently, there has been significant research to investigate and analyze rivet joints. Xiong and Bedair (1999) developed modeling procedures for the stress analysis of riveted lap joints in aircraft structures. They used an analytical method to determine the stresses in jointed plates containing single or multiple loaded holes. They also employed numerical methods and finite element analyses to simulate the rivet-hole interaction. One of their conclusions was that the linear analysis using spring elements predicts accurate overall stress distributions in the rivet joint but it is not appropriate for determining the peak stresses at the junctures between the contact and non-contact regions. Menzemer *et al*. (1999) investigated shear failure of rivet joints in aluminum alloy 6061-T6 and developed an experimental and analytical program to study shear failure. They concluded that block shear failure is a potential limiting state for connection plates having mechanical fasteners and should be considered in the assembly design process. Rahman *et al*. (2000) presented stress and fracture analyses of corner and surface cracks at a rivet hole to predict the crack growth and residual strength of the riveted joints. They illustrated the essential features of cracks at countersunk rivet holes and the effects of the shape of the crack, the location of the crack, the length of straight-shank hole, and the loading condition. Fawaz (1998) used 3D virtual crack closure techniques to calculate stress intensity factors for aluminum riveted lap-slice joints. He showed that the rivet load distribution on the bore of the rivet hole greatly influences the stress intensity factor solution for small cracks. Ryan and Monagham (2000) investigated failure mechanism of riveted joints in fiber metal laminates and compared their results with typical aluminum alloy fuselage material (2024-T3). They noted that if localized compressive hoop stresses in the panels were known before design, it is beneficial to the fatigue life of the joint.

2.2 Assembly Design Formalism

Related to assembly design formalisms, Deneux (1999) discussed the necessity of assembly feature in the design of a complex assembly. Van Holland and Bronsvoort (2000) proposed the concepts of a single-part feature model and an assembly feature model for assembly modeling. However, their assembly feature concepts, which considered only assemblies with mechanical fastening, cannot be employed to assembly modeling requiring various other joining processes. Whitney *et al.* (1999, 2001) proposed a formalism for assembly design and focused on only fully constrained assemblies and subassemblies. Even though they presented a general methodology for assembly design, many spatial relationships in actual mechanical assemblies, such as between two cylindrical surfaces and between a spherical surface and other surfaces were not addressed. In addition, the effects on spatial relationships from joining processes were not discussed.

Rémondini *et al.* (1998) proposed assembly operators to deal with the interface between geometric models and analysis models. Their research focused on geometric aspects of the mechanical analysis model and didn't include the formalism to capture the information about joining methods of assembly, which is essential to represent the relationship between the mechanical analysis model and the joining method. Fu *et al*. (1993) used graph grammar to represent and transform geometric features. Indeed, the method was to represent geometric features of single parts and thus it has a limitation on representing joining relationships between geometric features.

None of these researchers has developed an assembly formalism that accommodates assembly operation tools that can predict mathematical and mechanical-physical implications of the joining process in a collaborative assembly design environment. Thus, there is a strong need to develop an assembly formalism to capture designer intent on assembly design and to consider assembly processes and their effects.

2.3 Product Assembly Modeling and Spatial Relationships

Spatial Relationships (Liu and Nnaji 1991) were first proposed by Ambler and Popplestone (1997) in 1975 to describe the relative positions of parts in their final state by specifying feature relationships among them. The spatial relationships include *against, coplanar, fits, parax, lin, rot,* and *fix*. In the work of Ambler and Popplestone, the spatial relationships are concentrated on the configuration of a part. Liu and Nnaji (1991) focused on the mechanical assembly specifications as well as the configuration of a product, so that spatial relationships can be applied to general assemblies and are capable of accepting the design specifications. They defined design with spatial relationships not only for inferring the assembly positions, but also to capture designer's intentions. Each spatial relationship (e.g., *against, parallel, aligned, incline-offset, include-angle*, etc.) (Figure 2-10), constrains the degrees of freedom (d.o.f.) of motion between the mating entities (face, centerline, center point, etc.). For example, two faces are said to be against if the two faces are touching at some point and normal vectors of those faces are in opposition where they touch. Any combination of two features can possess this property. Spatial relationships support three different types of against relationships as follows:

1) Two planar faces are against one another,

2) A cylindrical feature touches a planar face along a line,

3) A spherical feature touches a planar feature at a point.

Two features are aligned if their centerlines are collinear. By selecting the appropriate combination of spatial relationships, the relative mating position with moving d.o.f. of motion can be inferred. The spatial relationships maintain this relative mating position irrespective of the size of the mating entities. Liu and Nnaji (2003-a, 2003-b) applied spatial relationships in their work, which evolved a framework for a collaborative design advisory system, based on a constraint-based product modeling environment, for mechanical assemblies. The detailed descriptions of other spatial relationships can be found in Liu and Nnaji (1991).

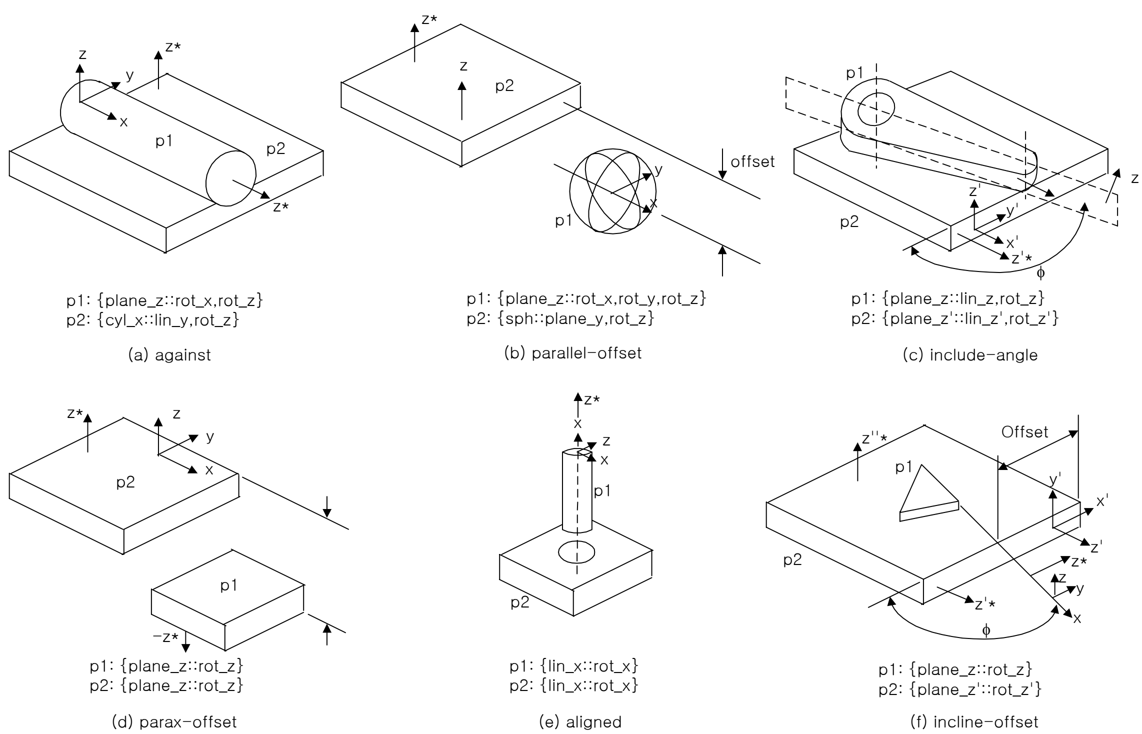


Figure 2-10 Six types of spatial relationships

There has been research to describe assembly relations by specifying spatial relationships (Liu and Nnaji 2003-b, Kim *et al.* 2003-a). Most notably, RoboWeld-S, an automatic process planner for robotic arc welding of Sheet Metal Weld Assemblies (SMWA) (Liu and Nnaji 2003-b, Kim *et al.* 2003-a) was developed. Spatial relationships and assembly feature formations relevant to the feature-based modeling of SMWA were presented in the research. Still, their methodology does not provide a joining protocol to explain the physical and mathematical effects before/after joining. Their assembly feature should be expanded to fully represent assemblies requiring joining processes.

In this work, an efficient assembly formalism is introduced to represent general assemblies requiring assembly operations. Design with spatial relationships is the kernel of this assembly design formation.

2.4 Virtual Prototyping

Prototyping technologies are emerging as powerful tools that shorten the product design and development process (Swaelens and Kruth 1993, Pratt 1994). Virtual prototyping has been in steady development since the 1970s (Chua *et al.* 1999). This practice implies the testing and analysis of 3D solid models on computing platforms. Fang and Liou (1997) developed a computer prototype modeling system for mechanical assemblies with deformable components. Srinivasan *et al*. (1999) and Shyamsundar *et al*. (1998) presented methodologies to perform selective disassembly analysis from the assembly model. Several researchers have presented frameworks for transparent analysis and attempted to integrate CAD and analysis tools. Su and Amin (2001) proposed a CGI (Common Gate Interface) based system for executing software programs via the Internet and tested their system on a gear optimization software package. They acknowledged that their system inherited the limitation of the CGI approach (i.e. speed and multiple user handling). In fact, they didn't discuss a transparent analysis methodology to capture the physical effects of assembly operations in a heterogeneous computing environment. Gee (2001) proposed an agent-based system to integrate CAD, FEM (Finite Element Modeling), and CFD (Computational Fluid Dynamics) in distributed and heterogeneous computing systems. He presented a framework for the integration between CAD and analysis systems; however, he did not discuss about how to capture the mathematical and physical implication of joining operations in a collaborative design environment. Sahu and Grosse (1994), Sheehy and Grosse (1997), and Shanbhag *et al*. (2001) have researched the integration of CAD and finite element modeling and presented some solutions to enable adaptable analysis. Shanbhag *et al*. (2001) employed meta-objects to achieve seamless exchange of data between analysis models. Goriatchev *et al*. (2001) developed a distributed system for CFD simulation. They used Java-enabled technology to achieve a distributed computation environment.

2.5 Distributed Assembly Design

Concurrent Engineering offers substantial benefits for new product development and many companies are taking a strong interest in this collaborative approach. Distributed assembly design is just starting to emerge as a viable alternative to the traditional assembly design process, in which an assembly design can be developed via an iterative process among designers, manufacturers, marketing people, and ultimately customers. This emergence can be linked to the recent outburst of growth in the development of the Internet and associated technologies such as the Java programming language, as well as the rapid advancements made in computing technology that have led to the proliferation of powerful, yet affordable computers. In fact, there are already several systems, both commercial and research, that are investigating the formalisms and protocols necessary for collaborative engineering (Cutkosky *et al*. 1996, Kim *et al.* 1998, Mueller 1999, PTC 2000, Gadh 2002). Nevertheless, it is not still fully clear how assembly design should be implemented in collaborative design environments (Rojas and Songer 1999, Krishnamurthy and Law 1997, Florida-James *et al*. 2000).

Several research teams have generated partial solutions to the distributed concurrent design and development problem. Wagner *et al*. (1997) performed a feasibility study for how the Internet can be used as an interactive resource during design and manufacturing process and they tested their concept on a simple fixture design. Cheng *et al*. (2001) presented a methodology to implement an Internet and Java-based design support system. Boujut *et al*. (1997) presented a distributed design system for the design of forged parts in an attempt to achieve agility in design and manufacturing. Their works still have limitations on evaluating assembly models with respect to manufacturability, assemblability, and "joinability."

2.6 Collaborative Assembly Design

The design of a mechanical product requires concurrent availability of dozens of technical supports from various engineering and non-engineering fields, such as drawing, material, manufacturing process, quality, marketing, maintenance, government regulations, etc. There have been many computational tools in those different areas. However, there are still problems that inhibit them to work together automatically with little human intervention. Problems come mostly from the lack of common protocols for them to communicate, such as different CAD data formats, different computer operating systems, different programming languages, etc. The Internet provides an opportunity for these engineering tools to work together and utilize these services optimally. To connect these "islands of automation," universally accepted protocols are needed.

Existing research is focused mainly on the feasibility for product design and manufacturing collaboration using networked computers in a distributed environment. The importance of design collaboration has gained the attention of industry (NSF Workshop 2000, FIPER 2001, OneSpace 2002, Windchill 2002, CATIA 2002). Meanwhile, several academic research groups have studied the possibility of distributed environment for product designers and manufacturers. Next-Cut (Brown *et al*. 1989) permits human and computational agents to cooperate in design and manufacturing through a central knowledge base. CyberCut (Smith and Wright 1996, Chui and Wright 1999) allows remote designers to access distributed servers and perform functions of CAD, CAPP, and CAM through the World Wide Web (WWW), in which design, planning, and fabrication agents communicate using direct socket connections. FixtureNet (Wagner *et al*. 1997) provides interactive fixture design service on the WWW by considering possible modular fixtures for a given part. The communication between users (HTML pages and Java Applets) and the fixture design server is through a Hypertext Transfer Protocol (HTTP) server by socket connection. COCADCAM (Kao and Lin 1996, Kao and Lin 1998) allows two geographically dispersed CAD/CAM users to work together on co-designing through distributed CAD/CAM modules. Unix Interprocess Communication (IPC) and Network File System (NFS) are used for data file transmission between CAD/CAM modules and local clients, while a socket interface is used for client-server and server-server communication. Larson and Cheng (2000) developed a web-based interactive cam design system, where the cam profile, transmission angle, position, etc., can be designed through web browsers. WPDSS (Quang *et al*. 2001) supports commercial CAD software to perform collaborative design through the WWW. This group of research achieves data exchange by WWW protocols and/or direct socket connections.

Some research utilizes middleware technologies for communication. Han and Requicha (1998) developed a distributed system for feature recognition. Clients such as feature recognizers, feature-based design systems, and graphics renderers communicate with a central geometry server by Remote Procedure Call (RPC) protocols. DOME (Pahng et al. 1998-a, Pahng et al. 1998-b, Abrahamson et al. 2000) is a framework for the modeling and evaluation of product design problems in a computer network-centric design environment. Design problems are decomposed into modular subproblems in order to distribute responsibility among designers. Communication among modules is completed using Common Object Request Broker Architecture (CORBA) protocols. NetFEATURE (Lee *et al.* 1999) includes web-enabled feature modeling clients, neutral feature model servers, and database managers. Agents are defined on server-side to serve clients for feature modeling by means of CORBA protocols. Mervyn *et al*. (2003) employed Java RMI (Remote Method Invocation) and XML technologies to realize an interactive fixture design system.

None of these researchers addressed the integration between joining design and analysis to consider the various effects of assembly operations during the actual assembly design process in an Internet-based collaborative design environment. In this work, joining analyses processes are integrated into the actual joining design process in a transparent and remote manner.

2.7 Service-oriented Collaborative *e*-Product Design and Realization

The worldwide availability of technology, capital, information, and labor makes today's manufacturing enterprises global. Within this distributed economic and technological environment, the problem of how to let engineers collaborate globally during product development periods arises. Information incompleteness, inconsistency, and improccessability are problems that collaborative design groups are facing. Collaborative design tools are needed to improve the collaboration among distributed groups, endorse knowledge sharing, and assist better decision making.

Instead of looking at various engineering tools from a traditional computation viewpoint, Nnaji *et al*. (2003-b) focused on the engineering implications of those tools from a more abstract level. This approach assures good openness for collaborative design and engineering systems. Their view of design collaboration is service-oriented. With the rapid growth of the number of networked computers, a tremendous amount of resources is available online. The Internet is no longer a simple network of computers. From an application perspective, the Internet is a network of potential services. For example, in a three-tier web-based database system consisting of a web browser, server, and database, the web browser provides web document presentation services for human users; the web server provides data processing and retrieval services for the web browser; and the database provides data storage services for the web server. The Internet can be regarded as a complex system of service chains. Computer-aided design and engineering tools can be linked to the design platform through the Internet to provide certain services resulting in a distributed product development environment. This incorporates different engineering services and makes them available for automatic transactions in a collaborative assembly design environment. This product development environment is called an "*e*-product design and realization environment."

Service oriented product design was implemented on the Pegasus system (Nnaji et al. 2003-b), which is an *e*-product design and realization platform for mechanically engineered products being developed by the National Science Foundation Industry/University Cooperative Research Center (NSF I/UCRC) for *e*-Design and Realization of Engineered Products and Systems at the University of Pittsburgh. The new *e*-design paradigm results in remote multidisciplinary, direct preference and constraint imposition on a design object. Also, it results in evolving a methodology to represent functional requirements and transitioning from concept generation to form realization. This revolutionary design paradigm allows design platforms to call on design service tools from the Internet and for customers and supply chain vendors to participate in product design. The *e*-design system requires virtual prototyping and transparent analysis on the CAD platform. Such analysis should be transparent to the designer within the design platform. The assembly operation tools developed in this work were implemented on the Pegasus architecture as a "plug-and-play" assembly design/analysis module. The assembly operation tools are currently serving the Pegasus system as a service provider.

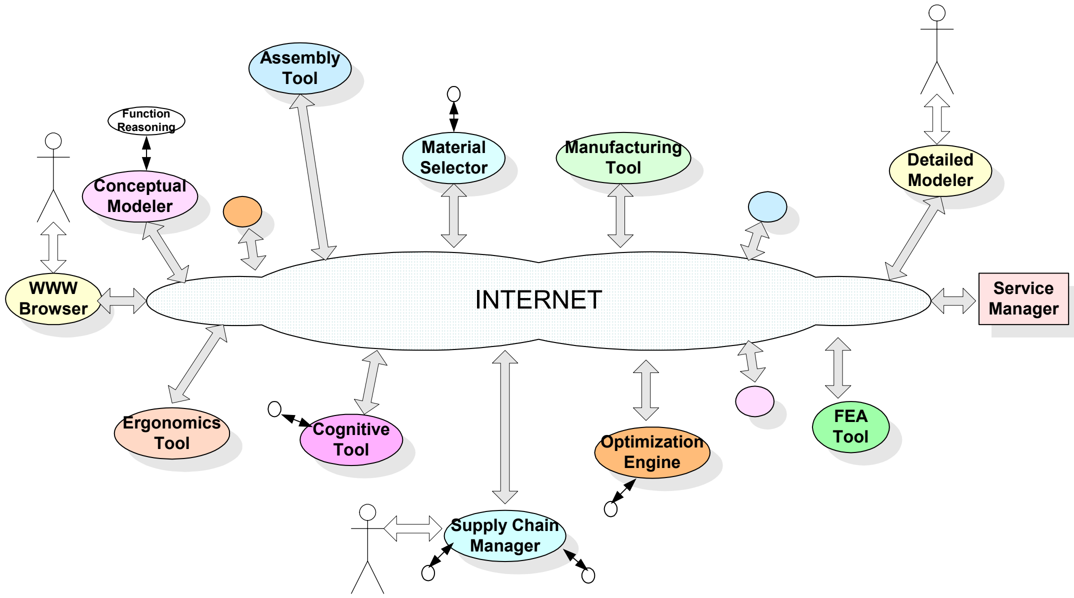


Figure 2-11 Peer-to-peer relationships in service-oriented architecture

In a service-oriented collaborative architecture, CAD/CAM/CAE tools can be linked to the Internet to form a distributed product development environment, which incorporates different engineering services over the Internet, and making them available for transparent transactions in product development. Each design tool for a designer system can be a server that provides certain services requested by clients, either within or external to the designer system (Nnaji *et al*. 2003-b). As shown in Figure 2-11, servers within the system have a peer-to-peer relationship with each other.

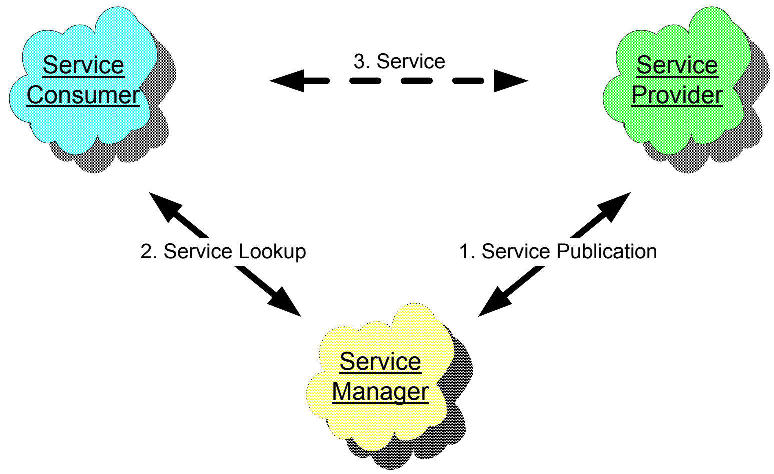


Figure 2-12 Service triangular relationship

Service is defined as a process that provides a functional use for a person, an application program, or another service in the system. Services should be specified from the functional aspect of service providers. To make an existing tool available online or to build a brand new tool for such a system, services associated with this tool should be defined. The service transaction among service providers, service consumers, and the service manager within this architecture is illustrated in Figure 2-12. Once a service is registered at a central administrative manager, called the service manager, it is then available within the whole system. This process is *service publication*. When a service consumer within the system needs a service, it will request a lookup service from the service manager. This process is *service lookup*. If the service is available, the service consumer can request the service from the service provider by the aid of the service manager. Most importantly, this service triangular relationship should be built at run-time. The service consumer (client) does not know the name, the location, or even the way to invoke the service from the service provider (server) during the system and tools development period.

Service providers that provide different services, such as assembly design component repository, finite element analysis (FEA), material, etc., can be developed independently. As shown in Figure 2-11, servers that can provide different engineering services are linked by the Internet. Each node in this network can both require and provide certain services; thus, it can be both client and server at different times. The client/server relationship is determined at run-time, so the system is open for the future expansion and extension, when more services become available.

Assembly design participants, such as customers, suppliers, assembly designers, production engineers, and other stakeholders, need to exchange assembly design information seamlessly in a collaborative environment. There should be common data models and protocols available for them to share information. Those models and protocols should be widely acceptable and easily implementable, as well as efficient for information transferring. In this work, assembly design (AsD) models are generated based on the AsD formalism and exchanged among design participants through the service-based architecture. In addition, a virtual assembly analysis tool integrates assembly design and analysis transparently and remotely through service-oriented architecture.

3.0 ASSEMBLY DESIGN FORMALISM AND ASSEMBLY DESIGN ENGINE

An *assembly relationship* indicates which components are assembled and their *joining relationships*, where the *joining relationship* denotes how assembly components are joined. For example, two plates, A and B, are assembled by a welding operation. The plates A and B have a joining relationship of welding. To fully describe this assembly, detailed information related to the assembly/joining relationship should be captured in an assembly design. The assembly formalism, capturing assembly/joining relationships of a product assembly, is comprised of five phases: 1) spatial relationship specification, 2) mating feature extraction, 3) joint feature formation and extraction, 4) assembly feature formation, and 5) assembly engineering relation extraction (see Figure 3-1). Each of these phases will be described thoroughly in the following sub-sections.

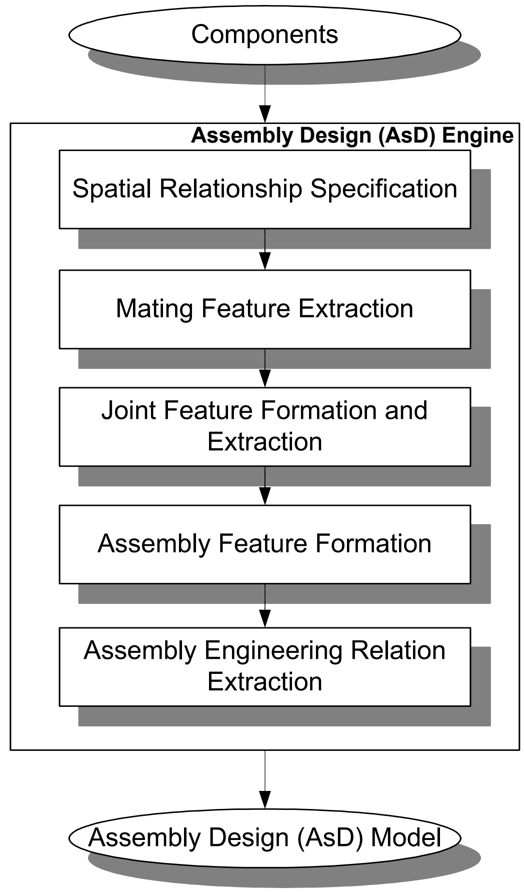


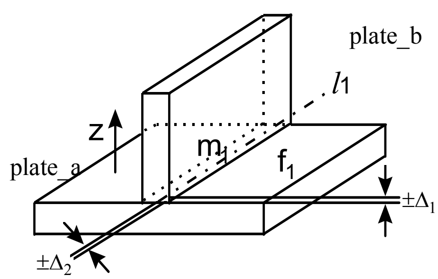
Figure 3-1 Procedures of the assembly design formalism

3.1 Spatial Relationships Specification

By interactively assigning spatial relationships, the designer can assemble components together to make final products and infer the d.o.f. remaining on each of the components. In the work of Ambler and Popplestone (1975), the spatial relationships are concentrated on the configuration of a part. Liu and Nnaji (2003-b) focused on the assembly specifications as well as the configuration of a part, so that spatial relationships can be applied to a general assembly and are capable of accepting the design specifications. Some of the spatial relationships are defined in the following paragraphs.

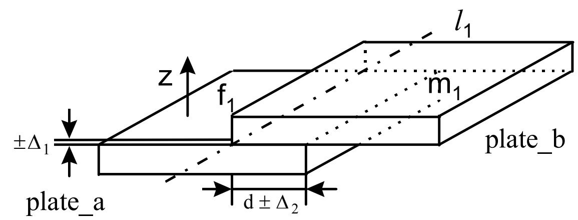
Against is defined as two faces that touch at some point. Any combination of two faces can possess this property. There are three different types of against relationships as follows: 1) two planar faces are against one another, 2) a cylindrical feature touches a planar face along a line, and 3) a spherical feature touches a planar feature at a point. Two features are *aligned* if their centerlines are collinear. Application examples include insertion and any assembly requiring an alignment with cylindrical shafts or holes.

The types of d.o.f. are classified as follows (Nnaji *et al*. 1993); *lin_n*: linear translation along *n* axis, where, *n* contains a fixed point and a vector; *rot_n*: rotation about *n* axis, where, *n* contains a fixed point and a vector; *cir_n*: translating along a circle with *n* axis, where, the fixed point of *n* is the center of the circle and vector of *n* is perpendicular to the circle; *plane_n*, *cyl_n*, and *sph*: translating along a planar, cylindrical, spherical surface.



plate_a	plate_b	spatial relationship	tolerance	d.o.f.
f_1	m_1	against	$\pm\Delta_1$	{plane_z::rot_z}
l_1	l_1	aligned	$\pm\Delta_2$	{lin_l_1::lin_l_1}

a) T-joint



plate_a	plate_b	spatial relationship	overlap	tolerance	d.o.f.
f_1	m_1	against	.	$\pm\Delta_1$	{plane_z::rot_z}
l_1	l_1	aligned	d	$\pm\Delta_2$	{lin_l_1::lin_l_1}

b) Lap_joint

Figure 3-2 Examples of assemblies and their spatial relationships

Figure 3-2 shows assembly modeling examples by spatial relationship. Plate_a and plate_b represent two plates engaged in spatial relationships. Plate_b is joined onto a relatively fixed plate_a and either plate can be a component plate of a sub-assembly. l_1 is an intersecting line. The tables in Figure 3-2-a and Figure 3-2-b provide the information on spatial relationships, geometric tolerances, and d.o.f. associated with each assembly. The d.o.f. of a part are expressed as {degrees of freedom of moving within the coordinate system of the relative moving part :: degrees of freedom of moving within its own coordinate system}, with respect to the other mating parts of the assembly. A detailed description on this can be found in Nnaji *et al*. (1993). For a *plane_z*$_a$ d.o.f., a body may move on a planar surface along two *lin*. When a new *plane_z*$_b$ is introduced, the remaining d.o.f. are derived by intersecting these two *planes*. The intersection of surface d.o.f. is available only when a plane is involved, such as, *circle_n* is the result of intersecting *plane_z*$_a$ with *cyl_n* (or *sph_n*) together. In the intersection of two rotational d.o.f., say *rot_z*$_a$ and *rot_z*$_b$, if they share the same rotational axis then *rot_z*$_a$ or *rot_z*$_b$ remains; if not they cancel each other. Nnaji *et al*. (1993) presented some general reduction rules for d.o.f. In assigning spatial relationships, the mating features are defined and extracted from the parts.

3.2 Mating Feature Extraction

Generally, a feature is a region of interest within a part or an assembly. Features might be considered from the aspect of functionality, manufacturing, inspection, assembly, etc. In other words, features are defined by attaching some sort of attributes according to the user's intention on the design. For assembly modeling, it is necessary to first define and extract the mating features for the operation of product assembly. Usually, two assembly parts do not make contact over their whole surface area; only features of each part are in contact. The mating features are then derived from these features in contact. The definitions of features used in this work are listed below.

1) A feature is defined as:

A set of geometric entities (surfaces, edges, and vertices) together with specifications of the bounding relationship between them and which imply an engineering function on an object (Liu and Nnaji 1991);

2) A form feature is defined as:

A set of geometric entities (surfaces, edges, and vertices) together with specifications of the bounding relationship between them and which have engineering/functional implications and/or provide assembly aid, such as a center line of a hole, on an object (Liu and Nnaji 1991, Shah and Rogers 1988);

3) A mating feature is defined as:

A set of component geometric entities of form features which are needed to relatively locate the parts according to their spatial relationships in the whole product assembly.

Mating features are very important for representing assembly and joining relationships between assembly components, because actual assembly operations occur at the mating features. Each spatial relationship has specific mating features. For example, when a planar surface of one part is assigned an against spatial relationship with a cylindrical surface of another part, the planar face and the cylindrical surface are the mating features of interest in the assembled pair (Figure 3-3-a). In this fashion, spatial relationships and related mating features provide fundamental elements used to describe assembly. One reason for this is to easily extract mating features universally from parts or pre-defined features, which are usually diverse because they are intent-oriented. From this point of view, the mating feature of a part could be a planar face, a centerline of a cylinder, an edge of a face boundary, etc. Therefore, the mating features are determined by the types of spatial relationships being specified and the geometric entities being selected (see Figure 3-3) and represented by the spatial relationships, mating component (selected form feature), and mating entities (selected geometric entities). Mating feature extraction is a preliminary step to capturing joining information; however, the mating feature extracted directly from spatial relationship specification, is not sufficient to represent joining processes. For example, in Figure 3-3-d, the centerline features are not enough to represent the welding operation since actual weld seams will be around the contact area of the two components. As shown in Figure 3-4, the mating feature between p1 and p2 should include the cylindrical surfaces. Unfortunately, detailed joining information cannot be directly captured from spatial relationships and mating features. The next step is a joint feature formation process.

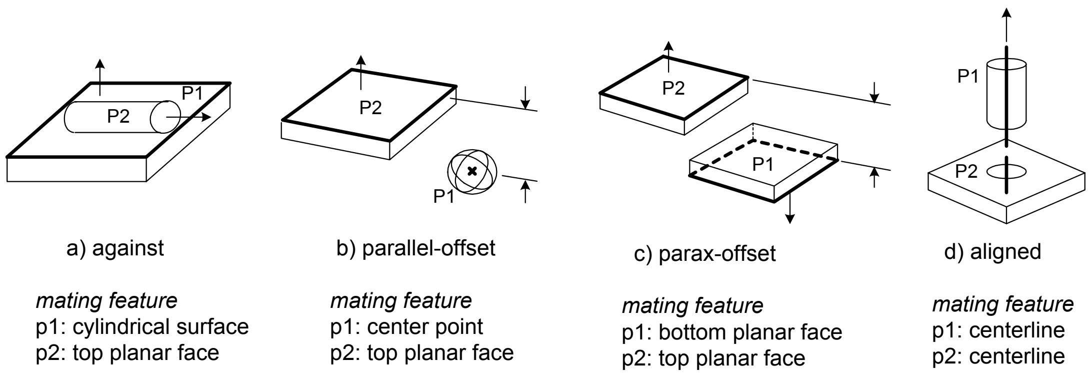


Figure 3-3 Spatial relationships and their mating features

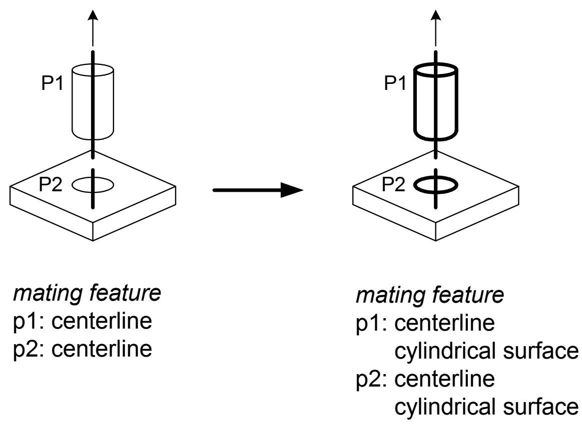


Figure 3-4 Mating features expansion of an *aligned* spatial relationship between a cylinder and a hole

3.3 Joint Feature Formation

Generally, joining processes, such as welding and gluing, happen on the mating entities, such as weld seams. Current mating features of the mating entities have limitations on representing special configurations for joining (e.g., weld seams and grooves). As described above, the mating features of Figure 3-3-d are not enough to represent a joining location (weld seam), a joining method (welding), and groove shapes. In this research, to enable the description of joining relationships, a new category of feature (joint feature) is defined. The joint feature is defined as:

A set of information including joining methods, groove shapes, joining components and entities, and joining constraints, which is used to represent assembly/joining relations.

The joint feature captures the information of actual joining and it is represented in a symbolic manner as follows:

{*joining method* | *groove shape* | [*joining components* (*joining entities*)] | [*joining constraint*]}

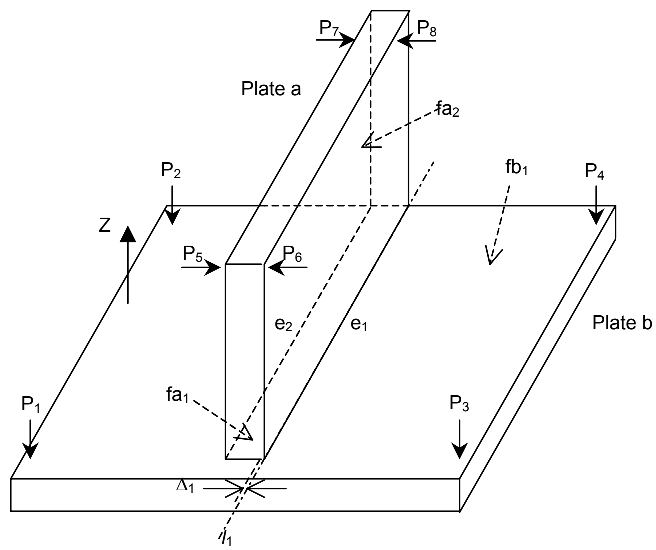


Figure 3-5 Example of a welded t-joint

For example, if a simple t-joint (Figure 3-5) is double-fillet welded on two weld seams (e_1, e_2) assisted by fixtures located at p_1, p_2, …, p_8, the joint feature can be described as {*gas metal arc welding* | *double fillet* | [fa_1 (e_1), fb_1 (l_1)], [fa_1 (e_2), fb_1 (fb_1)] | [p_i] }, i = 1, 2, ...,8. This joint feature is generated from the input from designer's specifications. The joining entities, e_1 and e_2, are part of mating features, fa_1 and fb_1; fa_1 is a bottom surface of plate a and fb_1 is a top surface of plate b. The joining entities should be part of the mating features extracted in the previous stage. If a designer specifies a geometric entity, which does not belong to the mating features as a joining entity, then it violates the validity of joining. After joint features are determined, the system is then ready to proceed to the next stage, assembly feature formation.

3.4 The Assembly Feature Formation Process

The purpose of assembly feature formation is to group the mating features and joint features together and thus integrate the data embedded at the component design stage with new assembly information for subsequent processes such as assembly analysis, assembly violation detection, process planning, etc. Having designated spatial relationships, mating features, and joint features, the system can then trace back to the component design stage to determine from which form features these mating features originate and what their design specifications are. The definition of an assembly feature is:

A group of assembly information including form features and joint features associated with mating relations and assembly/joining relations, such that the association includes a set of spatial relationships between mating features, mating bonds, material, remaining degrees of freedom, as well as other constraints implied by the original intents on the form features.

Table 3-1 Example of assembly feature (for the welded t-joint shown in Figure 3-5)

	Plate a		Plate b	
Mating Features	fa1 (planar_face)	l_1 (face_edge)	fb1 (planar_face)	l_1 (face_edge)
Spatial Relationships	against	aligned	against	aligned
Mating Bonds	MB1(against)	MB2(aligned)	MB1(against)	MB2(aligned)
Joint feature	{*gas metal arc welding* | *double fillet* | [fa_1 (e_1), fb_1 (l_1)], [fa_1 (e_2), fb_1 (fb_1)] | [p_i] }, i = 1, 2, ...,8.			
Material	Aluminum 6061-T6			
Designed D.O.F.	{plane_z :: rot_z}, {lin_l_1 :: lin_l_1}			
Implied Constraints	tolerance: $\pm\Delta_1$			

It is noted that the mating features provide links between assigning spatial relationships for assembling components and capturing engineering information (e.g. geometry and joining relationships) necessary for the succeeding process, and the assembly features act as media to carry and transmit all information to the downstream steps. Different assembly operations imply different information and thus corresponding assembly features should be defined for specific assembly operations. For instance, in weld assembly modeling, a weld assembly feature is composed of joint features for welding, mating features, spatial relationships, mating bonds for each spatial relationship, and implied constraints as shown in Table 3-1. The mating bond is used to capture detailed engineering relations among assembly components and is explained in the next section. Note that the design specifications such as dimensions, positions, and joining methods represent the design constraints at the component level and imply some constraints in mating and joining relations from the viewpoint of assembly and joining. An example of the implied constraints can be found in the case of that the tolerance of pin affects the tolerance between the hole and the pin. After the assembly feature formation steps are completed, assembly engineering relations of an assembly can be easily obtained from the assembly features.

3.5 Extraction of Assembly Engineering Relations

Assembly engineering relations of the entire assembly can be extracted based on the assembly features after specifying the spatial relationships and joining methods between components. A mating bond and a generic assembly relationship diagram (GARD) are used to represent the engineering relationships on the entire structure. The mating bond was originally introduced by Liu and Nnaji (2003-a) for assembly representation. However, the mating bond has a limitation on representing assembly engineering relations, in which joining is considered. In this research, the GARD is introduced to designate assembly engineering relations graphically and the mating bond is extended to pass necessary information to downstream assembly analyses. The mating bond and the GARD provide an efficient design data sharing mechanism in a collaborative assembly design environment.

The structure of mating bonds is shown in Figure 3-6. There are two dominant groups of information defined in a mating bond: mating pair and mating conditions. A mating pair contains two mating features involved in the joining. The inter-feature association of form features related to assembly is used to record form features, subassemblies or assemblies in which the form features associate. From the mating features, the system traces back to its original form features and inherits the implied constraints, such as tolerance of the hole. The mating conditions include the assigned spatial relationships, designed d.o.f., and assembly/joining relations. Figure 3-7 illustrates the concept of the mating bond with the mapping of two parts constrained in an aligned spatial relationship to the mating bond

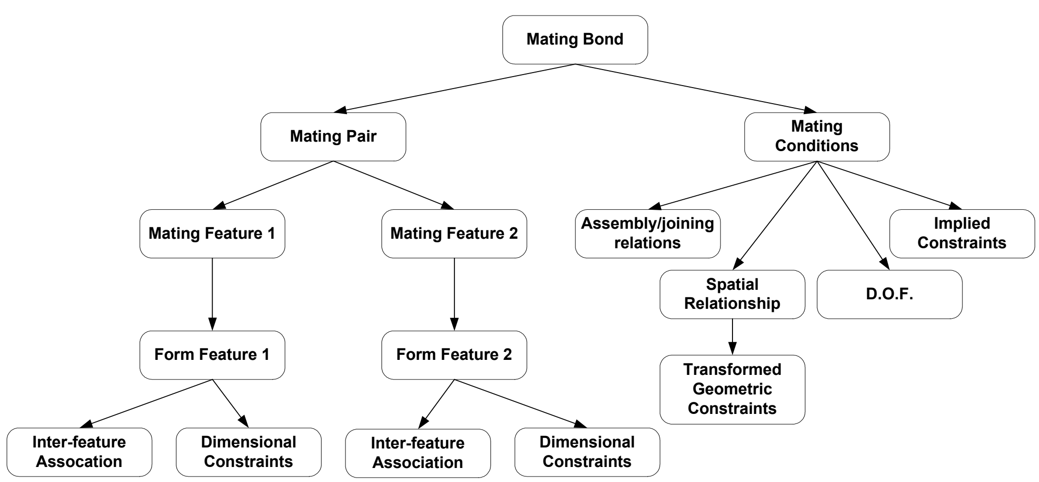


Figure 3-6 Mating bonds

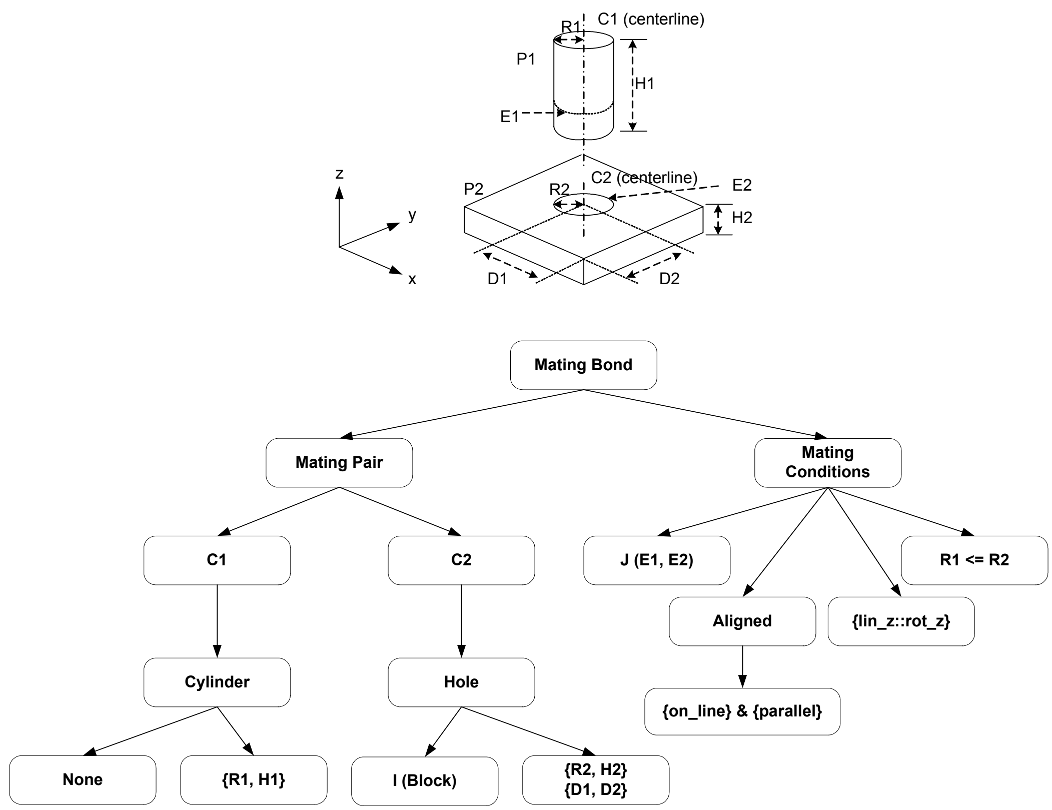


Figure 3-7 An example of mating bond for aligned spatial relationship

As mentioned above, assembly engineering relations of an assembly structure can be extracted based on the mating features and joint features after specifying the spatial relationships and joining methods between assembly components. A mating bond is created once two mating features on different components are selected and positioned with each other, and joint features are formatted. Assembly features are organized by a set of one or more mating bonds.

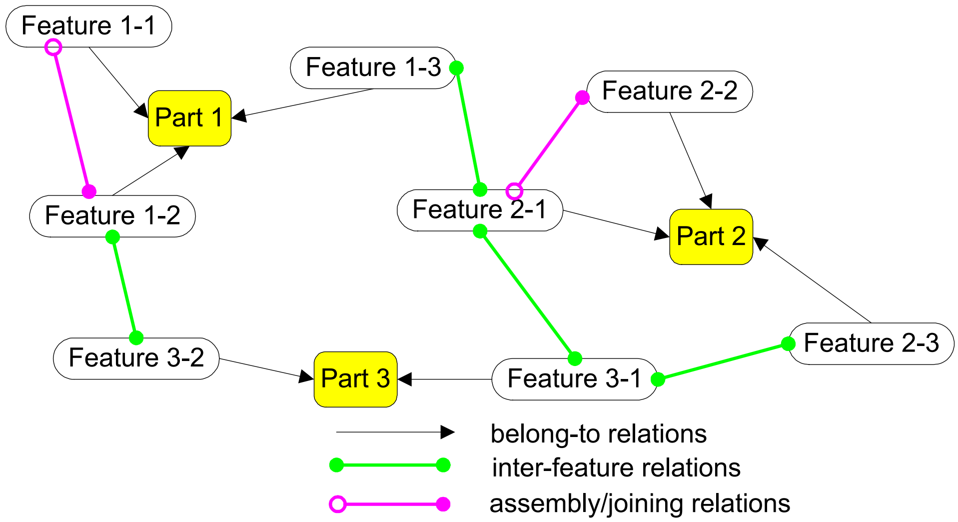


Figure 3-8 Assembly engineering relations among assembly components

The concept of assembly engineering relations is illustrated in Figure 3-8. Note that the lines with an arrowhead are interpreted as *belong-to* relations, the lines with solid roundheads as *inter-feature* relations, and the lines with one solid roundhead and one hollow roundhead as *assembly/joining* relations. In other words, each part has been completely designed with its associated features, and functional relationships are built between those part features by attaching some linkages (relations). A GARD is designated to represent graphically feature-to-feature linkages and feature-to-part linkages. A GARD represents assembly hierarchy and the connectivity of the whole product, based on these inter-feature association and assembly/joining relations. Inter-feature associations, assembly/joining relations, and GARDs are explained thoroughly in the following section. Mating bonds capture data structure of assembly engineering relations while GARDs represent these relations diagrammatically.

Notations:

FF: form feature; DC: dimensional constraint;

MF: mating feature; JF: joint feature;

AF: assembly feature;

MComp: mating component;

ME: mating element;

JComp: joining component; JE: joining element

JConst: joining constraint

MB: mating bond;

MP: mating pair; MC: mating condition;

JM: joining method; GS: groove shape

IA: inter-feature association relation;

J: assembly/joining relation;

P_j^i is a member of part class **P,** $P_j^i \in \mathbf{P}$.

FF_{jk} is a member of form feature class **FF,** $FF_{jk} \in \mathbf{FF}$.

$\mathbf{A_i}$ is an assembly structure class.

J is a member of the assembly operation class ϑ**,** J $\in \vartheta$.

R is a member of the relationship class $\Re$, R $\in \Re$.

DC_r is a member of dimensional constraint class **DC**, $DC_r \in \mathbf{DC}$.

RC_{pq} is a relational constraint between FF_{jp} and FF_{jq}, $RC_{pq} \in \{0, 1, 2\}$.

$$RC_{pq} = \begin{cases} \mathbf{0, if}\ FF_{jq} \in FF_{jp} \\ \mathbf{1, if}\ FF_{jp} \in FF_{jq} \\ \mathbf{2, otherwise} \end{cases}$$

MF_r is a member of mating feature class, $MF_r \in \mathbf{MF}$, $MF_r \in FF_{jk}$.

JF_r is a member of joint feature class, $JF_r \in$ **JF,** $JF_r \in FF_{jk}$.

$:\rightarrow$ stands for a *belong-to* relation.

$\Leftrightarrow$ stands for an *inter-feature association* relation.

$\otimes$ stands for an *assembly/joining* relation.

3.6 Assembly Relation Model and Generic Assembly Relationship Diagram

There is a strong need for collaborative assembly design systems to communicate and exchange needed design data without transferring whole files from one design collaborator to another (FIPER 2001, Pegasus 2003). This selective lean information exchange is intended to overcome the bandwidth limitations on Internet/Intranet and to achieve secured relationships among participants. In order to ensure complete transfer of assembly model information during this selective transition, assembly engineering relations should be maintained.

In this work, a new Assembly Relation Model (ARM) including an Assembly Relationship Diagram, GARD is introduced to efficiently capture engineering relations among form features and parts for a collaborative design environment. Assembly engineering relations between features as well as between features and parts are defined as below:

Definition 1: *Belong-to* relations

A part P_j^i and a form feature FF_{jk} are said to have a *belong-to* relation,

$$B_{jk}^{(i)}: FF_{jk} :\rightarrow P_j^i,\ k = 1, 2, \ldots, n,$$

if $P_j^i \in \mathrm{A}_i, j = 1, 2, \ldots, m$; and $FF_{jk} \in P_j^i$.

In GARDs, a *belong-to* relation between P_j^i and FF_{jk} is represented by an arrow (see Figure 3-9-a). Parts are illustrated as dotted-line circle and form features as solid-line circle.

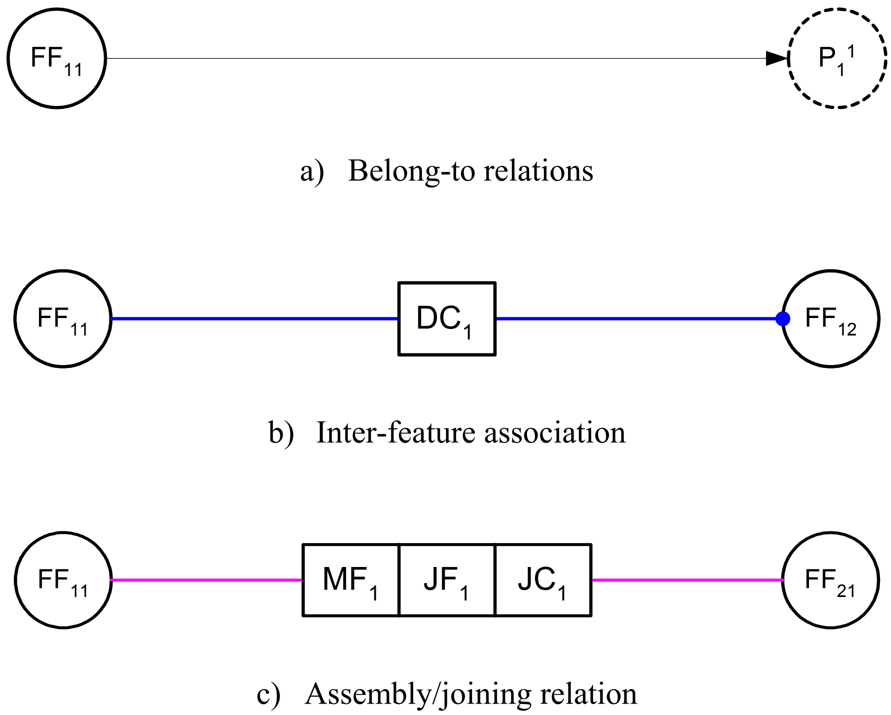


Figure 3-9 Relations in GARD

Definition 2: ***Inter-feature association*** **relations**

A form feature FF_{jp} and another form feature FF_{jq} are said to have an *inter-feature association* relation,

$$I_{pq}^{(j)}: FF_{jp} \Leftrightarrow FF_{jq},\ \ p = 1,2,\ \ldots,\ n,\ q = 1,2,\ \ldots,\ l,$$

if $P_j^i \in \mathrm{A}_i$, j = 1, 2, …, m; FF_{jp}, $FF_{jq} \in$ **FF;** FF_{jp}, $FF_{jq} :\rightarrow P_j^i$; DC_r and RC_{pq} are satisfied, where $r \in IDI_{pq}^{(j)}$; and $IDI_{pq}^{(j)}$ is an index set depending upon this pair, FF_{jp} and FF_{jq} .

The *inter-feature association* relation represents the relations between form features. The *relational constraint* (RC_{pq}) stands for the relationship between two form features in the form feature hierarchy. For example, a block (FF_{jq}) can have a blind hole (FF_{jp}) at a certain location. The distance between the coordinates of the block and the blind hole is a dimensional constraint. Since the block form feature contains the hole form feature (the block is a parent class of the hole), their *relational constraint* (RC_{pq}) is 0. Figure 3-9-b illustrates the *inter-feature association* relation in a GARD. The line with a square stands for the *inter-feature association* relation. The square represents a dimensional constraint. Here, the solid dot at the end of line stands for the relational constraint. The circle indicates a form feature associated to the *inter-feature association* relation. Figure 3-9-b illustrates a case of "FF_{11} and FF_{12} have an *inter-feature association* relation subjected to DC_1 and FF_{12} contains FF_{11} ($FF_{jq} \in FF_{jp}$)".

Definition 3: *Assembly/joining* relations

A form feature FF_{gp} and another form feature FF_{hq} are said to have an *assembly/joining* relation,

$$\vartheta_{pq}^{(gh)}: FF_{gp} \otimes FF_{hq},$$

if P_g^i and $P_h^i \in \mathbf{A}_i$, $g = 1, 2, \ldots, m_1$, $h = 1, 2, \ldots, m_2$; $FF_{gp} \in P_g^i$, $p = 1, 2, \ldots, l_1$; $FF_{hq} \in P_h^i$, $q = 1, 2, \ldots, l_2$; $FF_{gp}, FF_{hq} \in$ **FF;** $FF_{gp}, FF_{hq} \in \mathbf{J}$; $MF_{r1}, JF_{r2} \in \mathbf{J}$; and JC_{r3} is satisfied, where $r1 \in JMI_{pq}^{(gh)}$, $r2 \in JJI_{pq}^{(gh)}$, and $r3 \in JCI_{pq}^{(gh)}$; and $JMI_{pq}^{(gh)}$, $JJI_{pq}^{(gh)}$, and $JCI_{pq}^{(gh)}$ are index sets depending upon this pair, FF_{gp} and FF_{hq} .

The *assembly/joining* relations are represented by lines in a GARD. Figure 3-9-c illustrates an *assembly/joining* relation between FF_{11} and FF_{21}, that is "FF_{11} and FF_{21} are assembled subjected to MF_1, JF_1, and JC_1".

Definition 4: *Generic assembly relationship diagram*

Let d be a generic assembly relationship diagram of an assembly ($\mathbf{A}_i$) which has a set of parts P = $\{P_j^i\}$ for j = 1,2, …, m. Each part P_j^i has a set of form features FF = $\{FF_{jk}\}$ for k=1, 2, …, n_j. There exists a set of belong-to relations, $R^B = \{B_{jk}^{(i)}\}$, a set of inter-feature association relations of part j, $R^I = \{I_{pq}^{(j)}\}$ for p =1,2, …, l_1 and q = 1,2,…, l_2, and a set of assembly/joining relations of part g and part h, $R^\vartheta = \{\vartheta_{rs}^{(gh)}\}$ for g = 1, 2, …, m_1, h = 1,2, …, m_2, r =1,2, …, l_3, and s = 1,2,…, l_4. If E_{uv} denotes an edge between the nodes N_u and N_v of a diagram; then the assembly relationship diagram of $\mathbf{A}_i$ denoted by $d(\mathbf{A}_i)$ is defined by:

$$d(\mathbf{A}_i) = (V, E),$$

where

$V = \{V_1, V_2, \ldots, V_{m+N}\}$ is the set of nodes in $d(\mathbf{A}_i)$, where $N = \Sigma_j n_j$ and

$E = \{E_{uv}\}$ is the set of edges, where $u \in FFI_u$, $v \in FFI_v$; FFI_u and FFI_v are index sets depending upon the number of nodes;

such that:

There is a one-to-one correspondence between sets V and $P \cup FF$;

There is a one-to-one correspondence between sets E and $R^B \cup R^I \cup R^\vartheta$.

After assembly features are generated, intra-feature and inter-feature relationships are automatically captured in an AsD model. The AsD model contains an assembly relation model connected to a solid model. All geometric entities in an assembly relation model are linked to a related solid model. This ARM goes together with geometric data (solid model) in assembly data transitions, capturing assembly and joining information consistently in a collaborative assembly design environment. The ARM can be transformed into three representations (views) (i.e., symbolic, mathematical, and pictorial). The pictorial representation of the ARM is generated based upon the GARD. These three views serve as a communication media and help a designer's understanding.

Table 3-2 shows a symbolic representation of an AsD model (ARM) generated for a simple assembly in Figure 3-10. Here, P_1^1= pin_a; P_2^1 = plate_a; FF_{11} = cylinder_a; FF_{21} = block_b; FF_{22} = hole_b. Note that the designed d.o.f. in the assembly feature (AF) inferred as {fix}. This d.o.f. is inferred from the specified joining method, that is, gas metal arc welding (GMAW). Spatial relationship implication due to joining is discussed in the next section. From the AF, two MBs are generated for two aligned spatial relationships. Table 3-3 shows a mathematical assembly relation model and Figure 3-11 illustrates a pictorial representation of the ARM using GARD symbols. In Figure 3-11, DC1 (dimensional constraint) of the inter-feature relation is the location of the hole in the block. JC1 and JC2 are the welding condition and the fixture location respectively.

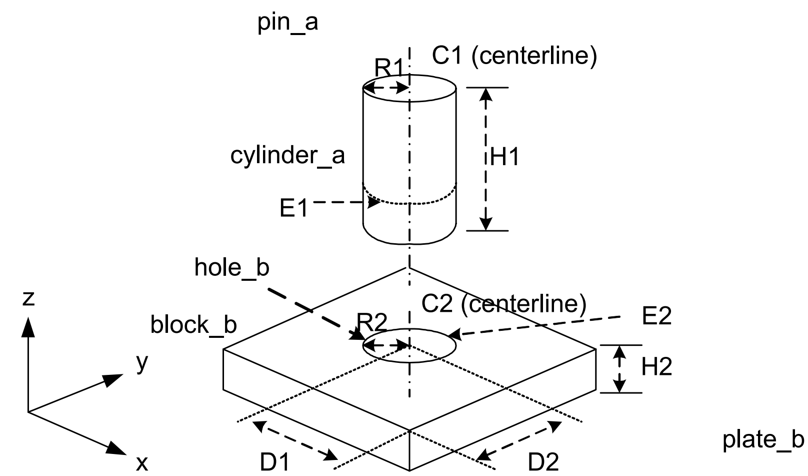


Figure 3-10 An assembly with a pin and a plate with hole

Table 3-2 Symbolic representation of the assembly relation model for Figure 3-10

Parts	Features and MB	Representation
P_1^1 & P_2^1 (pin_a & plate _b)	AF	▪ AF_1 = {mating features | mating bonds | joint features | [material] | [designed d.o.f.] | [implied constraints]} = { MF_1, MF_2 | MB_1, MB_2 | JF_1 | [Aluminum Alloy 6061 - T6] | {fix} | $\pm\Delta_1$}
	MF	▪ MF_1 = {S/R, [*mating components* (*mating entities*)]} = {aligned, [FF_{11} (C_1), FF_{22} (C_2)]} ▪ MF_2 = {aligned, [FF_{11} (E_1), FF_{22} (E_2)]}
	JF	▪ JF_1 = {*joining method* | *groove shape* | [*joining components* (*joining entities*)] | [joining constraint]} = {GMAW | single fillet | [FF_{11} (E_1), FF_{22} (E_2)] | [welding_condition], [fixture_location]}
	MB	▪ MB_1 = {mating pair ([mating features (form feature (inter-feature association, dimensional constraint))]) | mating conditions (assembly/joining relations (form features), S/R (transformed geometric constraints), d.o.f., [implied constraints])} = {MP_1 (MF_1, [C_1 (FF_{11} (I (.), {R_1, H_1})), C_2 (FF_{22} (I(ℓ_{12}^2, RC_{12} = 0), {R2, H2}, {D1, D2}))]) | MC_1 ($\vartheta_{12}^{(12)}$(FF_{11}, FF_{22}), aligned ({on_line}, {parallel}), {lin_z::rot_z}, [R_1<=R_2]))} ▪ MB_2 = {MP_2 (MF_2, [E_1 (FF_{11} (I (.), {R_1, H_1})), E_2 (FF_{22} (I(ℓ_{12}^2, RC_{12} = 0), {R2, H2}, {D1, D2}))]) | MC_2 ($\vartheta_{12}^{(12)}$(FF_{11}, FF_{22}), aligned ({on_line}, {parallel}), {rot_z}, [R_1<=R_2]))}

Table 3-3 Mathematical representation for Figure 3-10

Parts	Assembly engineering relationships
P_1^1 & P_2^1	$\{FF_{11} :\rightarrow P_1^1; FF_{21} :\rightarrow P_2^1; FF_{22} :\rightarrow P_2^1; FF_{21} \Leftrightarrow FF_{22} \mid RC_{12} = 0;\ FF_{11} \otimes FF_{22};$ $IDI_{12}^{(2)} = \{1\}, JMI_{12}^{(12)} = \{1, 2\}, JJI_{12}^{(12)} = \{1\}, JDI_{12}^{(12)} = \{0\}\}$

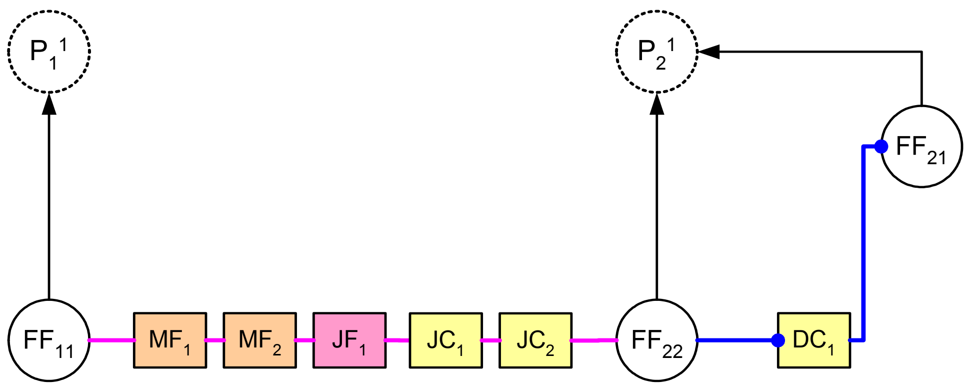


Figure 3-11 Pictorial representation for Figure 3-10

3.7 Assembly Design Engine

In this research, the AsD formalism for a collaborative assembly design environment is developed and implemented as a fundamental formalism for an AsD engine. The AsD engine generates an AsD model capturing assembly/joining relationships. The AsD model can be used for downstream assembly design activities, such as joining analysis. A designer can generate an assembly with the AsD engine by specifying spatial relationships, joining methods, weld seam/rivet locations, and joining constraints, such as welding conditions and fixture locations.

3.7.1 Demonstration of AsD tools

For this demonstration, the AsD engine is implemented using Microsoft's MFC, Spatial's ACIS, and Tech Soft's HOOPS. A connector assembly (Figure 3-12) is considered as a demonstration of the developed assembly design formalism. Figure 3-13 shows a graphic user interface of the AsD engine. Figure 3-14 shows a data structure of the assembly design formalism in terms of UML's static structure. Classes of the assembly design formalism can be found in Appendix A.

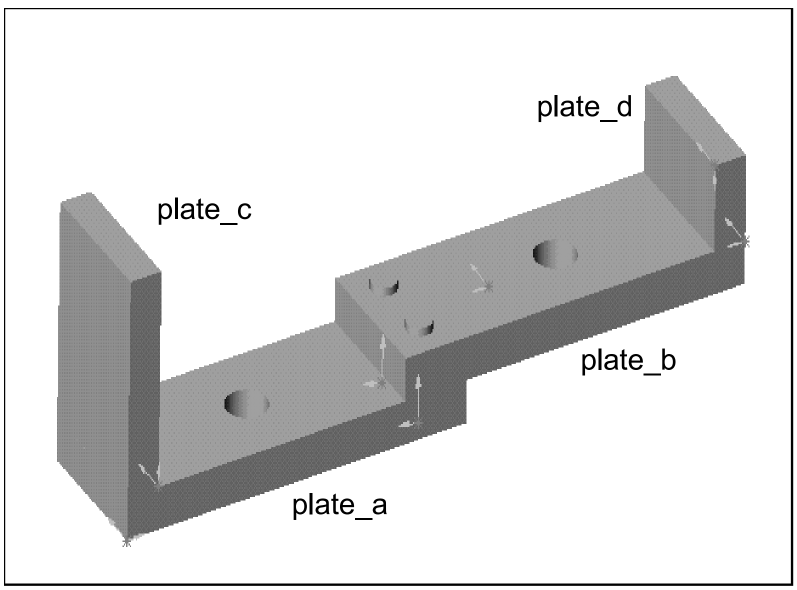


Figure 3-12 Connector assembly

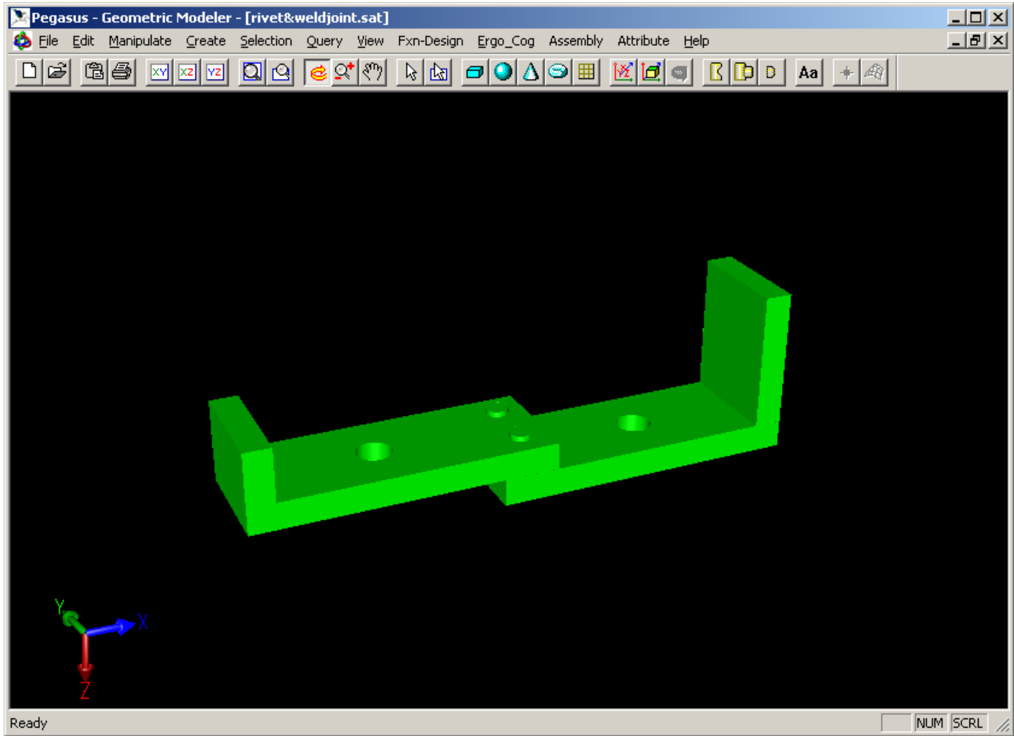


Figure 3-13 Graphic user interface of the AsD Engine

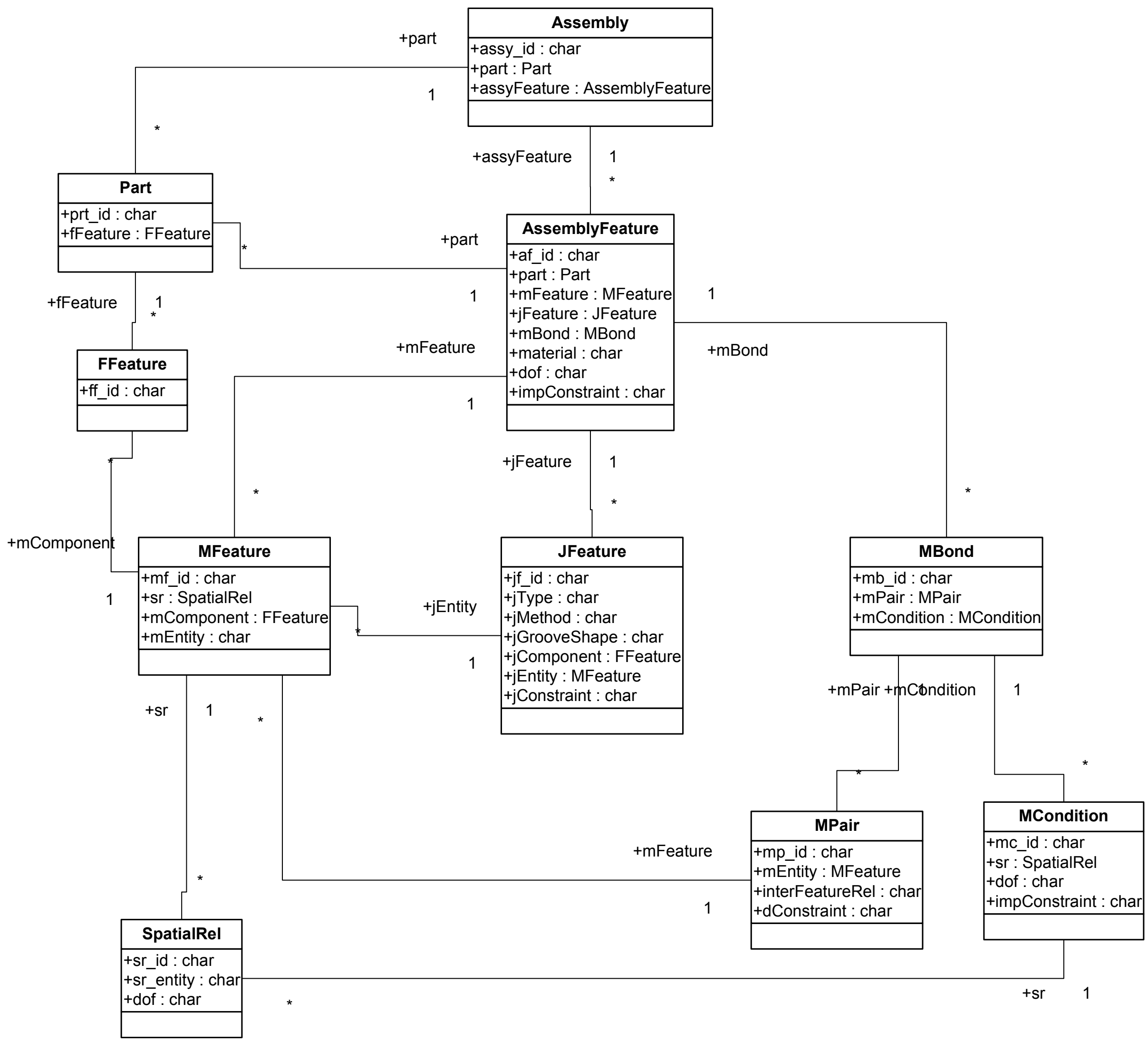


Figure 3-14 Data structure of the assembly design formalism

The following figures illustrate how a designer generates an assembly with joints and how the AsD engine generates the AsD model. Typical steps are listed as follows:

STEP 1: The designer specifies spatial relationships and corresponding mating features are extracted by the AsD engine (see Figure 3-15).

STEP 2: The designer determines a joining method and selected corresponding geometric entities on screen. The designer can provide joining conditions. Note that these joining conditions are essential information for succeeding assembly analyses (see Figures 3-16-a and 3-16-b).

STEP3: If the specified joining method satisfies the desired d.o.f., go to STEP4. Otherwise, go to STEP 2. This process is explained thoroughly in Chapter 4.

STEP 4: The AsD engine generates additional joint geometry, such as rivets or fasteners based upon joint conditions (see Figure 3-16-b). The designer can also determine materials for assembly components in this step.

STEP 5: Once the designer provides all information required to form assembly features, the AsD engine automatically generates an AsD model including assembly features and mating bonds.

STEP 6: For efficient assembly design data exchange, the XML data for the AsD model can be generated. This XML data is basic input used to generate the GARD.

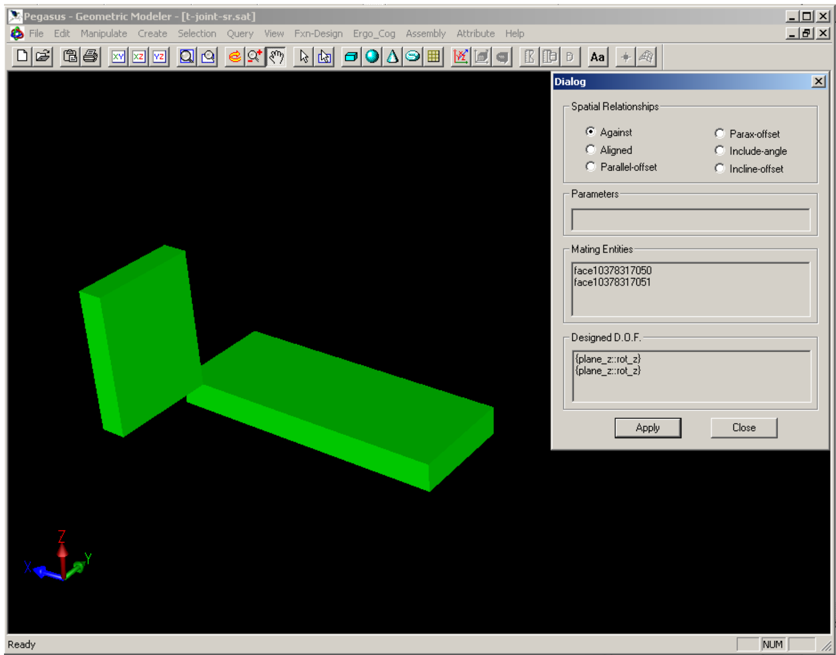
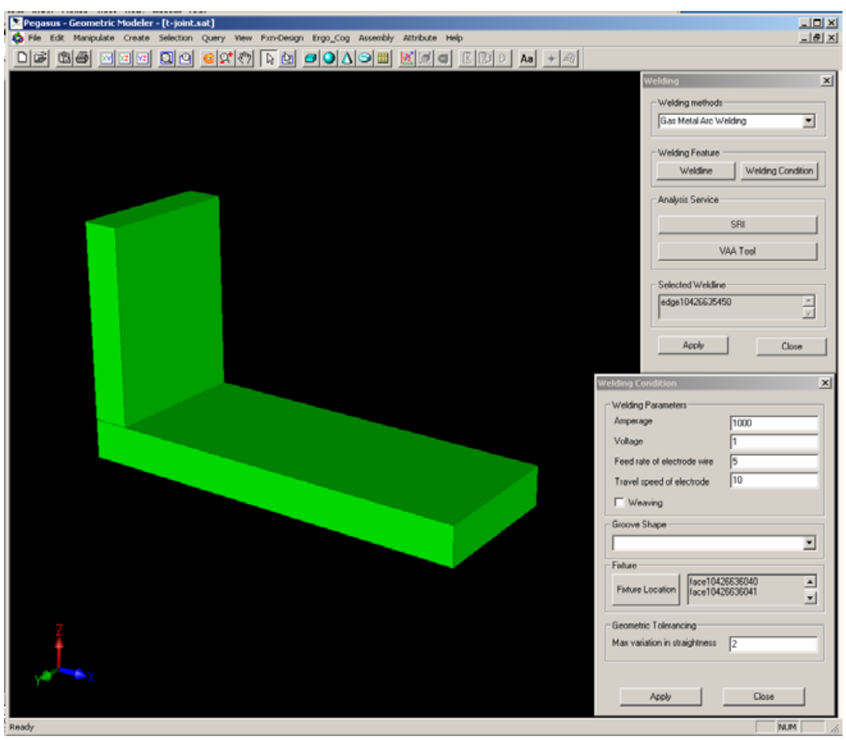
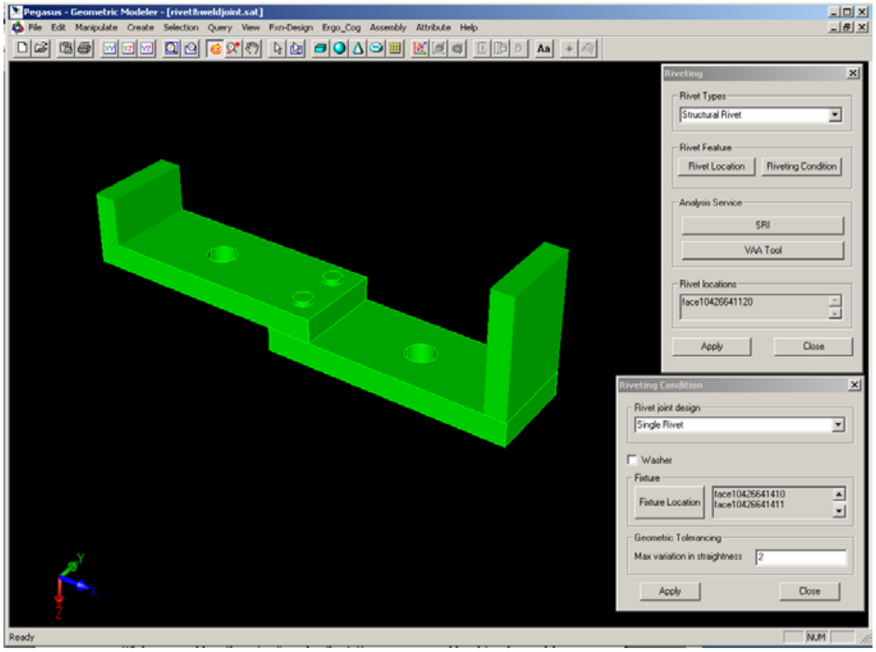


Figure 3-15 Spatial relationship specification and mating feature extraction



a) welding



b) riveting

Figure 3-16 Joining method specification and joint feature formation

All geometric entities specified in the XML data are linked to the solid model. In an ACIS solid model, attribute ID's are used as a linkage tag. This XML formatted AsD model goes together with the geometric data (solid model) in assembly data transitions. It allows assembly and joining information to be persistently captured in a collaborative assembly design environment.

While an assembly with joints is formed in the AsD engine, an ARM (AsD model) is generated internally. Table 3-4 shows the symbolic representation of the ARM of the connector (Figure 3-17). Table 3-5 shows the mathematical representation of the ARM of the connector (A_1). In this example, *plate_a* and *plate_b* are joined by two button rivets. *Top_surface* of *plate_a* and *bottom_surface* of *plate_b* have *against* relationships. The rivets are aligned along centerline of holes at the location that designer specified. *Plate_a* and *plate_c* are joined with gas metal arc welding (GMAW) and their mating feature entities are *top_surface* of *plate_a* and *bottom_surface* of *plate_c*. Similarly, *plate_b* and *plate_d* are joined by using GMAW. In this demonstration, the following notation is used:

e_i^{jk}: i^{th} edge of FF_{jk}

$P_1^1 = plate\_a$; $P_2^1 = plate\_b$; $P_3^1 = plate\_c$; $P_4^1 = plate\_d$;

FF_{11} = block (*length, width, height*) = block (L_{11}, W_{11}, H_{11}) = block (110, 40, 10);

FF_{21} = block (L_{21}, W_{21}, H_{21}) = block (110, 40, 10);

FF_{31} = block (L_{31}, W_{31}, H_{31}) = block (50, 40, 10);

FF_{41} = block (L_{41}, W_{41}, H_{41}) = block (20, 40, 10);

FF_{12} = hole (diameter, depth) = hole (DM_{12}, DT_{12}) = hole (12.81, 10);

FF_{22} = hole (DM_{22}, DT_{22}) = hole (12.81, 10);

JC_1(of FF_{11} and FF_{21}) = {location of rivets|tolerance} = {P_1^1(100, 10, .), P_1^1(100, 40, .)|$\pm\Delta_1$};

JC_2 (of FF_{11} and FF_{21}) = fixture locations;

JC_3 (of FF_{11} and FF_{31}) = welding condition;

JC_4 (of FF_{11} and FF_{31}) = fixture locations;

JC_5 (of FF_{11} and FF_{31}) = {datum planes | tolerance } = {*max_displacement* | $\pm\Delta_2$ };

JC_6 (of FF_{21} and FF_{41}) = welding condition;

JC_7 (of FF_{21} and FF_{41}) = fixture location;

JC_8 (of FF_{21} and FF_{41}) = { *max_displacement* | $\pm\Delta_3$ };

DC_1 (of FF_{11} and FF_{12}) = { location of hole | tolerance} = { P_1^1 (50, 20, .) | $\pm\Delta_4$ };

DC_2 (of FF_{21} and FF_{22}) = { P_2^1 (50, 20, .) | $\pm\Delta_5$ }.

Datum planes in JC_5 and JC_8 are reference planes that represent the designated tolerance limits of welding deformation. The datum planes are offset from the structure by the tolerance limits allowed.

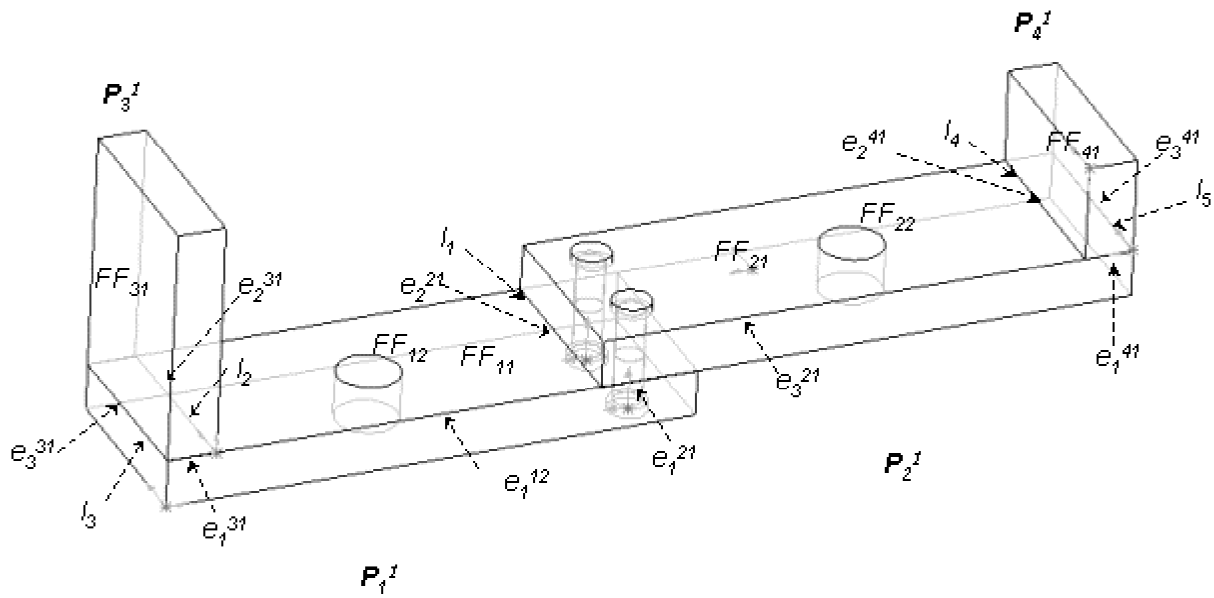


Figure 3-17 Connector assembly with two welded joints and one pin joint

Table 3-4 Symbolic representation for the connector in Figure 3-17

Parts	Features and MB	Representation
P_1^1 & P_2^1 (plate_a & plate _b)	AF	▪ AF_1 = {mating features | mating bonds | joint features | [material] | [designed d.o.f.] | [implied constraints]} = { MF_1, MF_2, MF_3| MB_1, MB_2, MB_3| JF_1 | [Aluminum Alloy 6061 - T6, Steel] | {fix} | [tolerance]}
	MF	▪ MF_1 = {S/R, [*mating components* (*mating entities*)]} = {against, [FF_{11} (top_surface), FF_{21} (bottom_surface)]} ▪ MF_2 = {aligned, [FF_{11} (l_1), FF_{21} (e_2^{21})]} ▪ MF_3 = {aligned, [FF_{11} (e_1^{11}), FF_{21} (e_1^{21})]}
	JF	▪ JF_1 = {*joining method* | *groove shape* | [*joining components* (*joining entities*)] | [joining constraint]} = {Button_Rivet | . | [FF_{11} (top_surface), FF_{21} (bottom_surface)] | [diameter_of_rivet, 5.66], [location_of_rivet, FF_{11} (100, 10, .), FF_{11} (100, 40, .)], [fixture_location]}
	MB	▪ MB_1 = {mating pair ([mating features (form feature (inter-feature association, dimensional constraint))]) | mating conditions (assembly/joining relations (form features), S/R (transformed geometric constraints), d.o.f., [implied constraints])} = {MP_1 (MF_1, [top_surface (FF_{11} (I (.), {L_{11}, W_{11}, H_{11}})), bottom_surface (FF_{21} (I (.), {L_{21}, W_{21}, H_{21}}))]) | MC_1 ($\vartheta_{11}^{(12)}$(FF_{11}, FF_{21}), against ({on_surface}), {plane_z::rot_z}, [.]))} ▪ MB_2 = {MP_2 (MF_2, [l_1 (FF_{11} (I (.), {L_{11}, W_{11}, H_{11}})), e_2^{21} (FF_{21} (I (.), {L_{21}, W_{21}, H_{21}}))]) | MC_2 ($\vartheta_{11}^{(12)}$(FF_{11}, FF_{21}), aligned ({on_line} & {parallel}), {lin_l_1::lin_l_1}, [.]))} ▪ MB_3 = {MP_3 (MF_2, [e_1^{11} (FF_{11} (I (.), {L_{11}, W_{11}, H_{11}})), e_1^{21} (FF_{21} (I (.), {L_{21}, W_{21}, H_{21}}))]) | MC_3 ($\vartheta_{11}^{(12)}$(FF_{11}, FF_{21}), aligned ({on_line} & {parallel}), {lin_e_1^{11}::lin_e_1^{11}}, [.]))}
P_1^1 & P_3^1 (plate_a & plate _c)	AF	▪ AF_2 = {mating features | mating bonds | joint features | [material] | [designed d.o.f.] | [implied constraints]} = {MF_4, MF_5, MF_6| MB_4, MB_5, MB_6| JF_2 | [Aluminum Alloy 6061 - T6] | {fix} | [tolerance]}
	MF	▪ MF_4 = {S/R, [*mating components* (*mating entities*)]} = {against, [FF_{11} (top_surface), FF_{31} (bottom_surface)]} ▪ MF_5 = {aligned, [FF_{11} (l_2), FF_{31} (e_2^{31})]} ▪ MF_6 = {aligned, [FF_{11} (e_1^{12}), FF_{31} (e_1^{31})]}
	JF	▪ JF_2 = {*joining method* | *groove shape* | [*joining components* (*joining entities*)] | [joining constraint]} = {GMAW | single fillet | [FF_{11} (l_2), FF_{31} (e_2^{31})] | [welding_condition], [fixture_location]}

Table 3-4 (continued)

<table>
<tr><td>P_1^1 & P_3^1
(plate_a & plate _c)</td><td>MB</td><td>▪ MB_4 = {mating pair ([mating features (form feature (inter-feature association, dimensional constraint))]) | mating conditions (assembly/joining relations (form features), S/R (transformed geometric constraints), d.o.f., [implied constraints])}
= {MP_4
(MF_4, [top_surface (FF_{11} (I (.), {L_{11}, W_{11}, H_{11}})),
bottom_surface (FF_{31} (I (.), {L_{31}, W_{31}, H_{31}}))]) |
MC_4
($\vartheta_{11}^{(13)}$(FF_{11}, FF_{31}), against ({on_surface}), {plane_z::rot_z}, [.]))}
▪ MB_5
= {MP_5
(MF_5, [l_2 (FF_{11} (I (.), {L_{11}, W_{11}, H_{11}})),
e_2^{31} (FF_{31} (I (.), {L_{31}, W_{31}, H_{31}}))]) |
MC_5
($\vartheta_{11}^{(13)}$(FF_{11}, FF_{31}), aligned ({on_line} & {parallel}), {lin_l_2::lin_l_2}, [.]))}

▪ MB_6
= {MP_6
(MF_6, [e_1^{12} (FF_{11} (I (.), {L_{11}, W_{11}, H_{11}})),
e_1^{31} (FF_{31} (I (.), {L_{31}, W_{31}, H_{31}}))]) |
MC_6
($\vartheta_{11}^{(13)}$(FF_{11}, FF_{31}), aligned ({on_line} & {parallel}), {lin_e_1^{12}::lin_e_1^{12}}, [.]))}</td></tr>
<tr><td rowspan="4">P_2^1 & P_4^1
(plate_b & plate _d)</td><td>AF</td><td>▪ AF_3 = {mating features | mating bonds | joint features | [material] | [designed d.o.f.] | [implied constraints]}
= {MF_7, MF_8, MF_9| MB_7, MB_8, MB_9| JF_3 |
[Aluminum Alloy 6061 - T6]| {fix} |
[tolerance]}</td></tr>
<tr><td>MF</td><td>▪ MF_7= {S/R, [mating components (mating entities)]}
= {against, [FF_{21} (top_surface), FF_{41} (bottom_surface)]}
▪ MF_8 = {aligned, [FF_{21} (l_4), FF_{41} (e_2^{41})]}
▪ MF_9 = {aligned, [FF_{21} (e_3^{21}), FF_{41} (e_1^{41})]}</td></tr>
<tr><td>JF</td><td>▪ JF_3 = {joining method | groove shape | [joining components (joining entities)] | [joining constraint]}
= {GMAW | single fillet | [FF_{21} (l_4), FF_{41} (e_2^{41})] | [welding_condition], [fixture location]}</td></tr>
<tr><td>MB</td><td>▪ MB_7 = {mating pair ([mating features (form feature (inter-feature association, dimensional constraint))]) | mating conditions (assembly/joining relations (form features), S/R (transformed geometric constraints), d.o.f., [implied constraints])}
= {MP_7
(MF_7, [top_surface (FF_{21} (I (.), {L_{21}, W_{21}, H_{21}})),
bottom_surface (FF_{41} (I (.), {L_{41}, W_{41}, H_{41}}))]) |
MC_7
($\vartheta_{11}^{(24)}$(FF_{21}, FF_{41}), against ({on_surface}), {plane_z::rot_z}, [.]))}
▪ MB_8
= {MP_8
(MF_8, [l_4 (FF_{21} (I (.), {L_{21}, W_{21}, H_{21}})),
e_2^{41} (FF_{41} (I (.), {L_{41}, W_{41}, H_{41}}))]) |
MC_8
($\vartheta_{11}^{(24)}$(FF_{21}, FF_{41}), aligned ({on_line} & {parallel}), {lin_l_4::lin_l_4}, [.]))}

▪ MB_9
= {MP_9
(MF_9, [e_3^{21} (FF_{21} (I (.), {L_{21}, W_{21}, H_{21}})),
e_1^{41} (FF_{41} (I (.), {L_{41}, W_{41}, H_{41}}))]) |
MC_9
($\vartheta_{11}^{(24)}$(FF_{21}, FF_{41}), aligned ({on_line} & {parallel}), {lin_e_3^{21}::lin_e_3^{21}}, [.]))}</td></tr>
</table>

Table 3-5 Mathematical representation for the connector in Figure 3-17

Parts	Assembly relationships
P_1^1 & P_2^1	$\{FF_{11} :\rightarrow P_1^1; FF_{12} :\rightarrow P_1^1; FF_{21} :\rightarrow P_2^1; FF_{22} :\rightarrow P_2^1; FF_{11} \Leftrightarrow FF_{12} \mid RC_{12} = 0;\ FF_{21} \Leftrightarrow FF_{22} \mid RC_{12} = 0; FF_{11} \otimes FF_{21}; IDI_{12}^{(1)} = \{1\}, IDI_{12}^{(2)} = \{2\}, JMI_{11}^{(12)} = \{1, 2, 3\}, JJI_{11}^{(12)} = \{1\}, JCI_{11}^{(12)} = \{1, 2\}\}$
P_1^1 & P_3^1	$\{FF_{11} :\rightarrow P_1^1; FF_{12} :\rightarrow P_1^1; FF_{31} :\rightarrow P_3^1; FF_{11} \Leftrightarrow FF_{12} \mid RC_{12} = 0; FF_{11} \otimes FF_{31}; IDI_{12}^{(1)} = \{4\}, JMI_{11}^{(13)} = \{4, 5, 6\}, JJI_{11}^{(13)} = \{2\}, JCI_{11}^{(13)} = \{3, 4, 5\}\}$
P_2^1 & P_4^1	$\{FF_{21} :\rightarrow P_2^1; FF_{22} :\rightarrow P_2^1; FF_{41} :\rightarrow P_4^1; FF_{21} \Leftrightarrow FF_{22} \mid RC_{12} = 0; FF_{21} \otimes FF_{41}; IDI_{12}^{(2)} = \{5\}, JMI_{11}^{(24)} = \{7, 8, 9\}, JJI_{11}^{(24)} = \{3\}, JCI_{11}^{(24)} = \{6, 7, 8\}\}$

3.7.2 XML AsD Format and GARD tool

In this research, the AsD model generated by the developed AsD formalism is represented in a XML format to exchange/share AsD information in software and hardware in an independent way. XML stands for E**X**tensible **M**arkup **L**anguage. Tags enclosed in "<" and ">" characters are used to define the structure and data elements of an XML text or string. These tags are not predefined in XML. Hence, one is required to define custom tags for new implementations. XML uses a Document Type Definition (DTD) or a Schema to describe the data. A DTD or Schema is designed to be self-descriptive.

The primary and sole purpose of XML is to carry data. XML was designed to describe data and to focus on what data is. It is created to structure, store, and to exchange information. It is a cross-platform, software and hardware independent tool for transmitting information. This makes it particularly applicable to represent AsD data that may be exchanged between different CAD platforms and systems.

With XML, information/data can be stored in separated XML files and exchanged as text between incompatible systems. Since XML data is stored in plain text format, it provides a software and hardware independent way of sharing data. This makes it much easier to create data that different applications can work with. It also makes it easier to expand or upgrade a system to new operating systems, servers, applications, and new browsers. In the CAD industry, designer packages contain data in incompatible formats. One of the most time-consuming challenges for developers has been to exchange data between such systems. The use of an XML data format in AsD can greatly reduce this complexity and create data that can be read by many different types of applications. Hence, it helps to overcome inter-operability problems associated with traditional CAD systems.

Plain text files can be used to store XML formatted AsD information in databases and also be used in a collaborative design environment where data is transmitted to distributed design participants at remote locations.

3.7.2.1 XML Syntax. The syntax rules of XML are very simple and very strict. XML documents use a self-describing and simple syntax. The first line in the document - the XML declaration - defines the XML version and the character encoding used in the document. In the above example on attribute representation, the document conforms to the 1.0 specification of XML and uses the ISO-8859-1 (Latin-1/West European) character set.

The following example is a simple XML description of an assembly feature.

```
<?xml version="1.0" encoding = “ISO-8859-1”?>
<!-- AsD XML Description By Assembly Design Formalism ! -->
<assembly-design>
        <info> </info>
        <assembly-feature>  </assembly-feature>
</assembly-design>
```

The first tag in an XML document is the root tag. In the above example, the next line describes the root element of the document (like it was saying: "this document is an assembly design"): <assembly-design>. All XML documents must contain a single tag pair to define the root element. All other elements must be nested within the root element. All elements can have sub elements (children). Sub elements must be correctly nested within their parent element. The next two lines describe two child elements of the root (information and assembly feature). In XML, all elements must have a closing tag. In the example, the last line defines the end of the root element: </assembly-design>

3.7.2.2 XML AsD Data Format. The XML schema for the AsD model is listed below: A brief description of each tag is given below.

```
<?xml version="1.0" ?>
<!-- AsD XML Description By Assembly Design Formalism ! -->
<ASD>
      <info> </info>
      <AF>
            <name> </name>
            <MF> </MF>
            <JF> </JF>
            <MB> </MB>
            <Material> </Material>
      </AF>
</ASD>
```

The XML declaration

The first line in the document (<?xml version= "1.0" ?>) is the XML declaration. It defines the XML version used in the document. In the AsD model above, the document conforms to the 1.0 specification of XML. The statement enclosed within “<!--“ and “!-->” are comments.

The root tag

The first tag (<ASD>) is the root tag. It describes the root element of the document (like it was saying: "this document is a assembly design model"). It begins the definition of an instance of the AsD model in XML. The last line defines the end of the root element: </ASD>. It marks the end of the XML data of the AsD model. All the other information concerning the AsD must be enclosed within the opening and closing tags.

The info tag

This tag (<info> </info>) contains the general information about the AsD model. This information consists of: name, unit, and description of the AsD model. The schema for this information is shown below.

AsD information XML Schema:

```
<info>
        <name> </name>
        <unit> </unit>
        <description>  </description>
</info>
```

The AF tag

This assembly feature tag (<AF> </AF>) contains the core information about the AsD model. This tag includes five child elements; those are name, mating feature (<MF>), joint feature (<JF>), mating bond (<MB>), and material (<Material>).

The MF tag

This tag (<MF> </MF>) contains the information related to the mating feature extraction. This tag consists of: id, spatial relationship, mating components, and mating entities. The schema for this information is shown below.

AsD mating feature XML Schema:

```
<MF>
        <MF-ID> </MF-ID>
        <SR> </SR>
        <mating-component> </mating-component>
        <mating-entity> </mating-entity>
</MF>
```

The JF tag

This tag (<JF> </JF>) contains the information related to the joint feature formation. This tag consists of: id, joining method, joining components and entities, joining conditions, and tolerance child elements. Joining conditions of different joining methods vary and the schema of each condition should defined considering the characteristics of joining methods. The schema for arc welding and riveting are shown below.

AsD joint feature XML Schema of welding:

```
<JF>
        <JF-ID> </JF-ID>
        <joining-method>  </joining-method>
        <joining-component> </joining-component>
                <joining-entity> </joining-entity>
        <groove-shape>  </groove-shape>
        <joining-constraint>
                <welding-condition>
                                <amperage>  </amperage>
                                <voltage> </voltage>
                                <feedrate> </feedrate>
                                <weld-speed> </weld-speed>
                                <weaving> </weaving>
                </welding-condition>
                <fixture-location>
                                <id> </id>
                </fixture-location>
         </joining-constraint>
         <tolerance>
                 <max-var-straightness> </max-var-straightness>
          </tolerance>
</JF>
```

AsD joint feature XML Schema of riveting:

```
<JF>
        <JF-ID> </JF-ID>
        <joining-method>  </joining-method>
        <joining-component> </joining-component>
                <joining-entity> </joining-entity>
        <groove-shape>  </groove-shape>
```

```
        <joining-constraint>
                <riveting-condition>
                        <washer>  </washer>
                </riveting-condition>
                <fixture-location>
                        <id> </id>
                </fixture-location>
        </joining-constraint>
        <tolerance>
                <max-var-straightness> </max-var-straightness>
        </tolerance>
</JF>
```

The MB tag

This tag (<MB> </MB>) contains the information of a mating bond, and consists of: id, mating pair and mating condition. Mating pair tag (<mating-pair> </mating-pair>) contains two associated mating feature child elements (<mating-feature> </mating-feature>). Mating condition tag (<mating-condition> </mating-condition>) contains five child elements; those are assembly joining relation (<assembly-joining-relation> </assembly-joining-relation>), spatial relationship (<SR> </SR>), transformed geometric constraint (<transformed-geometric-constraint> </transformed-geometric-constraint>), degree of freedom (<DOF> </DOF>), and implied constraints (<implied-constraint> </implied-constraint>). The schema of this information are shown below.

AsD mating bond XML Schema:

```
<MB>
        <MB-ID> </MB-ID>
        <mating-pair>
                <mating-feature>
                        <ID> </ID>
                        <form-feature>
                                <ID> </ID>
                                <inter-feature-association></inter-feature-association>
                                <dimensional-constraint> </dimensional-constraint>
                        </form-feature>
```

```
            </mating-feature>
           <mating-feature>
                  <ID> </ID>
                  <form-feature>
                       <ID> </ID>
                       <inter-feature-association></inter-feature-association>
                       <dimensional-constraint> </dimensional-constraint>
                  </form-feature>
            </mating-feature>
      </mating-pair>
      <mating-condition>
           <assembly-joining-relation>
                    <form-feature> </form-feature>
                    <form-feature> </form-feature>
           </assembly-joining-relations>
           <SR> </SR>
          <transformed-geometric-constraint>
                  </transformed-geometric-constraint>
          <dof> </dof>
          <implied-constraint>  </implied-constraint>
      </mating-condition>
</MB>
```

The Material tag

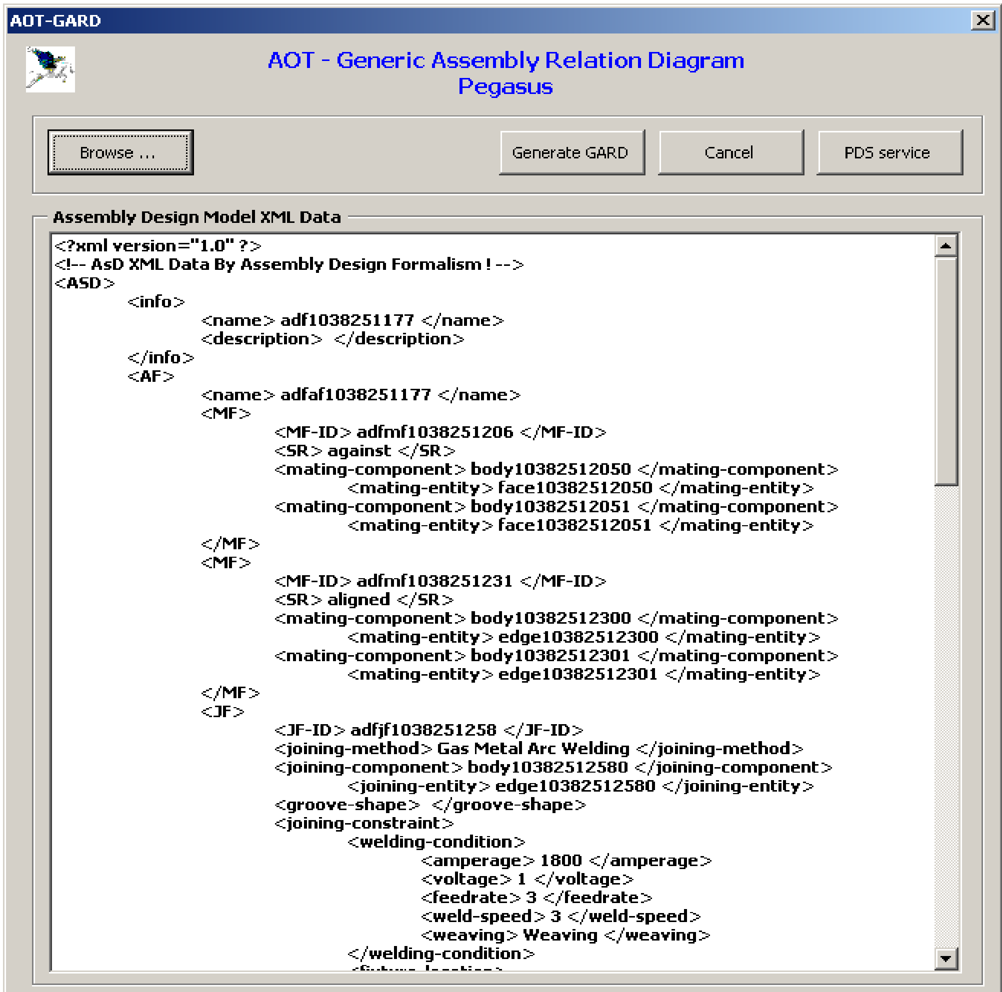
This material tag (<Material> </Material>) contains information regarding the materials assigned to each part. This tag consists of: part id and material name. The schema for this information is shown below.

AsD material XML Schema:

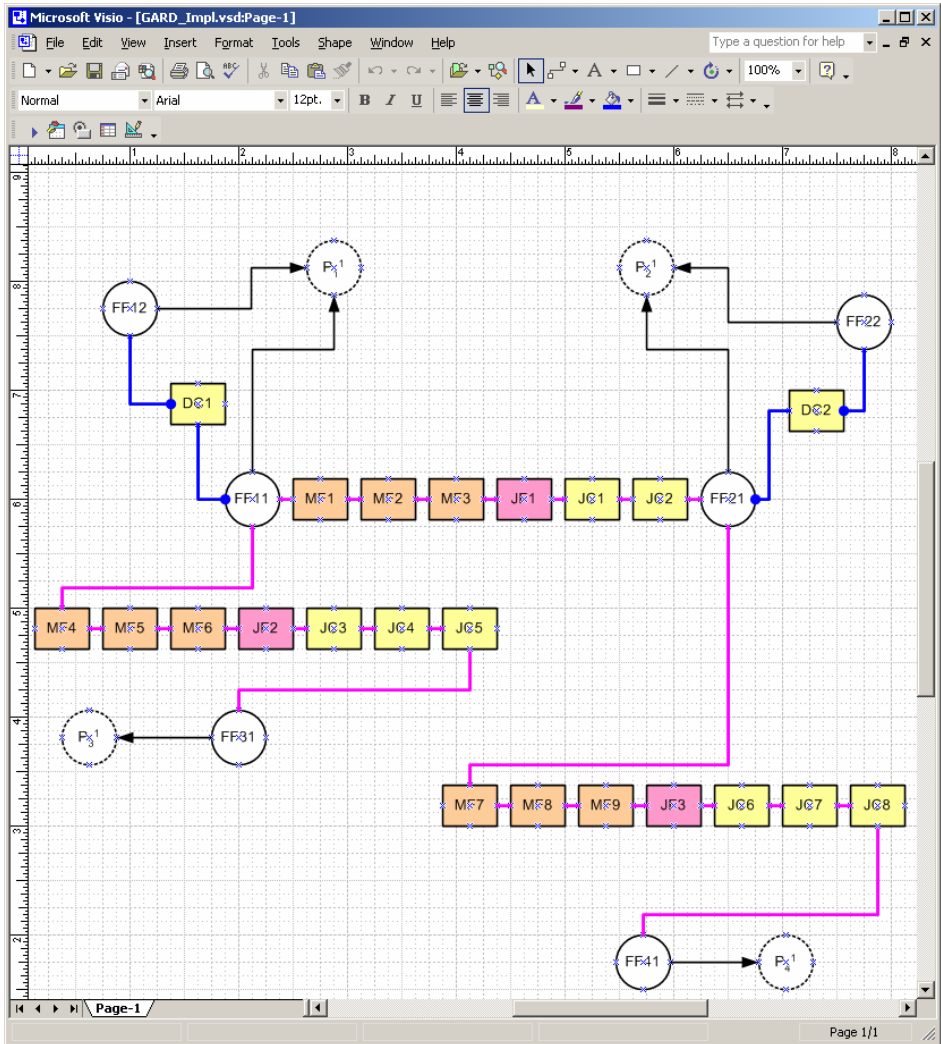
```
<Material>
      <Part-ID>  </Part-ID>
      <Name> </Name>
</Material>
```

3.7.2.3 GARD Tool. Based upon the AsD model in a XML format, the corresponding GARD can be generated with the aid of a developed GARD tool to help the designer to easily understand the assembly and joining relationships in an assembly. The GARD tool is implemented using Microsoft's Visio and Visual Basic for Applications (VBA). An XML Parser is used to read an

XML document. Loading an XML file into the parser extracts the data embedded in the XML file. A function (code) written in VBA is used to accomplish the parser function.



a) Initial interface



b) GARD for the connector assembly

Figure 3-18 GARD tool

Figure 3-18-a shows an initial window of the GARD tool. The designer can open the AsD model with this tool and generate a GARD corresponding to the AsD model. Figure 3-18-b shows a GARD for the connector assembly. By clicking each diagram's entities, relevant assembly/joining information and design model can be retrieved locally or remotely through the service-oriented architecture. This GARD tool provides a very useful communication media for a service-oriented collaborative assembly design.

3.7.3 AsD Formalism and AsD Tools and a Service-Oriented Collaborative Assembly Design

The next paragraphs explain how the developed AsD formalism and the AsD tools can be integrated in a service-oriented collaborative assembly design environment. A typical scenario is when a system integrator, such as an auto manufacturer, out-sources the design and manufacturing of sub-systems, such as car frames from different vendors. If the auto manufacturer wants to design a complete car frame with sub-frames designed by vendors A and B, the vendors provide an XML-formatted AsD model, which are simple ASCII files, of the sub-frames to the auto-manufacturer instead of sending the entire CAD model. The system integrator can easily generate GARDs from the AsD model. The GARDs are linked to the corresponding design models of sub-system components. These design models in a certain proprietary CAD format are translated into a CAD kernel format, such as SAT of ACIS, as soon as the vendors send the AsD model to the system integrator. The design models in the CAD kernel format can be provided to the system integrator when the system integrator indicates a request for viewing a specific component from the GARD tool. The auto-manufacturer therefore can decide the assembly components to be joined. The determined assembly components are loaded into the AsD engine and the system integrator can specify a joining method between assembly components. The new AsD model and its corresponding GARD are generated automatically based upon the AsD formalism.

During collaborative assembly design, assembly design participants typically use different CAD systems. To generate a complete assembly, each design model needs to be translated into a single CAD format, which can be accomplished by using specialized translators. However, this often causes problems in the numerical accuracy of geometric model, since different CAD systems employ different methods of CAD model generation. The first solution of this is to provide a modeler that has the ability to process models created in different CAD formats. The second solution is to use solid modeling kernels. Typically, a CAD system is built on a solid modeling kernel. Based upon the solid modeling kernel, suitable interfaces and high-level operations, such as feature based modeling and editing, of the CAD system are implemented. It is to be noted that while there are a large number of proprietary CAD formats, there are relatively few solid modeling kernels that are available, such as ACIS and Parasolid. This AsD engine utilizes the second method and it is implemented with the ACIS kernel.

Figure 3-19 shows how assembly design collaborators (*e*-designers) can share AsD model interacting with different CAD systems in the service-oriented collaborative assembly design environment. Consider a system integrator, such as an auto manufacturer, who wants to assemble two components designed by vendor 1 and vendor 2. Through the Pegasus architecture, AsD models of assembly components can be provided remotely to the system integrator and the system integrator can generate an assembly. In case the vendors' CAD systems provide different CAD kernels, a Pegasus multi-kernel agent manages to maintain consistent kernel format. Detailed processes are described below. The numbers in Figure 3-19 stand for the index of each process.

1. A system integrator requests AsD models of sub-assemblies/components interested (①).

2. Vendors provide requested AsD models in XML format to the system integrator, while the corresponding CAD models are translated to the CAD kernel model and stored in the local database of each vendor (①). If the vendor doesn't have the capability to translate the CAD model to the kernel model, a third-party multi-kernel agent can be employed (②).

3. The system integrator reviews sub-assemblies/components with the aid of the GARD tool and a product viewer (③, ④, and ⑥). According to the system integrators' needs, the GARD tool can selectively retrieve necessary parts in kernel format from the vendors' database (⑤).

4. After determining which sub-assemblies/components are to be joined, the system integrator can load kernel models of selected individual parts into the AsD engine and specify joining methods between the parts (⑦).

5. An AsD model for the new assembly is generated based upon the AsD Formalism (⑦). The new AsD model can be sent to the vendors to share assembly information (⑧).

6. When the system integrator needs to know additional assembly design information, such as the physical effects of joining, the system integrator can request relevant service using AsD models (⑨).

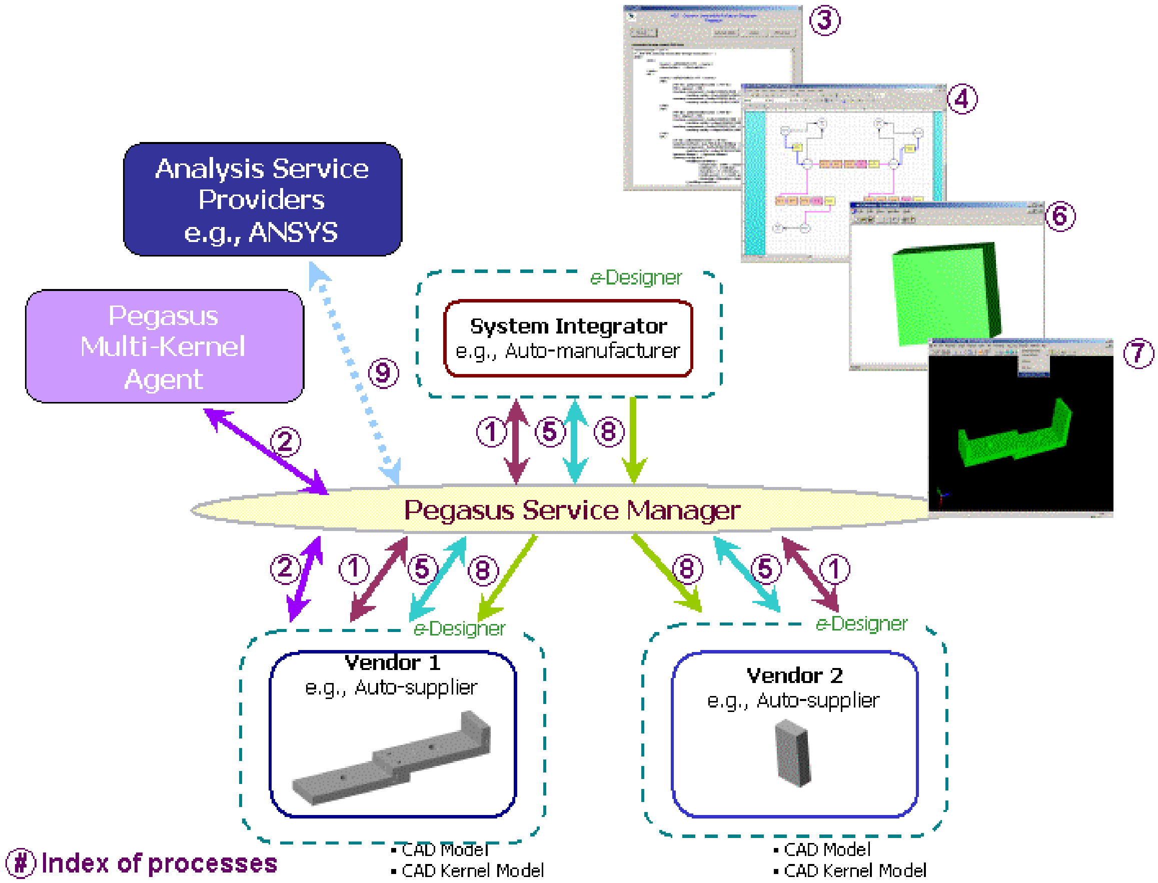


Figure 3-19 AsD tools in a service-oriented collaborative assembly design

3.8 Summary

Joints in product design are common because of the limitations of component geometric configurations and material properties, and the requirements of inspection, accessibility, repair, and portability. Collaborative product design is emerging as a viable alternative to the traditional design process. Collaborative assembly design methodologies are needed for distributed product development. Existing assembly design methodologies have limitations on capturing the non-geometric aspects of a designer's intent on joining and are not efficient for a collaborative design environment. This work introduces an AsD formalism and associated AsD tools to capture joining relations. This AsD formalism allows the joining relations to be modeled symbolically for computer interpretation, and the model can be used for inferring mathematical and physical implications. An AsD model generated from the AsD formalism is used to exchange assembly design information transparently in a collaborative assembly design environment. ARM and GARD capture assembly and joining information concisely and persistently. As a demonstration, the developed AsD formalism and AsD tools are applied on a connector assembly with arc weld and rivet joints. The contributions of this work are summarized as follows:

Contributions

1. The developed AsD formalism specifies the assembly and joining relations symbolically to support collaborative assembly design. By using the AsD formalism, assembly and joining relations are extracted from the AsD model and ARM has mathematically solvable implications.
2. A spatial relationship kernel preserves design intent on the assembly. The spatial relationship implication is inferred to validate the specified joining method that satisfies the designer's intent.
3. The AsD model supplements geometric and topological information of nominal geometry with assembly/joining information, which is essential for various assembly design activities, such as joining analysis, process planning, and integrated simulation.
4. The ARM has three views (i.e., symbolic, mathematical, and pictorial). The pictorial view, GARD, serves as a media to exchange assembly design and joining information concisely, persistently, and in a user-friendly manner in a collaborative design environment.

5. AsD tools, including the AsD engine and GARD tool, are developed to implement the AsD formalism, which leverage an efficient assembly data sharing mechanism and transparent assembly information flow for a collaborative assembly design environment. The GARD tool interprets the symbolic representation of ARM and generates relevant pictorial representations in the format of GARD.

4.0 ASSEMBLY IMPLICATION ENGINE

The Assembly Implication (AsI) engine extracts various assembly implication information, that is, spatial relationship (S/R) implications and physical effect from the assembly model. This implication information is essential for the designer to make an appropriate decision on joining methods under geometric constraints. The AsI engine consists of two tools, a Spatial Relationship Implication (SRI) tool and a Virtual Assembly Analysis (VAA) tool.

4.1 Spatial Relationship Implication (SRI) Tool

Spatial relationships are specified /imposed during the assembly design process. As described in the previous sections, each spatial relationship can be interpreted as a constraint imposed on the d.o.f. between relative mating or interacting features. Given a set of spatial relationships, the resultant d.o.f. can be inferred. In other words, any allowable motion for parts has to follow a path along the directions specified by the d.o.f. in order to maintain their spatial relationships.

In assembly design, spatial relationships can be assigned to achieve intended d.o.f. These desired spatial relationships are realized and maintained (or enforced) in the physical assembly by joining. Figure 4-1 illustrates how spatial relationships implied by joint design can be used for a designer’s intent analysis. As shown in the figure, each joining method infers specific spatial relationships and the corresponding d.o.f. are implied by these spatial relationships. The designer's original intent imposed on assembly design can be analyzed by comparing the implied d.o.f. and the designed d.o.f. For example, a designer wants to permanently join two plates (Figure 4-2) and he/she assigns spatial relationships to fix those plates (Table 4-1). If the designer considers a welded joint and specifies a welding operation as a joining method, then the d.o.f. corresponding to the welding operation can be inferred and used to check whether this welding operation will satisfy the designer's intent on the assembly. The welding operation causes 1) an *against* spatial relationship between the mating faces, 2) an *aligned* spatial relationship between joining entities on the weld seam, and 3) the two assembly components (two plates) to loose all d.o.f. and become fixed. In this case, the specified joining method (welding) fully satisfies the designed d.o.f.

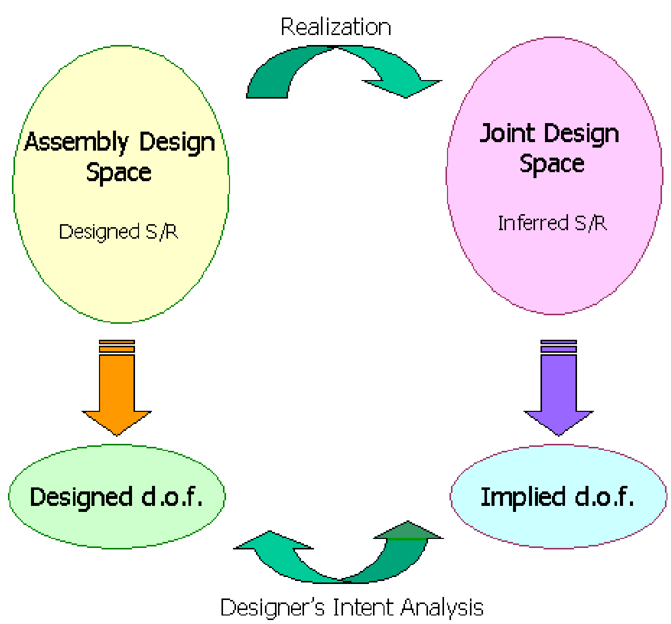


Figure 4-1 Spatial relationship implication

In other cases, some joining methods may either under-constrain or over-constrain the d.o.f. on an assembly. As an illustration, consider the case shown in Figure 4-2 with the corresponding designed S/R in Table 4-1. The two plates are intended to be joined and their d.o.f. are fixed by assigning a series of spatial relationships. As shown in Table 4-2, if a designer wants to join the two plates by applying one cylindrical rivet at p_1, the intended d.o.f. (*fixed*) is under-constrained. In a riveting operation, the end of the rivet shank is deformed after upsetting (Figure 4-3). However, after upsetting, the assembly can still have rotational d.o.f., if there is enough tangential (rotational) force applied to the two plates. Table 4-3 shows the d.o.f. implication rule when one cylindrical rivet joint is used. When two rivets are used to join the assembly components, the d.o.f. of components are fully constrained based on the reduction rule in Table 4-4. Note that more than two rivets can increase structural rigidity, even though d.o.f of the assembly are over-constrained and joining cost and time are increased. The proper number of rivets can be determined by assembly operation analysis.

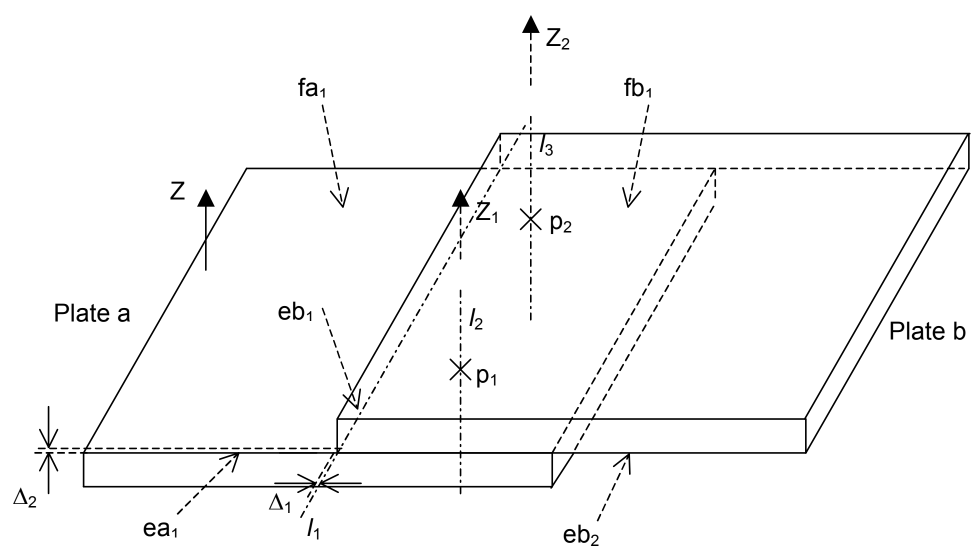


Figure 4-2 Lap joint with spatial relationships

Table 4-1 Designed spatial relationships of Figure 4-2

Plate a	Plate b	Spatial relationship	Designed d.o.f.
fa_1 l_1 ea_1	fb_1 eb_1 eb_2	against aligned aligned	{plane_z::rot_z} {lin_l_1::lin_l_1} {fixed}

Table 4-2 Spatial relationship implication of joining methods

Joining Method	Plate a	Plate b	Inferred spatial relationship	Implied d.o.f.
Welding	fa_1 l_1	fb_1 eb_1	against aligned (weld)	{plane_z::rot_z} {fixed}
One rivet (Rivet Q_1 at P_1)	fa_1 l_2	fb_1 l_2	against aligned	{plane_z::rot_z} {fixed::rot_z_1}
Two rivets (Rivet Q_1 at P_1, Rivet Q_2 at P_2)	fa_1 l_2 l_3	fb_1 l_2 l_3	against aligned aligned	{plane_z::rot_z} {fixed::rot_z_1} {fixed}

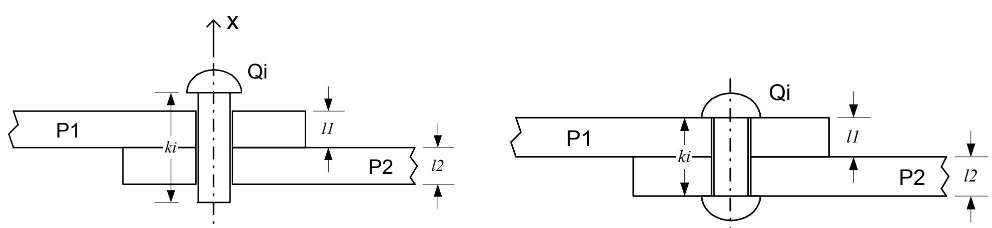


Figure 4-3 Riveting operation

Table 4-3 Implied degrees of freedom reduction for a rivet joint

Config-uration	Condition	Inferred S/R	Implied d.o.f
Two plates	▪ $k_i \le l_1 + l_2$ ▪ $\sum_{i \ne j} j = 0$ where $j = \begin{cases} 1, for k_j \ge l_1 + l_2 \\ 0, otherwise \end{cases}$ ▪ diameter(Q_i) = diameter(*hole_P₁*) ▪ diameter(Q_i) = diameter(*hole_P₂*) ▪ upsettingOperation (Q_i)	▪ against(*bottom_plane*(*head_Qi*), *top_plane*(P_1)) ▪ against(*bottom_plane*(P_1), *top_plane*(P_2)) ▪ aligned(*center_line*(Q_i), *center_line*(P_1)) ▪ aligned (*center_line* (*hole_*P_1), *center_line* (*hole_*P_1))	▪ P_1: {fix | rot_x} ▪ P_2: {fix | rot_x}

Table 4-4 Implied degrees of freedom reduction for multiple rivet joints

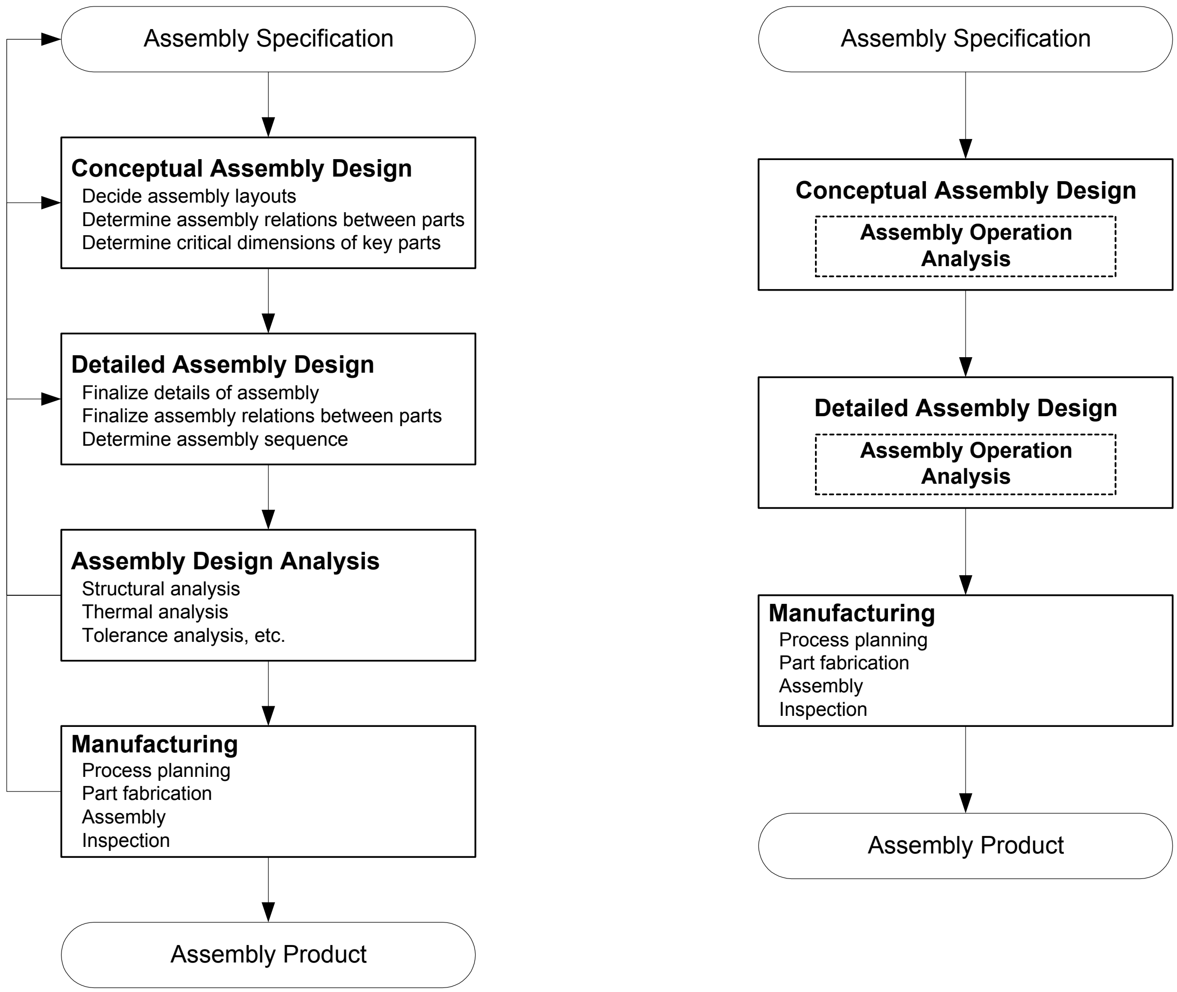
Config-uration	Condition	Inferred S/R	Implied d.o.f
Two plates	▪ $\sum_{i \neq j} j \geq 1$, where $j = \begin{cases} 1, for k_j \geq l_1 + l_2 \\ 0, otherwise \end{cases}$ j=0,1, …, N. N is the number of rivets through P_1 and P_2 ▪ diameter(Q_j) < diameter($hole_j\_P_1$) ▪ diameter(Q_j) < diameter($hole_j\_P_2$) ▪ upsettingOperation (Q_j)	▪ against(*bottom_plane*(*head_*Q_j), *top_plane*(P_1)) ▪ against(*bottom_plane*(P_1), *top_plane*(P_2)) ▪ aligned(*center_line*$_j$(Q_j), *center_line*$_j$(P_1)) ▪ aligned (*center_line*$_j$ ($hole_j\_P_1$), *center_line*$_j$ ($hole_j\_P_1$))	▪ P_1: {fix} ▪ P_2: {fix}

4.2 Virtual Assembly Analysis (VAA)

Many current customers are demanding customization and rapid delivery of innovative products (Welch 1996). Industries now realize that the best way to reduce life cycle costs is to evolve a more effective product development paradigm using the Internet and web based technologies (FIPER 2001, iSIGHT 2002). Yet, there remains a gap between these current market demands and current product development paradigms (Pegasus 2003). One of the reasons for the gap is that the existing CAD systems require that a product developer possess all the design analysis tools in-house making it impractical to employ all the necessary and newest tools. Recently, commercial CAD companies including PTC, SolidWorks, and IBM have shown strong interest in the integration of CAD and CAE environments. These companies have developed their own integrated analysis tools. Nonetheless, those tools are locally integrated and are not sufficient for a distributed, collaborative design environment.

In this work, an innovative Virtual Assembly Analysis (VAA) process is introduced. Unlike the current sequential design process (Figure 4-4-a) used for verifying an assembly design, the

VAA process integrates assembly design and assembly analysis in collaborative *e*-product design. In addition, VAA components are developed to predict the various effects of joining in the actual assembly design stage. The information obtained from the VAA process can guide designers to make appropriate design decisions in the early stages of assembly design (Figure 4-4-b). VAA helps the designer to generate an assembly design for joining and can eliminate the time-consuming feedback processes between the assembly design process and the assembly analysis process. Previous research has largely focused on assembly modeling and assembly process planning without considering assembly operations and their effects in a distributed and collaborative design environment. Thus, there is a strong need to develop a methodology that integrates the assembly design and assembly operation analysis processes in a distributed, collaborative design environment. The developed VAA paradigm provides a concurrent environment for designers to predict physical effects transparently and remotely. The captured physical effects of assembly operations provide information critical to realizing an Internet-based collaborative assembly design environment.



a) Traditional assembly design process b) Integrated assembly design process

Figure 4-4 Assembly design processes

Instead of the current sequential process for verifying an assembly design concept, the virtual assembly analysis (VAA) predicts the various effects of joining during actual assembly design. The VAA process is a transparent and remote assembly analysis process utilized in a service-oriented collaborative assembly design environment. Figure 4-5 illustrates the concept of VAA. An *e*-designer, who participates in the service-oriented collaborative *e*-design, can request analysis services through the Internet/Intranet. An analysis service provider solves the analysis problem requested and provides the results to the *e*-designer. This VAA process is embedded into the distributed assembly design environment and it can guide designers to make appropriate design decisions. It generates an assembly design for joining and eliminates the time-consuming feedback processes between the assembly design and analysis processes. In this research, the VAA process is realized in a service-oriented product development architecture. In the service-oriented architecture, each engineering tool, such as mechanical analysis solvers, can be a server that provides certain services requested by clients.

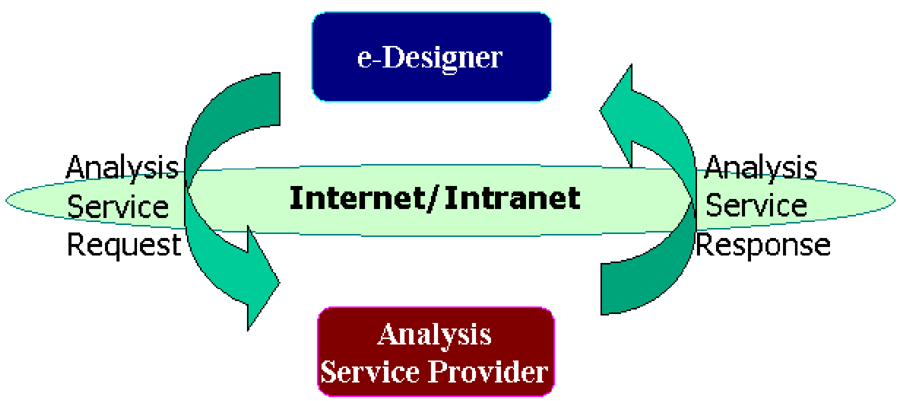


Figure 4-5 Virtual assembly analysis

4.3 Service-oriented VAA Architecture and VAA Service Components

To realize VAA in the service-oriented architecture, an appropriate VAA service triangular relationship should be developed. In this triangular service relationship, each analysis service provider has its own service defined and published at the service manager. For example, an Assembly Design (AsD) engine and the VAA tool provide the services of assembly functional specification, engineering relations construction, and design presentation to end users. Many third-party analysis solvers can serve as the analysis service provider; the ANSYS solver provides the services of structural nonlinearites, heat transfer, dynamics, electromagnetic analyses, etc., and the CFX solver provides the services of CFD (Computational Fluid Dynamics). During the process of service, one service provider may require some other services from other service providers. It then will send a service request to the providers, which provide these additional services. This service chain action should be transparent to the end consumer. For instance, when a design engineer completes the design of two parts, he/she may want to build an assembly model based on the part models. The detailed modeler then calls the assembly procedure. When the assembly model is finished, the design engineer may want to do further mechanical analysis of the assembled parts by calling the service of a FEA tool through the VAA tool. The locations of various service providers are not known until run-time and the relation between the service consumers and the service providers is built dynamically. As illustrated in Figure 4-6, this relation can be viewed as a service chain, which connects service providers with client/server affiliation. Figure 4-7 illustrates the service-oriented VAA architecture. An *e*-designer can request analysis service through the Internet/Intranet. An analysis service provider solves the analysis problem requested and provides the results to the *e*-designer. As shown in Figure 4-7, the VAA architecture consists of four major service components (i.e., VAA tool, Pegasus service manager, *e*-design brokers, and service providers).

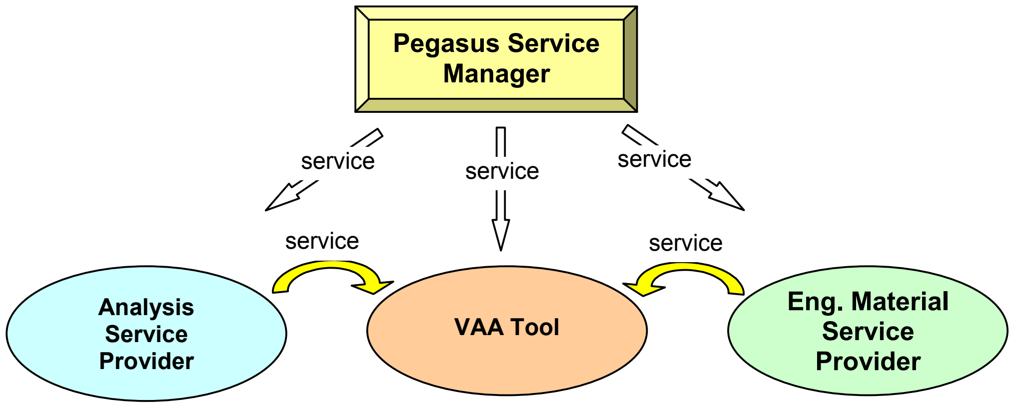


Figure 4-6 VAA service chain

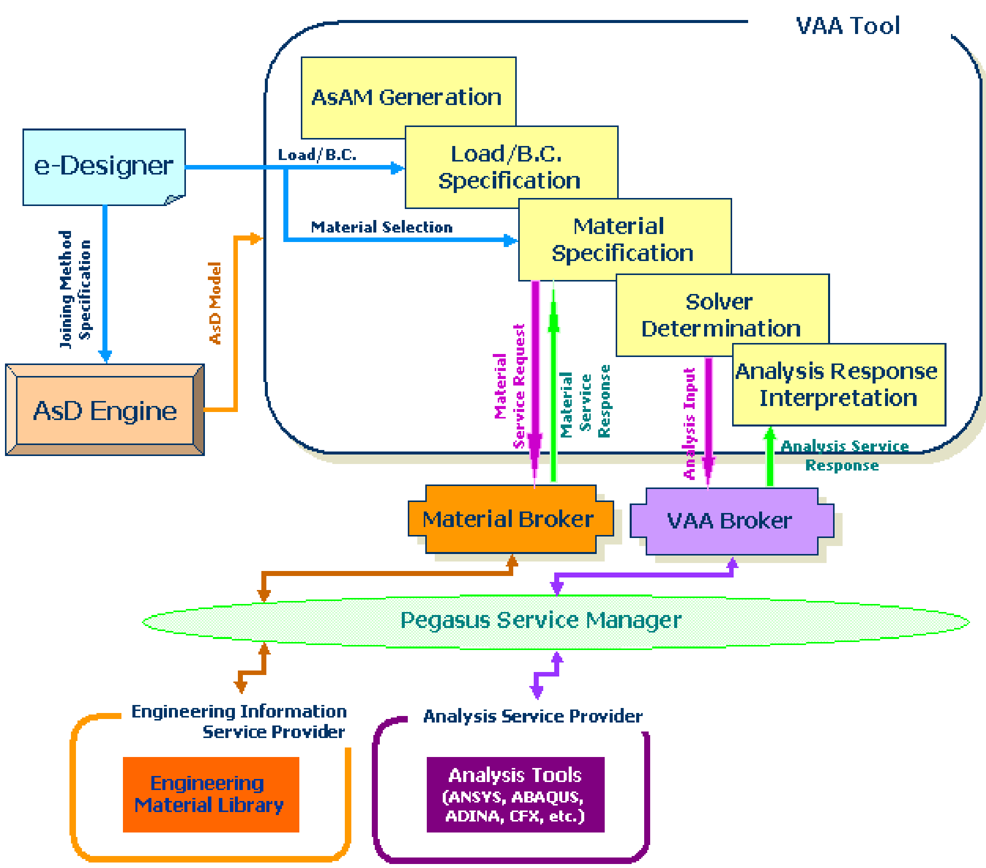


Figure 4-7 Service-oriented VAA architecture

4.3.1 VAA Tool

The VAA tool is an interface for VAA processes. When the designer wants to know the physical effects of the specified joining, the VAA tool is triggered. If the designer doesn't possess any analysis tools in house, and/or has not any expertise in mechanical analysis, the designer can request VAA services remotely and transparently by using this VAA tool.

The assembly operation analysis setup process is cumbersome and requires a certain level of expertise. This process can be automated by imposing assembly/joining information on an AsD model and extracting assembly analysis information from the AsD model. The developed AsD formalism is used to persistently capture assembly/joining information in collaborative assembly design.

4.3.1.1 Assembly Design Formalism and Assembly Design Model Generation. As discussed in Chapter 3, the AsD formalism specifies assembly/joining relations symbolically and the AsD engine generates an AsD model. This AsD formalism is comprised of five phases: 1) spatial relationship specification, 2) mating feature extraction, 3) joint feature formation and extraction, 4) assembly feature formation, and 5) assembly engineering relation construction (see Chapter 3). By interactively assigning spatial relationships, the designer can assemble components together to make final products and infer the degrees of freedom remaining on the components. In assigning spatial relationships, the mating features are defined and extracted from the parts. Mating feature extraction is a preliminary step to capturing joining information. This process provides geometric information directly related to assembly operation. However, the mating feature is not sufficient to represent a joining operation. The joint feature captures the information of actual joining operations. The designer can specify specific joining methods and constraints, such as welding conditions and fixture locations in joint features. After joint features are generated, assembly features are formatted. The purpose of assembly feature formation is to group the mating features and joint features together and thus integrate the data embedded at the component design stage with new assembly information for subsequent processes such as assembly violation detection, process planning, etc. Having designated spatial relationships, mating features, and joint features, the system can then trace back to the component design stage to determine from which design features these mating features originate and what their design specifications are. From the generated assembly features, assembly engineering relations, including assembly/joining relations, are automatically extracted and mating bonds (MB) are generated. A MB is a data structure representing a mating pair and its mating conditions. Assembly engineering relations of the entire assembly are constructed based on the assembly features after specifying the spatial relationships and joining relationships between components. The MBs and the ARM are used to represent the engineering relationships of the entire structure.

From the AsD model, the VAA tool automatically generates an Assembly Analysis Model (AsAM) including the analysis variables, such as environmental variables, loading/boundary conditions, and material properties.

4.3.1.2 <u>Assembly Analysis Model (AsAM) Generation</u>. To integrate assembly design and assembly operation analysis, the assembly design models should be translated to an assembly analysis models. There has been some research conducted to integrate product design and analysis. Peak *et al*. (1998) presented a "multi-representation architecture" of intension of CAD-CAE integration. As an information-intensive mapping between design models to analysis models, a product model-based analysis model is researched and a framework to achieve design-analysis associativity is proposed. Rémondini *et al.* (1998) developed a mechanical analysis module to generate an analysis data model from a geometric data model and mechanical information. Even though their methodology can be solutions for a limited sense of CAD-CAE integration, they have not presented methods to integrate assembly design and assembly operation analysis to capture the physical effects of joining in the distributed assembly design environment. To perform VAA, the assembly/joining information necessary to assembly operation analysis can be extracted using an assembly-analysis solution model to explain physical phenomena based upon the assembly/joining information. The assembly-analysis solution model (AASM) is an implantation of mapping functions (Ω) of assembly design and assembly operation analysis. It translates an assembly design model (AsDM) to an AsAM: ${}_{AsDM}\Omega_{AsAM}$. Figure 4-8 shows how AsAM is generated by AASM.

Figure 4-8 Assembly analysis model generation

Table 4-5 AASM for welding

Assembly Design Model		Assembly Analysis Model
Information	Source Features	
Assembly Component Geometry	Assembly Feature	Geometry
Material Name	Assembly Feature	Material property
Fixture Location	Joint Feature (Joining Constraint)	Fixity location
Joining Conditions: welding condition, such as amperage, voltage, welding speed	Joint Feature (Joining Constraint)	Loading condition e.g., heat input

Table 4-6 AASM for riveting

Assembly Design Model		Assembly Analysis Model
Information	Source Features	
Assembly Component Geometry	Assembly Feature	Geometry
Rivet Geometry	Joint Feature	Geometry
Material Name	Assembly Feature	Material property
Fixture Location	Joint Feature (Joining Constraint)	Fixity location
Joining Conditions: riveting condition, such as upsetting pressure	Joint Feature (Joining Constraint)	Loading condition e.g., pretension load

Tables 4-5 and 4-6 show AASMs for welding and riveting analyses. Information essential for an assembly analysis is extracted from the AsD model, which is generated by the AsD formalism.

The material of the assembly components from the joint features is translated to the material properties for AsAM. Through this mapping, the material property for the specified material name is automatically assigned from a material library. If the resident material library doesn't have the information about the specified material, the material service can be invoked through the service-based architecture; a designer doesn't need to hold all material information in-house.

Heat input on a weldline, which is essential information to perform a weld analysis, can be calculated based upon assembly operation information, such as welding conditions (e.g., amperage, voltage, and welding speed) and material properties. Deposition of weld metal is simulated by defining the weld elements at elevated weld deposition temperatures. All other nodes are defined at the ambient temperature as the initial temperature field. The principal welding heat source is the heat flow or heat output, q (J/s) in continuously acting sources. In arc welding, heat input ($\eta q/v$) is supplied to raise the weld metal with area to the weld deposition temperature (T_d). Here, q is the product of amperage I (A) and voltage U (V) at the arc in the case of direct current (Eq. 4-1). The net or effective heat q is related to the heat efficiency η_h of the welding processes. Table 4.7 shows the heat flow and efficiency of various fusion welding methods. In the case of alternating current, effective values resulting from the momentary products have to be used (generally in the form RI^2_{eff} with ohmic resistance R and effective amperage I_{eff}).

$$q = \eta_h UI = \eta_h RI^2_{eff} \qquad \text{(Eq. 4-1)}$$

The heat input per unit length of weld, q_w (J/mm) is used to consider seam welding with speed v (mm/s). From the equation below, the deposition temperature of a weld can be determined and used as a welding temperature (c and ρ are specific heat capacity and density).

$$q_w = q/v = c\rho\,(T_w - T_\infty) \qquad \text{(Eq. 4-2)}$$

Table 4-7 Output data of fusion welding methods used steel and aluminum (Radaj 1992)

Welding Method	Heat output q (kJ/s)	Welding speed v (mm/s)	Output per unit length q_w (kJ/mm)	Efficiency η_h
Covered electrode	1 - 20	< 5	< 3.5	0.65 - 0.90
Gas metal arc	5 -100	< 15	< 2	0.65 - 0.90
Gas tungsten arc	1 - 15	< 15	< 1	0.20 - 0.50
Submerged arc	5 - 250	< 25	< 10	0.85 - 0.95
Electron beam	0.5 - 10	< 150	< 0.1	0.95 - 0.97
Laser beam	1 - 5	< 150	< 0.05	0.80 - 0.95
Acetylene flame	1 - 10	< 10	< 1	0.25 - 0.85

According to the assembly model information and assembly engineering information, additional geometric features, such as a weld bead for welded joints and a rivet for riveted joints, can be generated for detailed joint modeling. This detailed joint modeling provides a realistic representation for engineering analyses. The configuration of the joint geometric features can be determined automatically from the assembly model information and assembly engineering information. For rivet joints, the designer specifies the location, head type, and radius of the rivet and the AsD model contains the information. For welded joints, the cross-sectional area of the weld bead can be determined from the existing theoretical relationships between the welding conditions and material properties imposed in the AsD model.

4.3.2 Pegasus Service Manager

The Pegasus service manager collaborates with third-party analysis servers (service providers), such as ANSYS, and achieves the VAA process (see Figures 4-6 and 4-7). In this work, the Common Object Request Broker Architecture (CORBA) is used to realize the service-oriented architecture for VAA. CORBA (Siegel 2000) is an architecture and specification for creating, distributing, and managing distributed program objects in a network. It allows programs at different locations and developed by different vendors to communicate in a network through an "interface broker". An ORB (Object Request Broker) acts as a "broker" between a client request for a service from a distributed object or component and the completion of that request. The ORB allows a client to request services from a server program or object without having to understand where the server is in a distributed network or what the interface to the server program looks like.

Service publication and lookup are the primary services provided by the service manager of VAA. As depicted in Figure 4-9, service publication includes *name publication*, *catalog publication* and *implementation publication*, which are provided for service providers. Name publication service is similar to the “white-page” service provided by telephone companies, by which the name of the service provider is published. Catalog publication service is similar to the “yellow-page” service where both the name and the functional description of the service provider are published. Implementation publication service is the procedure by which the service provider makes its implementation and invocation of services public so that clients can invoke the service dynamically. Correspondingly, service lookup includes *name lookup*, *catalog lookup* and *interface lookup*, which are for service consumers. Name lookup service is provided so that consumers can locate the service providers based on service names. Catalog lookup service is for those consumers who need certain services according to their needs and specifications but do not know the names of the services. Interface lookup service provides a way such that consumers can check the protocols of how to invoke the service in the case that clients do not have the knowledge of the service in advance.

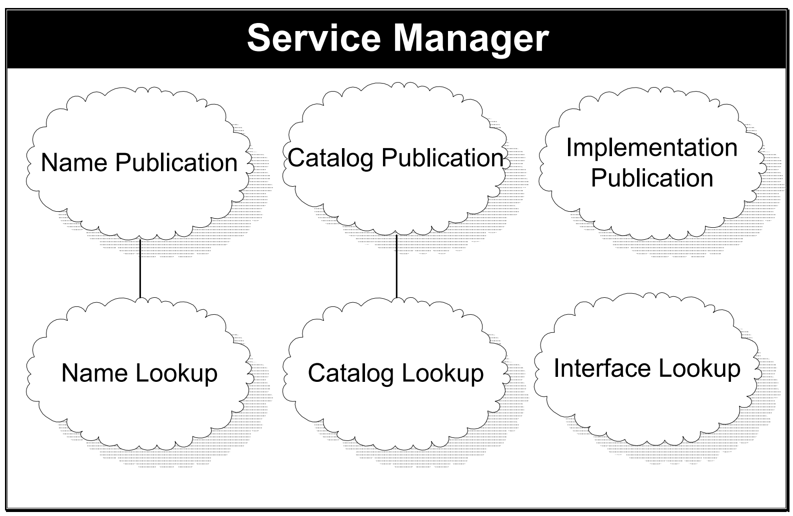


Figure 4-9 Services provided by service manager

Within the system, data transfers and transactions among servers can be completed based upon various distributed computing protocols, such as Hypertext Transfer Protocol (HTTP), CORBA, Distributed Component Object Model (DCOM), Simple Object Access Protocol (SOAP), etc. Currently, the VAA process is implemented by CORBA, which is shown in Figure 4-10. CORBA serves as a bond to integrate the whole system and provides good features of openness for collaborative computation. The components in the distributed system have peer-to-peer relationships with each other. From the end users' outlook, distributing application components between clients and servers does not change the look and feel of any single application, meaning, the system provides end users with a single system image.

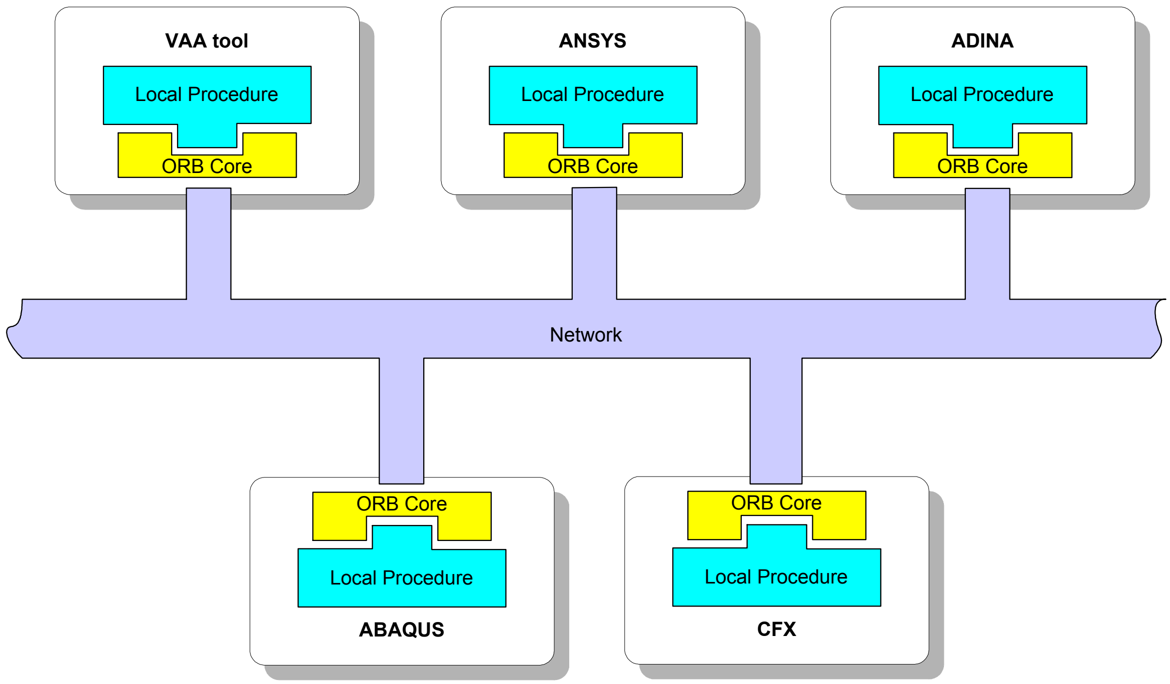


Figure 4-10 Peer-to-peer relationships among VAA components

4.3.3 *e*-Design Brokers

e-Design brokers handle service invocation and service result conveyance through the Pegasus service architecture. The brokers reside in local sites; each client, such as the VAA tool, and each service provider needs the brokers to request or register service. The VAA tool can request the services by invoking these service brokers with relevant service inputs, such as analysis input files and material names. It minimizes the code modification of a service requesting system and provides plug-and-play capability. Figure 4-11 illustrates how the *e*-design brokers are used in the VAA service architecture. Before the VAA process, the analysis service providers register their service through an *e*-design broker at each server site. When VAA service is requested by an *e*-designer, the VAA tool sends a request with an analysis input to the *e*-design broker at the client site and the *e*-design broker conveys the request to the Pegasus service manager. After an analysis result is obtained from the analysis service provider, the Pegasus service manager informs the client's *e*-design broker and conveys the result to the *e*-designer.

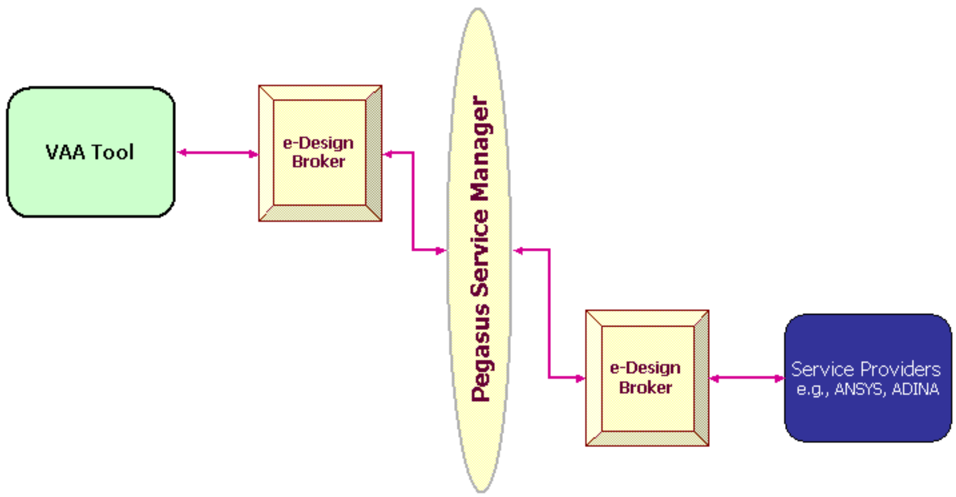


Figure 4-11 *e*-Design brokers and VAA

4.3.4 Service Providers

The Pegasus service manager and the service providers play key roles in the VAA service chain management (Figure 4-6). The Pegasus service manager allocates service resources according to service consumers' demand and service providers' capability and capacity while service providers respond to the requested service.

In this work, two types of service providers are considered: material service providers and analysis service providers. A specialized material service provider can provide the material properties, which are usually too cumbersome to store in the assembly designer's site. The *e*-designer can request certain material properties from the engineering material service provider by specifying material name or certain material specifications. Any available engineering material library can provide relevant material properties to the client. To perform VAA to predict the physical effects of the joining, FEA tools, such as ANSYS, ADINA, and ABAQUS, can provide various FEA services. Generally, FEA tools allow certain command-based external analysis inputs. Depending upon the FEA tools and analysis types, different sets of commands and analysis procedures are needed. Appropriate analysis procedures, including specific analysis commands, can be provided from available analysis service providers through an analysis procedure service. In this work, typical analysis procedures considering the characteristics of joining methods are investigated and appropriate analysis procedures are pre-determined. Analysis service providers provide analysis procedure templates based upon the analysis procedures.

4.4 Implementations of the SRI tool

The SRI tool is developed to capture the SRI of joining, and is embedded into the AsD engine. The following figures illustrate how the SRI tool works to indicate the SRI of joining. The SRI tool compares the inferred d.o.f. of the specified joining and the designed d.o.f., and it indicates whether the joining satisfies the designed d.o.f.

As shown in Figure 4-12, the two plates are intended to be joined and their d.o.f. are fixed by assigning a series of spatial relationships. If a designer wants to join the two plates by applying one structural rivet at p_1, the SRI indicates that the designed d.o.f. (fixed) is under-constrained by the one-rivet (see Figure 4-12). When two rivets are used to join the assembly components, the d.o.f. of components are fully constrained (see Figure 4-13). Figure 4-14 illustrates the SRI of arc welding. Once a weldline is specified, all d.o.f. of the two plates are fixed, so satisfies the designed d.o.f. By using this tool, an assembly intent analysis can be performed to check whether the specified joining method satisfies the original intent on the assembly.

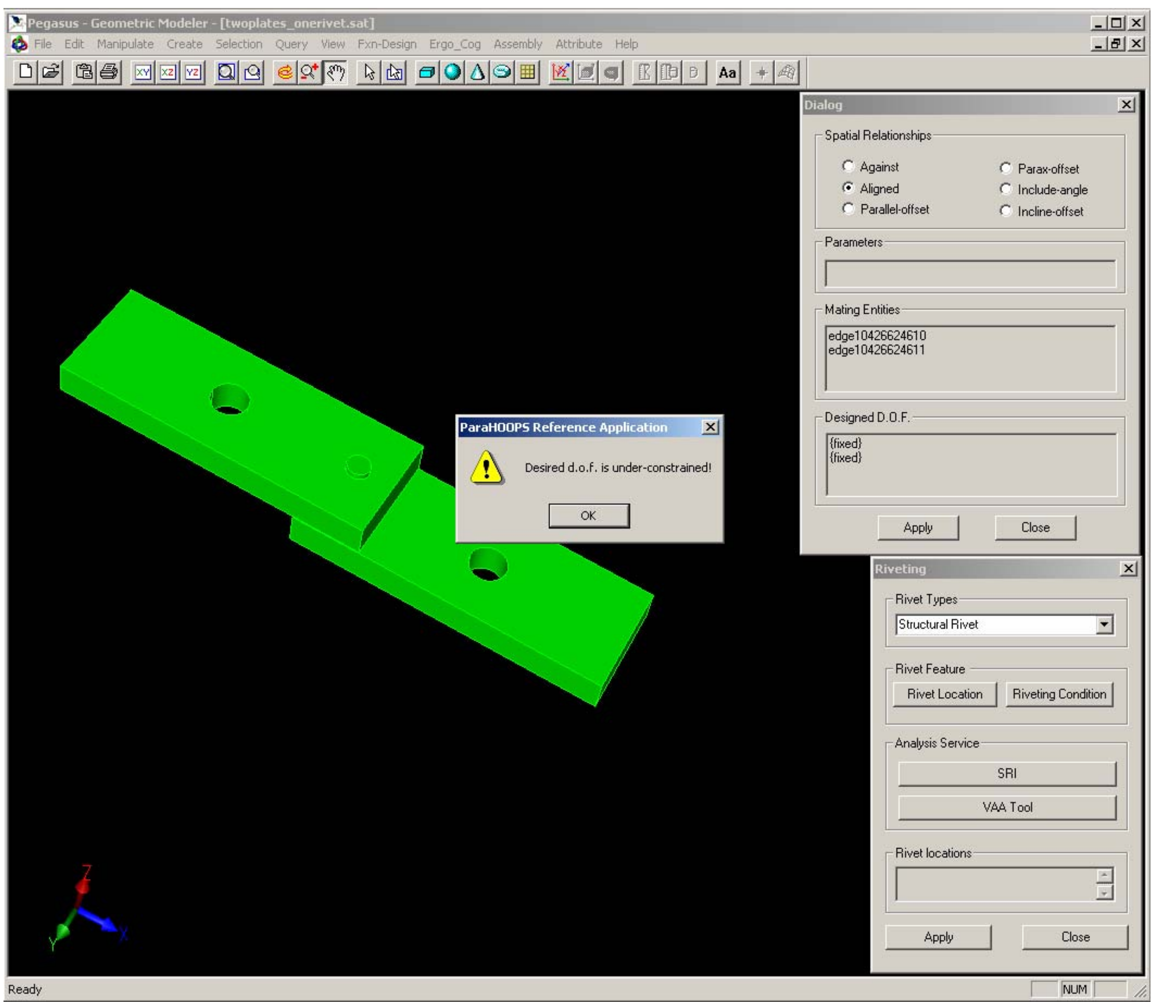


Figure 4-12 SRI tool indicating one-rivet's SRI

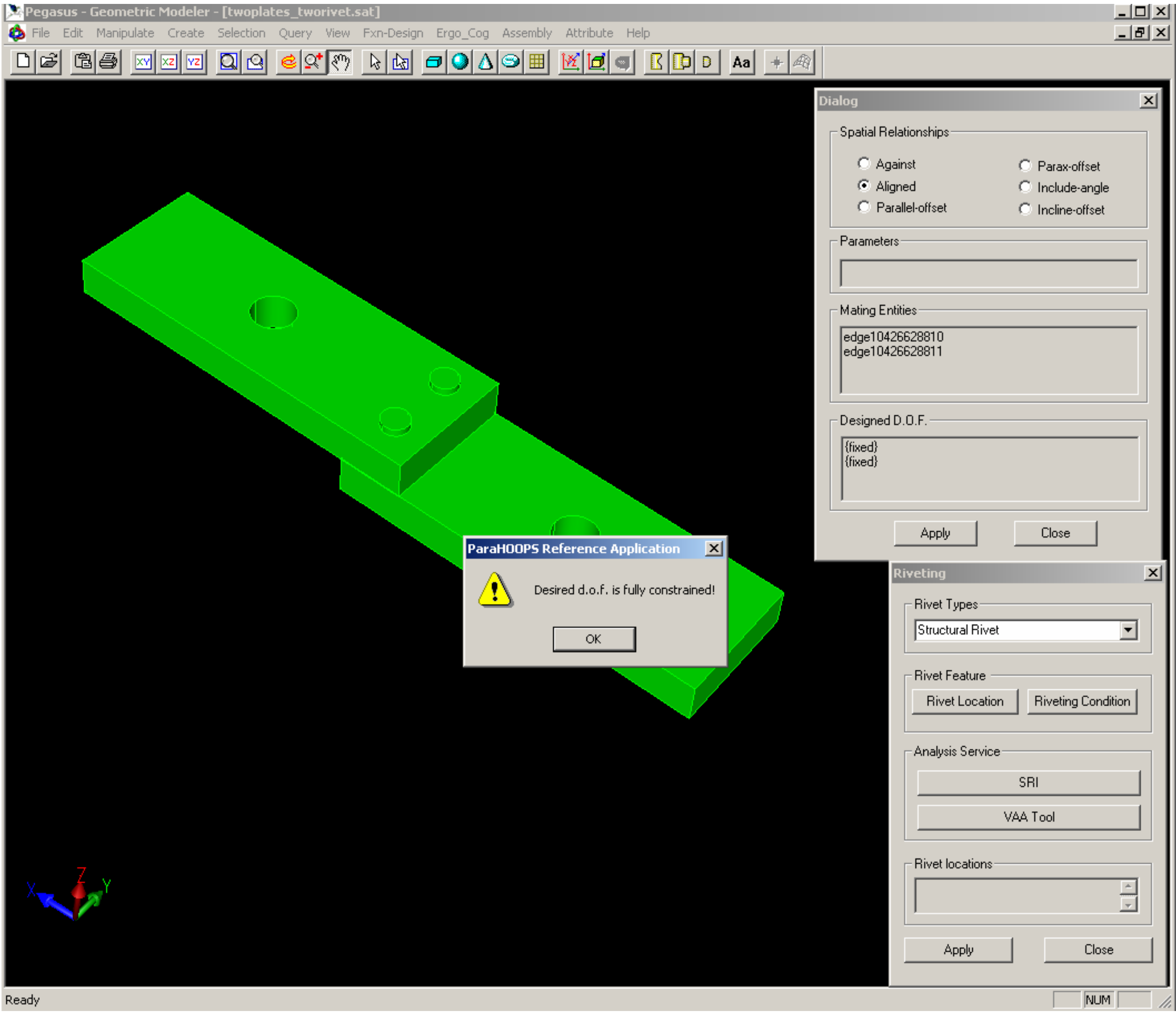


Figure 4-13 SRI tool indicating two-rivets' SRI

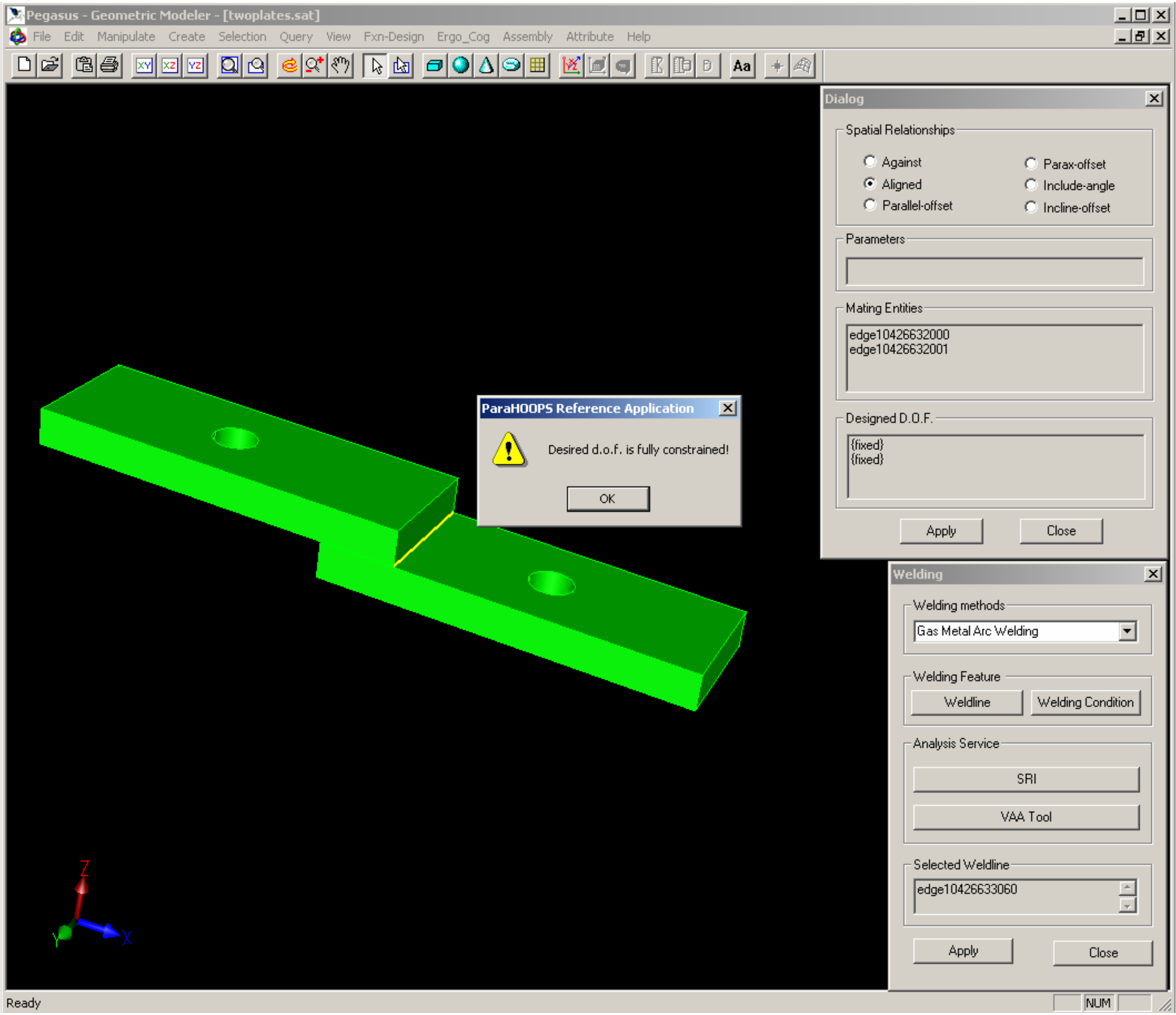


Figure 4-14 SRI tool indicating SRI of arc welding

4.5 Implementations of the VAA tool

The VAA architecture and components are developed to realize the VAA process. This VAA process predicts the physical effects of joining processes, in which the VAA tool is embedded into assembly design processes in collaborative product design environments. To realistically predict physical effects of joining, appropriate analysis procedures are required. The next sub-section describes examples of VAA procedures.

4.5.1 Examples of VAA Procedures

VAA for assembly operations requires specific analysis methodology and procedures. In this work, as a case study a thermo-structural analysis is used to understand the thermal and structural behavior of arc welding. In addition, structural analysis is employed to predict various structural phenomena of riveting. To enable VAA for specific joining processes, proper analysis procedures must be pre-investigated and built into an analysis procedure library.

4.5.1.1 <u>Thermo-Structural Analysis on an Arc Welding Process</u>. A thermo-structural analysis coupled by nonlinear heat conduction analysis and steady-state structural analysis, is used to analyze thermal distortion effects of welding operations. The thermo-structural analysis consists of: 1) steady-state thermal analysis to model heat input from welding; 2) transient thermal analysis to model cooling process after welding; and 3) transient structural analysis to obtain thermal distortion from welding.

$$\rho c\left(\frac{\partial T}{\partial t}+v_x\frac{\partial T}{\partial x}+v_y\frac{\partial T}{\partial y}+v_z\frac{\partial T}{\partial z}\right)=\dddot{q}+\frac{\partial}{\partial x}\left(K_x\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(K_y\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(K_z\frac{\partial T}{\partial z}\right) \quad \text{(Eq. 4-3)}$$

The transient heat flow in a three-dimensional isotropic solid bounded by a surface without internal heat generation is governed by the energy conservation equation (Eq. 4-3) in the Cartesian coordinate system (x, y, z) (Bae and Na 1995, Lewis *et al.* 1996, ANSYS 2002 –a). Here, T is temperature (= $T(x,y,z,t)$); c is specific heat; t is time; $\{v\}$ is the velocity vector for mass transport of heat, $\{v_x, v_y, v_z\}$; $\{q\}$ is the heat flux vector; $\dddot{q}$ is the heat generation rate per unit volume. In this work, it is assumed that there is no heat source ($\dddot{q} = 0$) and K_{xx}, K_{yy}, K_{zz} are constants.

Steady-state thermal analysis

This steady-state thermal analysis models conduction of heat from welding. When obtaining the temperature distribution of the weldments, it is assumed that welding is done instantly and the weld sequence is not considered. The conduction of heat is governed by Eq. 4-3 (in steady-state thermal analysis, $\partial T/\partial t = 0$). As the boundary conditions of this analysis, certain temperatures are specified over surfaces of welded bodies and surfaces of welds: $T_{body} = T_\infty$ and $T_{weld} = T_w$, where T_∞ is ambient temperature and T_w is welding temperature. The welding temperature can be determined based upon the deposition temperature of the welding conditions (explained in section 4.3.1.2).

Transient thermal analysis

Heat transferred from welding is conducted through the body. During welding, conduction is assumed to occur without transferring heat due to airflow. Steady analysis is therefore enough to explain its phenomenon as previously stated; however, the cooling process after welding is assumed to be dominated purely by convection. It requires transient thermal analysis. Eq. 4-4 with the initial and boundary conditions below is solved entirely using a finite element method. Two distinctive processes in weld modeling are coupled with the initial boundary conditions required in the second process. The temperature of the entire surface obtained from the steady conduction process is then imposed as our initial conditions for the purpose of the transient analysis. Convection boundary conditions are specified on all surfaces without considering the radiation effect by the assumptions.

$$\{q\}^T\{\eta\} = h_f\left(T_S - T_B\right) \text{ (Eq. 4-4),}$$

where $\{\eta\}$ is the unit outward normal vector; h_f is the convection coefficient; T_B is the bulk temperature of the adjacent fluid; T_S is the temperature at the surface of the model. The temperature history obtained is needed to perform transient structural analyses.

Transient structural analysis

Using the results from the previously described thermal analyses, coupled thermal-structural analyses are done to calculate the thermal distortion of the entire weldment. For the thermal distortion analyses, the weldment and weld beads are modeled by converting the SOLID70 element used for thermal analyses into the SOLID45 element (8-node fully coupled temperature-displacement solid element) (ANSYS 2002-b). This transient structural analysis is required to solve the thermal distortion problem of welding. The surface temperature history works as an input to determine thermal strains of all nodes.

The stress is related to the strains by

$$\{\sigma\} = [D]\{\varepsilon^{el}\} \text{ (Eq. 4-5),}$$

where $\{\sigma\}$ is stress vector, $[\sigma_x\ \sigma_y\ \sigma_z\ \sigma_{xy}\ \sigma_{yz}\ \sigma_{xz}]^T$; $[D]$ is the elasticity/elastic stiffness matrix or stress-strain matrix; $\{\varepsilon^{el}\} = \{\varepsilon\} - \{\varepsilon^{th}\}$ = elastic strain vector; $\{\varepsilon\}$ is total strain vector, $[\varepsilon_x\ \varepsilon_y\ \varepsilon_z\ \varepsilon_{xy}\ \varepsilon_{yz}\ \varepsilon_{xz}]^T$; $\{\varepsilon^{th}\}$ is the thermal strain vector, $\Delta T[\alpha_x\ \alpha_y\ \alpha_z\ 0\ 0\ 0]^T$; α_x, α_y, α_z are thermal coefficients of expansion in the x, y, and z directions, respectively; ΔT is $T\text{-}T_{ref}$; T_{ref} is the reference temperature.

Thermal strain (Eq. 4-6) is determined from the surface temperatures obtained from the previous thermal analyses.

$$\varepsilon^{th} = \alpha\left(T - T_{ref}\right) \quad \text{(Eq. 4-6)}$$

All degrees of freedom in the locations of the fixture are constrained. The locations are obtained from joining constraints of joint features.

4.5.1.2 Structural Analysis on Riveting Process. The finite element modeling for the shear lap rivet joints is performed using elastic-plastic structural analysis. The shear lap rivet joint involves two composite or metallic plates joined by single or multiple rivets. The structural analysis employs SOLID45 element, an 8-noded isoparametric quadrilateral solid element (ANSYS 2002-b). The material behaviors of the rivet and the assembly components are determined material constraints specified by users. A major difficulty in modeling the rivet and the plate is the idealization of the load transfer between the rivet and the plate. The resulting stress distribution around the rivet hole is largely influenced by the procedures followed in the idealization. The pretension or preload caused by the tightening of the rivet can be simulated by specifying pretension elements, PRETS179, on the specified preload across a pretension section (ANSYS 2002-b).

4.5.2 Demonstration

To enable VAA, four major service components serve in the developed service-oriented architecture: the VAA tool, the transaction manager, service brokers, and third-party analysis service providers. The Pegasus service manager is used as a transaction manager.

In this demonstration, the VAA tool is implemented in the AI*Workbench environment of ANSYS, Inc. The ANSYS solver is employed as the analysis service provider. Engineering material information is represented in XML format in the material database. The Pegasus service manager is implemented in Java. *e*-Design brokers are implemented in C++. IONA's ORBacus implementation of CORBA is used in the service architecture.

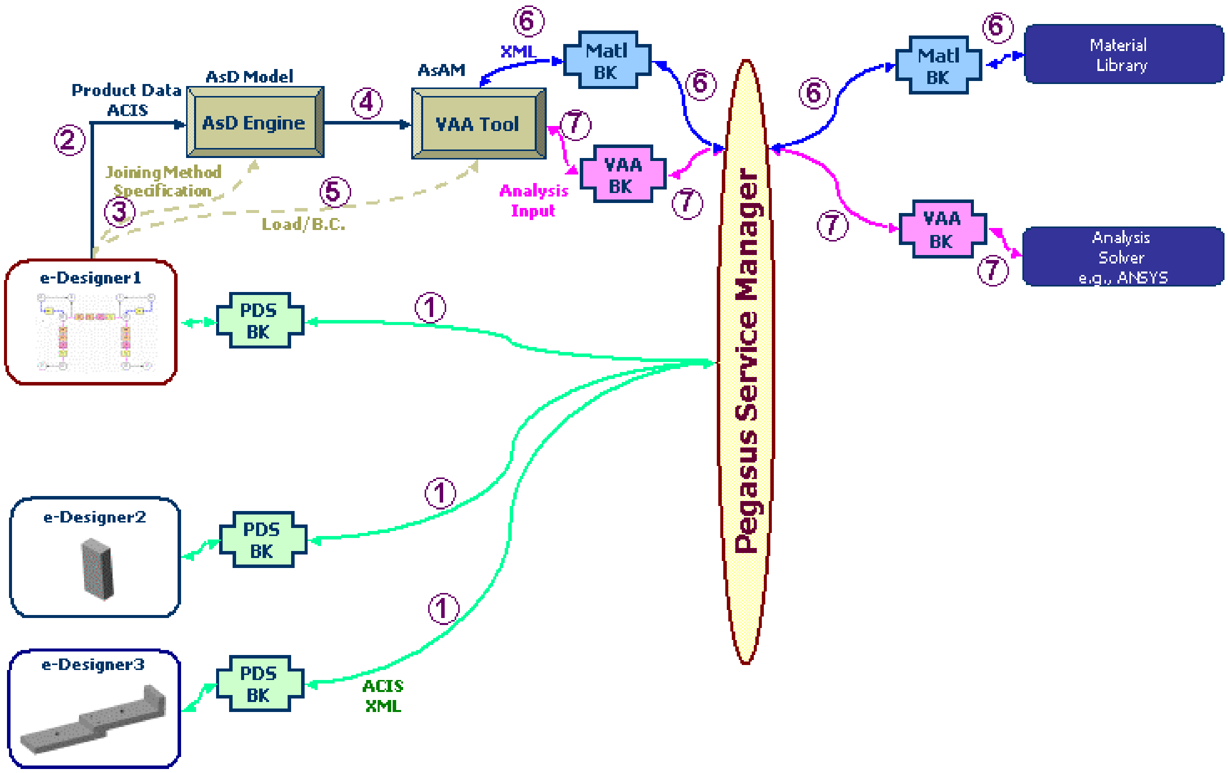


Figure 4-15 Service transactions in VAA

Figure 4-15 illustrates the transaction flow of services for VAA. Detailed processes are described below. The numbers in the figure stand for the index of each process.

STEP 1: *e*-Designers can exchange product data, such as AsD models, and select assembly components through the Product Data Sharing (PDS) service (①) (Explained thoroughly in Chapter 3).

STEP 2: The selected assembly components are loaded in an AsD engine to generate joints (②). The system integrator, *e*-designer 1, can specify joining methods on the assembly (③).

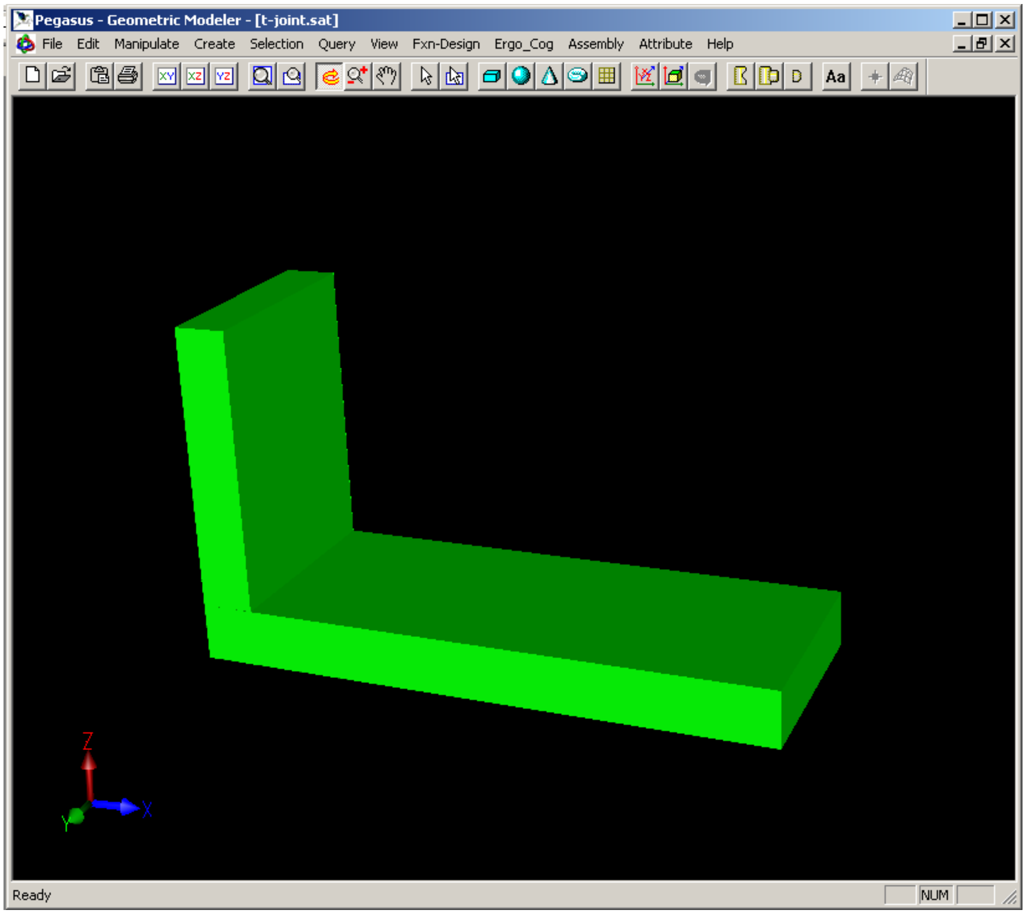
STEP 3: When the *e*-designer wants to know physical effects of the specified joining, the VAA tool is triggered and a newly generated AsD model is sent to the VAA tool

(④). From the AsD model, the VAA tool extracts analysis information and generates an AsAM. The designer can add additional loading and boundary conditions (⑤).

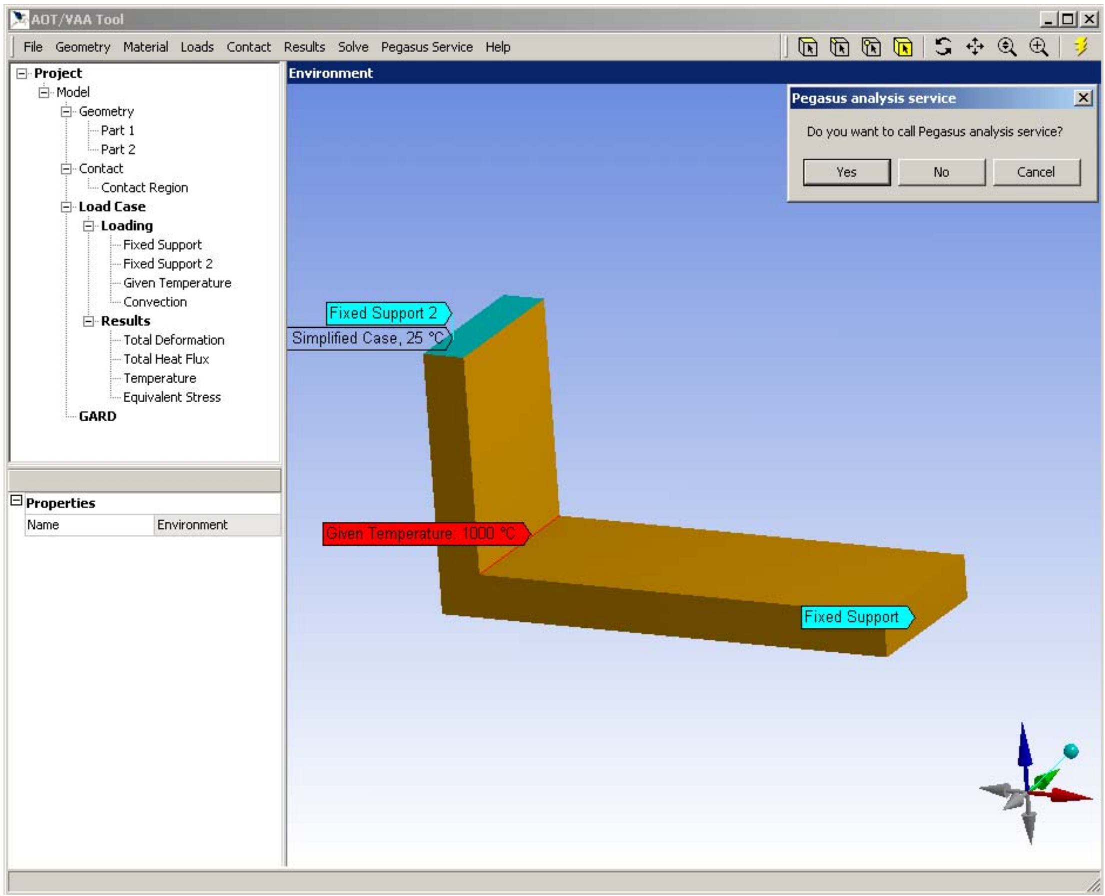
STEP 4: If the material specified in the AsD model doesn't exist in a local database, the material property is obtained from remote material libraries though the service-oriented architecture. The designer can also request a certain material to be entered in the VAA tool. The VAA tool dynamically requests the service by invoking the material service broker (Mtl BK) with relevant material information (⑥).

STEP 5: Once the VAA inputs are ready, the VAA tool invokes the VAA service broker (VAA BK) with the VAA input. When the analysis is completed, the analysis service provider returns the analysis results to the VAA tool (⑦).

As shown in Figure 4-15, PDS service, material service, and VAA service are accomplished through service brokers (i.e., PDS broker, material broker, and VAA broker). These service brokers at the user's site handle service invocation and service result conveyance through the service-oriented architecture. The VAA tool can request the services by invoking these service brokers with relevant service inputs, such as analysis input files and material names. Figure 4-15 also illustrates how the service brokers are used in the service architecture. For example, *e*-designers can exchange product data, such as AsD models, and select assembly components through PDS service.



a) Assembly design model



b) Assembly analysis model

Figure 4-16 Assembly models for VAA

The developed VAA tool (see Figure 4-16-b) is used as an interface to capture assembly and joining specifications. When the designer wants to know physical effects of the joining, the VAA tool is triggered to interpret the AsD model (Figure 4-16-a). From the AsD models, the VAA tool automatically generates an AsAM including the analysis variables, such as environmental variables (e.g., as convection and fixed support), loading condition (e.g., given temperature and force/pressure), and material properties (e.g., Young's modulus, specific heat, and thermal expansion coefficient) (see Figure 4-16-b). The joining parameters (e.g., welding conditions) re extracted from the AsD model and relevant analysis variables are obtained and assigned to the AsAM. For example, the degrees of freedom at fixture locations are restricted as fixed supports. Temperature at the specified weld seam is estimated from the welding condition. Through this analysis setup process, the designer can impose additional analysis constraints on AsAM in the VAA tool.

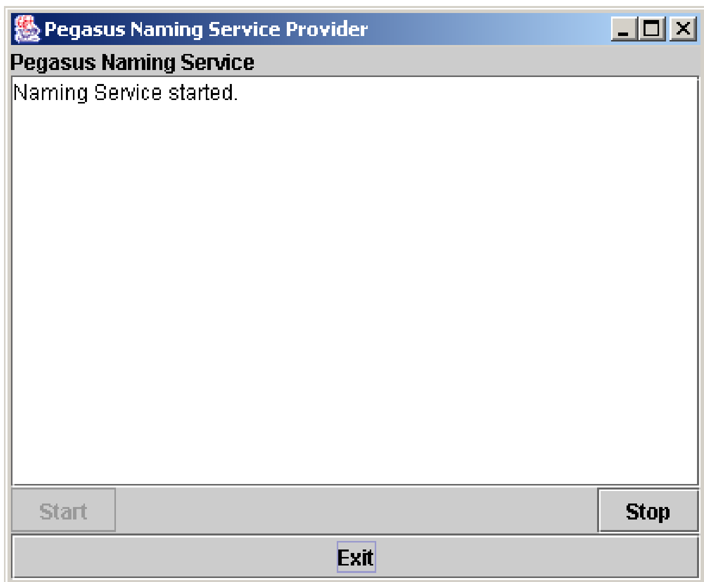


Figure 4-17 Pegasus service manager

The locations of various service providers are not known until run-time. The relation between the service consumers (such as the VAA tool) and the service providers is built dynamically. The Pegasus service manager allocates service resources according to service consumers' demand and service providers' capability and capacity. Figure 4-17 shows an implementation of the Pegasus service manager.

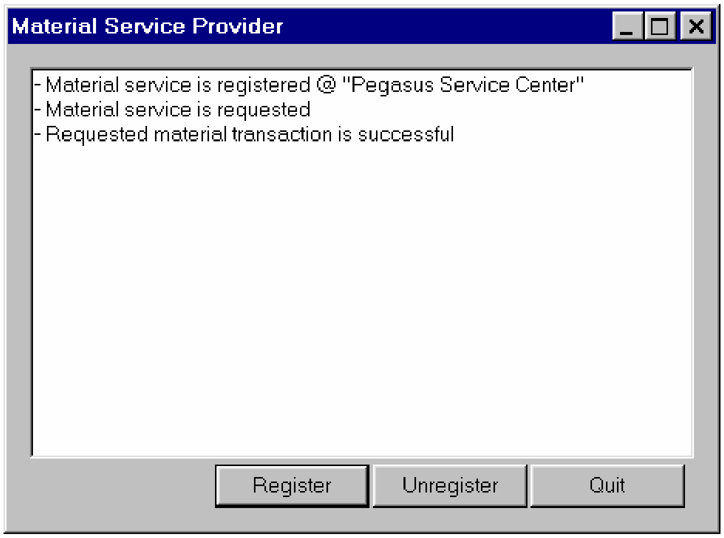


Figure 4-18 Engineering material service provider

A specialized material service provider can provide the material properties, which are usually too cumbersome to store in the assembly designer's site. Here, an engineering material service provider (see Figure 4-18) has this information and offers engineering material lookup services (see Figure 4-19). To perform VAA to predict the physical effects of the joining, the VAA tool (transparent to the analysis service provider) looks up and acquires the material information on the specified material type from the remote engineering material service provider.

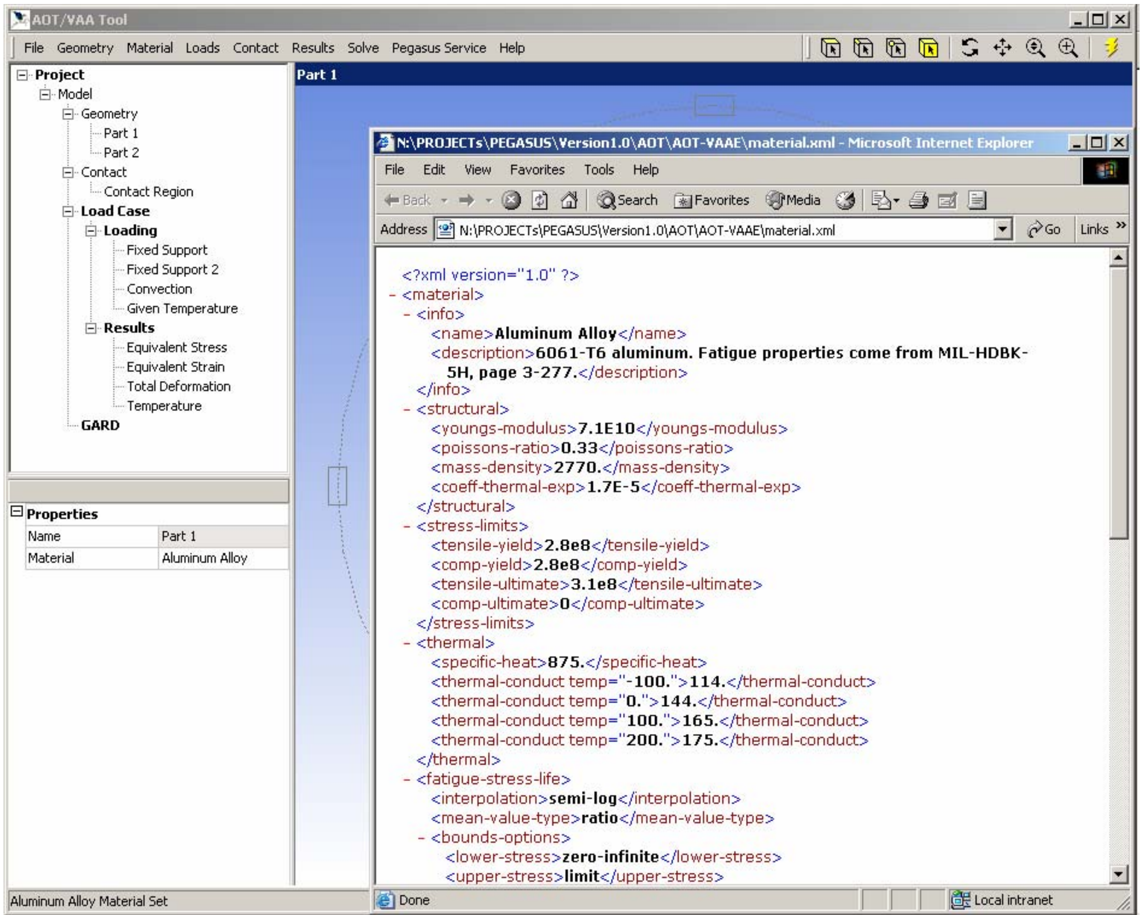


Figure 4-19 Material obtained by material service

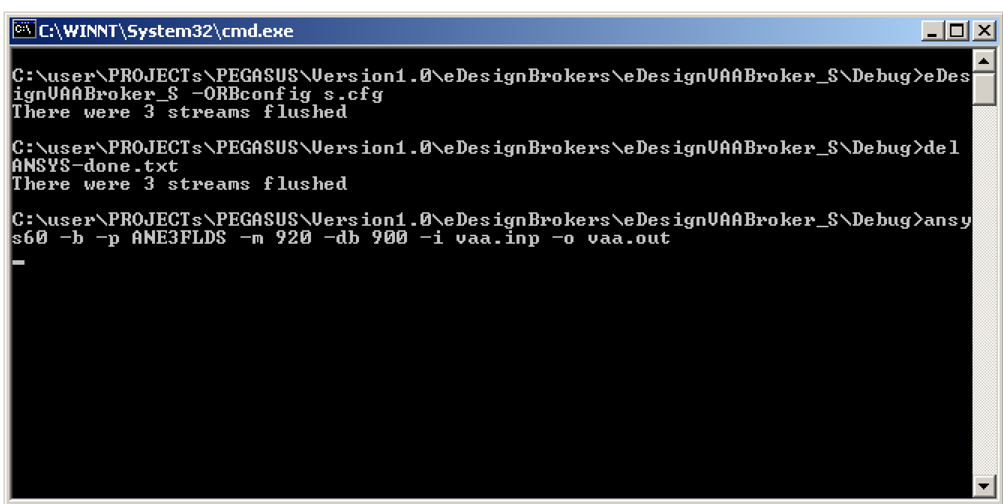


Figure 4-20 VAA service provider

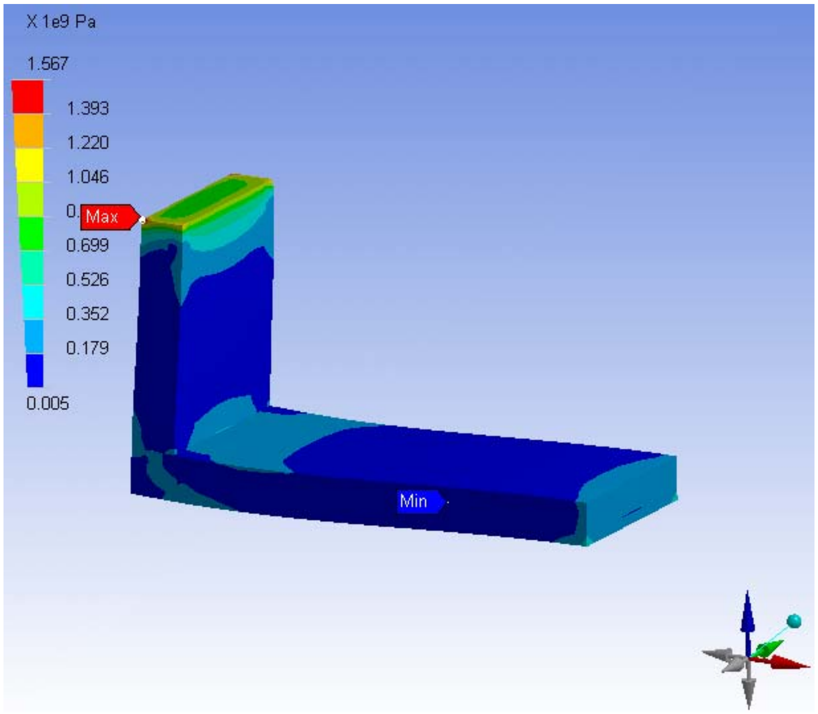


Figure 4-21 Equivalent stress and deformation obtained from VAA service

Once a complete AsAM is generated, and VAA service can be invoked. VAA input for available VAA service providers is generated by the VAA tool considering specified joining method's characteristics and analysis preferences. For example, if the designer wants to perform a thermal analysis for the welded joint, the tool can generate appropriate inputs for the available VAA service provider (Figure 4-20) to perform the thermal analysis. In this work, predetermined analysis procedures are used for VAA. Determining appropriate analysis procedures is very important for obtaining realistic analysis results. The service-oriented architecture provides an environment in which new analysis procedures are easily acquired from remote analysis service providers. Appendix C.1 shows an example of ANSYS analysis input generated by the VAA tool. When the analysis is completed, the analysis service provider (see Figure 4-20) returns the analysis results (e.g., output files, animation movies) to the VAA Broker, and eventually to the VAA tool (see Figure 4-21).

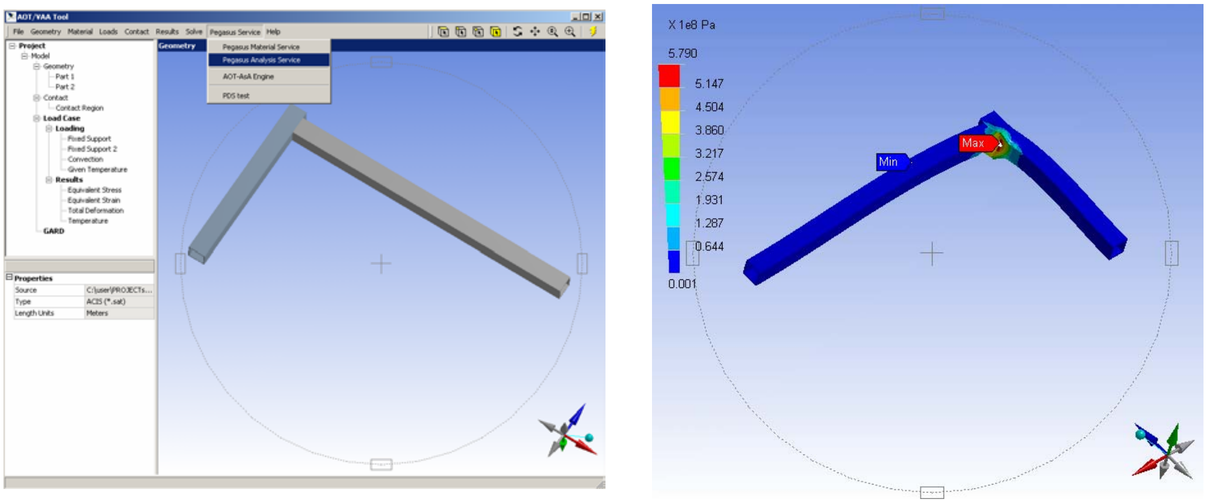


Figure 4-22 VAA analysis for a welded extruded frame

The VAA process is also implemented using realistic examples, such as an aluminum space frame assembly for an automobile (Figure 4-22) and hinge assembly with rivet joints (Figure 4-23). The material used for the space frame is aluminum 6061 extrusions. Recent emphasis on lightweight environmentally sound car design has opened up the possibility of substituting lower-density corrosion-resistant recyclable aluminum for steel in car bodies (Ashley 1994). However, the high distortion of aluminum alloy is a difficult problem to overcome to achieve precision manufacturing. Figure 4-22 illustrates the VAA result of a welded extruded frame. The result clearly shows deformation of this structure and stresses concentrated at the welded joint. Deformation beyond allowable tolerance will be indicated easily. Based on this result, the designer can make a decision on whether this joining method is feasible within this nominal geometry. This car frame example is thoroughly considered to validate the concepts and techniques developed in this work (see Chapter 6).

As another illustration, a hinge assembly with rivet joints is used. The material of the hinge is a structural steel. Figure 4-23 shows a structural VAA result for the hinge joint. Based upon this result, although stresses are concentrated on the top component and the stress affects one of rivets, the designer can clearly see that this joint is robust in this specific test environment.

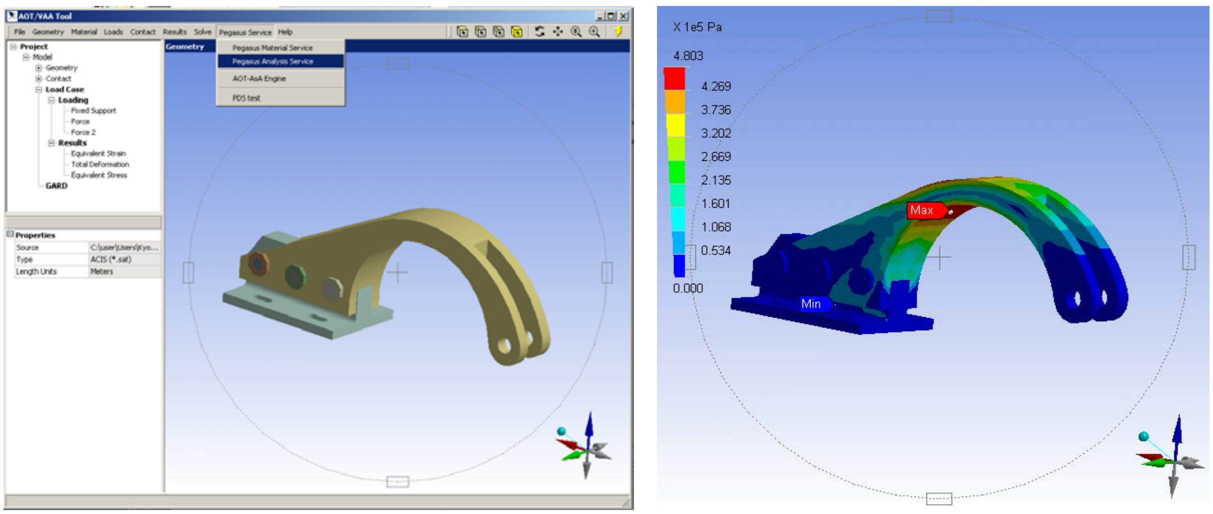


Figure 4-23 VAA analysis for a hinge with three rivets

4.6 Summary

The aim of this work is to integrate assembly design and assembly analysis in a service-oriented collaborative product development environment, *e-product design and realization* environment. An intelligent assembly design system should be able to assist a designer during joint design processes by predicting expected assembly design problems and providing alternative suggestions. Traditional solid modeling systems for assembly components, while adequate for visualization purposes, do not support downstream life-cycle activities. Furthermore, the existing CAD systems are unable to show the mathematical and physical effects of joining, such as thermo-mechanical effect of a weld. Currently, the effect of joining is analyzed upon completion of assembly modeling. This sequential process is arduous and time-consuming.

In this work, a new assembly analysis framework, *virtual assembly analysis* (*VAA*) and an AsI engine including a SRI tool and VAA tool are developed to predict the spatial relationship implication and physical effects of selected joining processes in the *e*-product design and realization environment. The SRI tool is developed to perform a designer's intent analysis for joining. The VAA architecture and components are developed to predict physical effects for mechanical assemblies. Unlike the typical, sequential design process used for verifying an assembly design, the VAA process transparently and remotely integrates assembly design and assembly analysis. The information obtained from the VAA process can guide designers to make appropriate design decisions in the early stages of assembly design. VAA helps the designer to generate an assembly design for joining and can eliminate the time-consuming feedback processes between assembly design and assembly analysis. The developed VAA framework provides a concurrent environment for designers to predict physical effects transparently and remotely. The captured physical effects of joining provide information critical to realizing an Internet-based collaborative assembly design environment. Using the VAA framework, decentralized assembly design tools can be efficiently integrated and collaborate with each other through the Internet. As a demonstration, the developed SRI tool and VAA tool are applied on the connector assembly and other realistic assembly examples. The contributions of this work are summarized below.

Contributions

1. The designer's original intent imposed on assembly design can be analyzed by comparing d.o.f. implied by specific joining methods and d.o.f intended by the designer. The SRI tool embedded in AsD engine can perform the intent analysis.
2. The VAA process transparently and remotely integrates assembly design and assembly analysis. It eliminates the time-consuming feedback processes between assembly design and assembly analysis. The information obtained from the VAA helps the designer to generate an assembly design for joining.
3. By using the VAA architecture, an assembly designer doesn't need to possess whole mechanical analysis capabilities in house. The VAA service can be invoked transparently and remotely through a service-oriented VAA architecture.
4. The developed service-oriented VAA architecture is scalable and extendable. This architecture provides an environment, in which new analysis tools and information are easily acquired from remote analysis service providers.

5.0 ASSEMBLY ADVISORY ENGINE

While the AsD engine and the AsI engine result in the realization of design for assemblability, the Assembly Advisory (AsA) engine supports a designer's decision on joining. In this work, a new Assembly Design Decision Making (ADDM) framework is developed to propose assembly alternatives to the designer by considering assembly design and implication information, obtained from the AsD and AsI engines, and assembly/joining and material knowledge bases.

5.1 The Assembly Design Decision Problem

An assembly design decision (ADD) problem occurs when the current assembly design violates the assembly specification, such as maximum allowance in surface straightness and maximum stress. When a problem on the current assembly design is indicated, a designer should make a decision whether to accept the current joint or modify it. If the joint should be modified, then should the current joining method be controlled or another joining method considered? Assembly design decision making (ADDM) provides appropriate decision on this dilemma. Until now, the ADD problem has been merely considered.

Hence, this chapter introduces a new method to resolve the ADD problem, called ADDM. The ADDM will propose assembly alternatives to the designer by considering assembly design information including physical effect information, assembly/joining knowledge, and material knowledge. A hierarchical semantic net (HSN) model is introduced as a core model to represent evaluation knowledge and assembly design knowledge, which is inevitable in knowledge-based design decision making like ADDM. In the HSN model, the semantic net is embedded in the alternative, which is a component of the Analytic Hierarchy Process (AHP) model. In general, the ADD problem is a multicriteria decision making problem. AHP is known as one of the well-respected multicriteria evaluation methods. However, since the ADD problem is by nature knowledge-intensive, the typical AHP model lacks knowledge representation power. Hence, a semantic net is employed to represent inner knowledge of assembly design alternatives of the AHP model.

5.2 Current Multicriteria Decision Making Techniques

Decision making is the process of making choices or reaching conclusions. Many theories and models have been reported in literature. When the feasible set of choices of a decision consists of a finite number of alternatives, the problem is known as a multicriteria evaluation problem, sometimes as a discrete multicriteria problem or selection problem. Many discrete selection techniques are found in the literature. The list of the existing technologies are as follows: the outranking approach (ELECTRE) by Roy (1973); ORESTE by Roubens (1982) and Pastijn and Leysen (1989); PROMETHEE by Brans *et al*. (1984); multi-attribute utility theory (MAUT) by Keeney and Raiffa (1976); the analytic hierarchy process (AHP) by Saaty (1977); the regime method by Hinloopen *et al*. (1983); the convex cone approach by Korhonen *et al.*

(1984); the hierarchical interactive approach by Korhenen (1986); the visual reference direction by Korhonen (1988), and the fuzzy set theory (Zadeh 1965, Zeleny 1982, Zimmermann 1991).

When the number of alternatives of a decision is uncountably infinite, they are not specified directly, but defined in terms of decision variables. This type of problem is called a continuous decision problem and is also referred as a multicriteria design problem or a continuous multicriteria problem. For simple problems with linear objectives and linear constraints, they can be modeled by linear programming. Much research has been done to develop multiple criteria design methods. Clarnes and Cooper (1961, 1978) proposed goal programming and data envelopment analysis (DEA) approaches. Geoffrion *et al.* (1972) used an interactive approach. Korhonen and Laakso (1986) introduced a referenced direction method. Korhonen and Wallenius (1988) used a Pareto race approach. Steuer and Choo (1983) introduced interactive weighted Tchebycheff procedures.

5.2.1 Decision Making in Design and Manufacturing

There has been much research reported on the topic of decision making for product design and manufacturing. Subru *et al*. (1999) used genetic algorithms for a design-manufacturing-supplier decision problem for an agile manufacturing environment. Rekiek *et al.* (2002) proposed a method to treat the resource planning for the assembly line problem, which was based upon a multiple objective grouping of generic algorithms, the branch-and-cut method, and the multicriteria decision support method. In their work, designer's preferences were captured by adjusting the weight of the different objectives. Zha (2002) introduced knowledge intensive Petri net models to integrate design and assembly planning and utilized knowledge-based agents acting as decision supporting tools. However, his research has not addressed how assembly design decisions can be made during the actual assembly design process considering joining. LeBacq *et al*. (2002) presented a methodology for the selection of joining methods. Their method was based on a questionnaire and a database including the characteristics of joining and the material, without considering the physical effects of the joining processes. None of the existing research has tackled the assembly design decision making problem considering assembly design knowledge and assembly implication knowledge.

Determining a proper assembly design has a multi-disciplinary nature. The early stage of an assembly design requires negotiations between diverse stakeholders, such as manufacturing engineers versus financial specialists and the triad of marketing analysts versus quality control experts versus designers; all must resolve performance goals and other trade-offs (Klein 1991, Peña-Mora *et al*. 1995, Singh and Johnson 1998, Gobeli *et al*. 1998). These negotiations point to the need for high-level interaction of multiple experts or sources of knowledge, including both humans and computer programs.

5.2.2 Analytical Hierarchy Process

The Analytical Hierarchy Process (AHP), developed by Saaty (1974, 1977) in the 1970's, is a general theory of measurement processes. It is used to derive ratio scales from both discrete and continuous paired comparisons in multi-level hierarchic structures. These comparisons may be taken from actual measurements or from a fundamental scale of absolute numbers, that reflect the relative strength of preferences, applied to homogeneous clusters of elements. The use of pivots from cluster to cluster inherently extends the scale through paired comparisons for beyond the 1 to 9 range. AHP has found its widest applications in multicriteria decision making, in planning and resource allocation, and in conflict resolution.

AHP is a systematic procedure for representing the elements of any problem. It organizes the basic rationality by breaking down a problem into its smaller constituent parts and then calls for only simple pairwise comparison judgments to develop priorities in each hierarchic level. It provides a comprehensive framework to cope with the intuitive, the rational and the irrational, and emotional at the same time. It is a method used to integrate perceptions and purposes into an overall synthesis. AHP does not require that judgments be consistent or transitive. The degrees of consistency (or inconsistency) of the judgments are calculated at each stage of the process.

5.3 Semantic Net

A semantic network (or net) is a unifying approach for pictorial knowledge representation (Chang 1989, Burns *et al*. 1989, Rada *et al*. 1989). One reason is that a semantic net offers an intuitive representation for pictorial knowledge; moreover, since the basic representation is a graph, a semantic net can be generalized to represent complex logical relations. A semantic net can also represent engineering relations. Greenhill and Venkatesh (1998) showed that semantic network-based representation allows for a consistent knowledge representation and it gives the ability to easily extend a knowledge model while retaining its semantics.

A simple example of a semantic net can be found in Figure 5-1. The t-joint consists of two plates. Table 5-1 shows facts and logical predicates related to the example shown in Figure 5-1.

Figure 5-2 illustrates a semantic net of the t-joint in Figure 5-1. This semantic net can be expanded to include additional information, such as spatial relationships and joining relationships of assembly components.

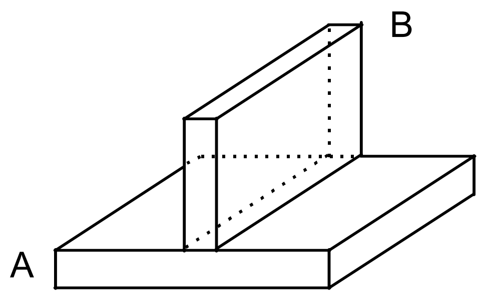


Figure 5-1 T-joint

Table 5-1 Facts and logical predicates

Facts	**Logical predicates**	**Logical predicates of Nilsson (1980)**	
		A	**B**
A is a plate	IS-A (A, plate)	IS-A: plate	IS-A: plate
B is a plate	IS-A (B, plate)	BOTTOM_OF: B	TOP_OF: B
A is on the bottom of B	BOTTOM_OF (A, B)	JOINED: J	JOINED: J
B is on the top of A	JOINED (A, J)		
	JOINED (B, J)		

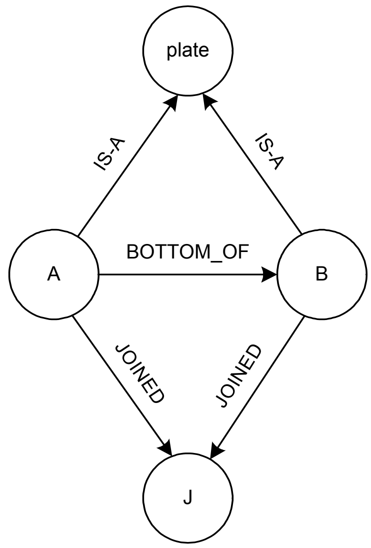
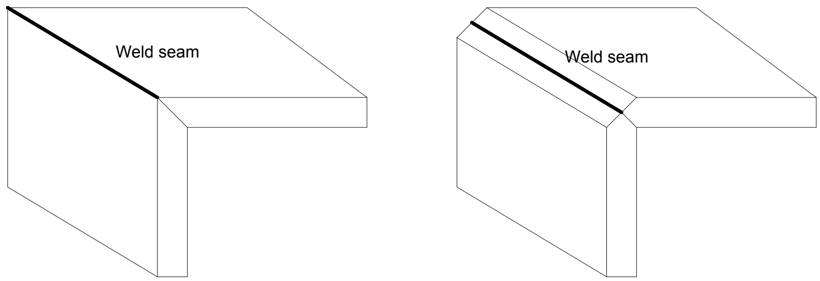


Figure 5-2 A semantic net of Figure 5-1

5.4 The Assembly Design Decision (ADD) Problem

The assembly design decision (ADD) problem occurs when the current assembly design violates the assembly specification, such as maximum allowance in surface straightness and maximum stress. A typical example of the ADD problem may be found in a corner joint. Let's consider a case that a designer specifies a sharp edge of a corner joint as a weld seam (see Figure 5-3-a); and a low weld penetration and high stress level around the weld seam is indicated by an assembly operation analysis. When an assembly design problem is indicated, the designer should make a decision whether to accept the current joint or modify it. If the joint should be modified, then should the current joining method be controlled or another joining method be considered? Assembly design decision making (ADDM) provides an appropriate decision to solve this dilemma.



a) Corner joint with a sharp edge b) Corner joint with a chamfered edge

Figure 5-3 Design alternatives for a weld joint

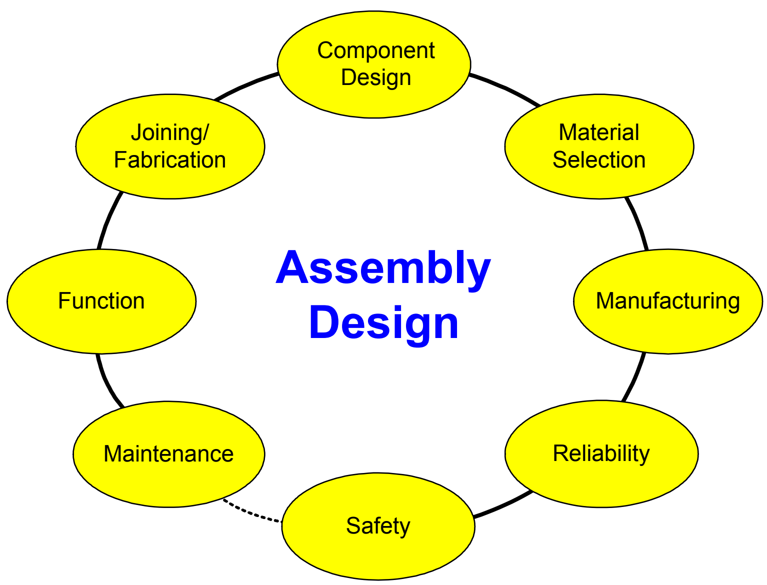
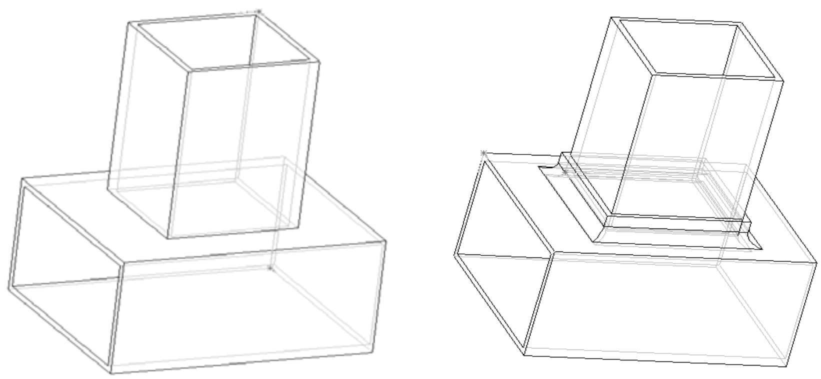


Figure 5-4 Assembly design considerations

The ADD problem has a multi-disciplinary nature. Figure 5-4 shows multidisciplinary considerations during the assembly design process. In assembly design, the interdependent behavior of the interactions of different disciplines is vital to a successful design. When design teams are collaborating on preliminary design tasks, they often try to achieve successful coordination with frequent meetings of the participants (Dettemer 1999, Gobeli *et al*. 1998, Singh and Johnson 1998). However, in an automated designer system it is critical to have an efficient mechanism to make an appropriate assembly design decision, considering multiple disciplines.



a) alternative I b) alternative II

Figure 5-5 Welded assembly between extrusions

Figure 5-5 illustrates a typical example of the interactions in the ADD problem. In this example, two assembly designs (welded extrusion beams) are considered as design alternatives. Alternative I is a simple T-joint and alternative II is an extruded joint. Three disciplines (e.g., manufacturability, assemblability, and material selection) can be considered to decide a final design from the welded extrusion beams. Assembly time ($\widetilde{A}$) is a combination of welding time (A_w), time to fix the base extrusion in a fixture (A_f), alignment time (A_a), and tack welding time (A_t). In addition, cost ($\widetilde{M}$) to manufacture extrusions is a combination of labor cost (M_l), machining time (M_m), tooling cost (M_t), and setup time (M_s). Material can be selected by considering mechanical properties for manufacturability, weldability, material cost, and tolerance. Naturally, interactions among disciplines are involved with ADDM. Table 5-2 shows examples of the interactions. In this work, the potential conflicts between different disciplines are resolved by an ADDM methodology based upon the AHP model, which resolves interactions by communicating with decision makers, and by capturing knowledge of design participants/domain experts.

Table 5-2 Examples of interactions between multiple disciplines in the ADD problem

Interactions	Examples
Manufacturing vs. assembly	▪ Shorter assembly time can cause higher manufacturing cost. $A_a(I) > A_a(II) \Rightarrow M_m(II) > M_m(I),\ M_t(II) > M_t(I),\ M_s(II) > M_s(I)$ ▪ Easy-to-manufacture parts can cause higher assembly time. $\widetilde{M}(I) < \widetilde{M}(II) \Rightarrow A_a(I) > A_a(II),\ A_t(I) > A_t(II)$
Manufacturing vs. assembly vs. material selection	▪ material having good weldability can have poor manufacturability for extrusion process, such as mild steel. ▪ material having good characteristics on extrusion process, such as aluminum alloy, can be sensitive on welding process.

Notations:

FF: form feature; DC: dimensional constraint;

MF: mating feature; JF: joint feature;

AF: assembly feature;

MComp: mating component;

ME: mating element;

JComp: joining component; JE: joining element

JConst: joining constraint

MB: mating bond;

MP: mating pair; MC: mating condition;

ME: mating element;

JM: joining method; GS: groove shape;

GS: groove shape; JComp: joining component;

JE: joining element; JConst: joining constraint;

IA: inter-feature association relation;

J: assembly/joining relation;

P_j^i is a member of part class **P,** $P_j^i \in$ **P**.

FF_{jk} is a member of form feature class **FF,** $FF_{jk} \in$ **FF**.

A_i is an assembly structure class.

J is a member of the assembly operation class ϑ**,** J $\in \vartheta$.

R is a member of the relationship class $\Re$, R $\in \Re$.

DC_r is a member of dimensional constraint class **DC**, $DC_r \in$ **DC**.

RC_{pq} is a relational constraint between FF_{jp} and FF_{jq}, $RC_{pq} \in \{0, 1, 2\}$.

$$RC_{pq} = \begin{cases} \mathbf{0, if}\ FF_{jq} \in FF_{jp} \\ \mathbf{1, if}\ FF_{jp} \in FF_{jq} \\ \mathbf{2, otherwise} \end{cases}$$

MF_r is a member of mating feature class, $MF_r \in$ **MF**, $MF_r \in FF_{jk}$.

JF_r is a member of joint feature class, $JF_r \in$ **JF,** $JF_r \in FF_{jk}$.

:$\rightarrow$ stands for a *belong-to* relation.

$\Leftrightarrow$ stands for an *inter-feature association* relation.

$\otimes$ stands for an *assembly/joining* relation.

5.5 Assembly Relation Model (ARM) and Semantic Net

As described in Chapter 3, the assembly relation model (ARM) developed in this work is employed to capture inner knowledge of assembly design alternatives. In ARM, assembly relations are mathematically defined and represented by using semantic net.

Assembly relations between features as well as between features and parts are defined below. A *belong-to* relation defines relations between a part and a form feature. Figure 5-6 shows a semantic net of a belong-to relation between part, P_j^i, and form feature, FF_{jk}.

Definition 1: *Belong-to* relations

A part P_j^i and a form feature FF_{jk} are said to have a *belong-to* relation,

$$B_{jk}^{(i)}: FF_{jk} :\rightarrow P_j^i,\ k = 1, 2, \ldots, n,$$

if $P_j^i \in \mathrm{A}_i, j = 1, 2, \ldots, m$; and $FF_{jk} \in P_j^i$.

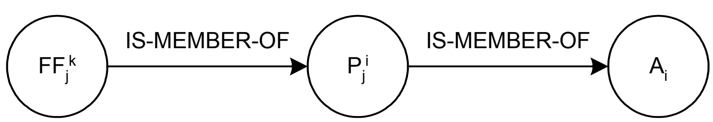


Figure 5-6 Semantic net of the Belong-to relations

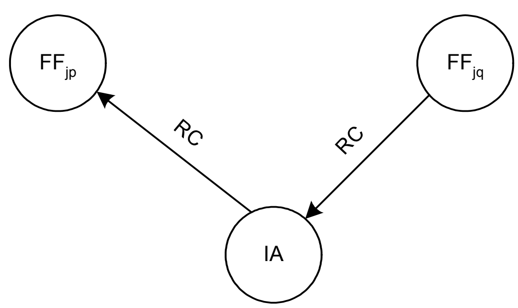
Definition 2: *Inter-feature association* relations

A form feature FF_{jp} and another form feature FF_{jq} are said to have an *inter-feature association* relation,

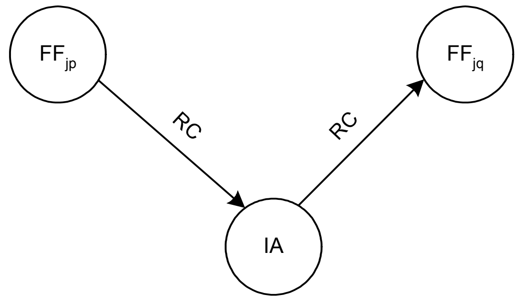
$$I_{pq}^{(j)}: FF_{jp} \Leftrightarrow FF_{jq},\ p = 1,2, \ldots, n,\ q = 1,2, \ldots, l,$$

if $P_j^i \in \mathrm{A}_i, j = 1, 2, \ldots, m; FF_{jp}, FF_{jq} \in \mathbf{FF}; FF_{jp}, FF_{jq} :\rightarrow P_j^i; DC_r$ and RC_{pq} are satisfied, where $r \in IDI_{pq}^{(j)}$; and $IDI_{pq}^{(j)}$ is an index set depending upon this pair, FF_{jp} and FF_{jq} .

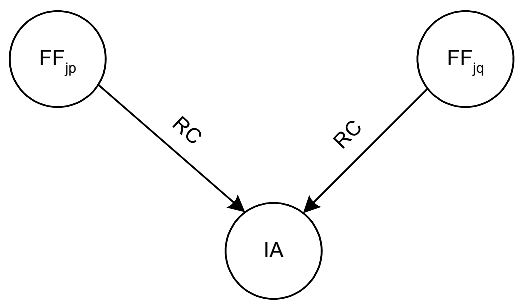
The *inter-feature association* (*IA*) relation represents the relations between form features. The relational constraint (RC_{pq}) stands for the relationship between two form features in the form feature hierarchy. For example, a block (FF_{jq}) can have a blind hole (FF_{jp}) at a certain location. The distance between the coordinates of the block and the blind hole is a dimensional constraint. Since the block form feature contains the hole form feature (the block is a parent class of the hole), their *relational constraint* (RC_{pq}) is 0. Figure 5-7 shows semantic nets representing *RC* and Figure 5-8 illustrates an example of the *inter-feature association* relation.



a) $RC_{pq} = 0$



b) $RC_{pq} = 1$



c) $RC_{pq} = 2$

Figure 5-7 Representation of RC

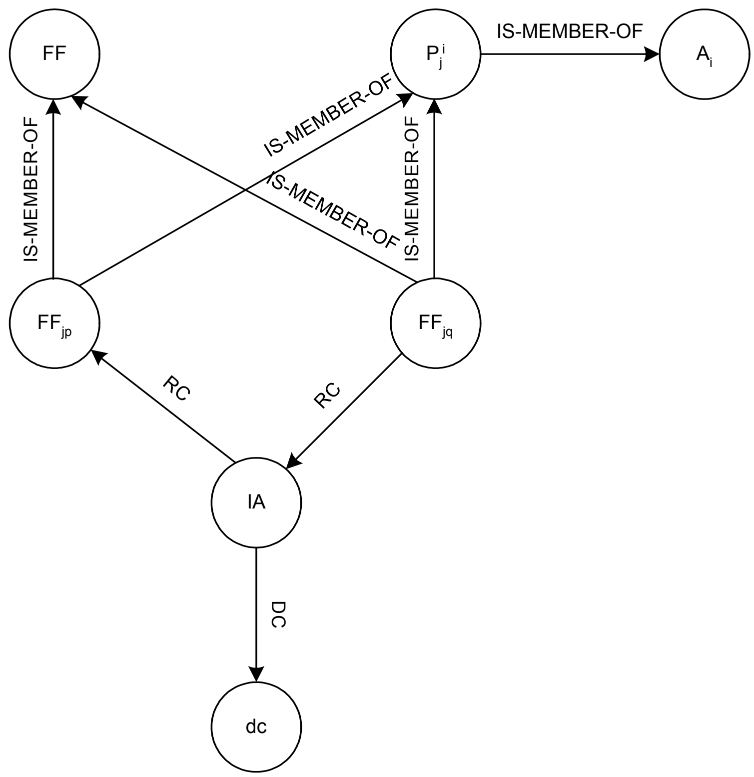


Figure 5-8 Semantic net of an *inter-feature association* relation

Definition 3: *Assembly/joining* relations

A form feature FF_{gp} and another form feature FF_{hq} are said to have an *assembly/joining* relation,

$$\vartheta_{pq}^{(gh)}: FF_{gp} \otimes FF_{hq},$$

if P_g^i and $P_h^i \in \mathbf{A}_i$, $g = 1, 2, \ldots, m_1$, $h = 1, 2, \ldots, m_2$; $FF_{gp} \in P_g^i$, $p = 1, 2, \ldots, l_1$; $FF_{hq} \in P_h^i$, $q = 1, 2, \ldots, l_2$; FF_{gp}, $FF_{hq} \in$ **FF;** FF_{gp}, $FF_{hq} \in \mathbf{J}$; MF_{r1}, $JF_{r2} \in \mathbf{J}$; and DC_{r3} is satisfied, where $r1 \in JMI_{pq}^{(gh)}$, $r2 \in JJI_{pq}^{(gh)}$, and $r3 \in JDI_{pq}^{(gh)}$; and $JMI_{pq}^{(gh)}$, $JJI_{pq}^{(gh)}$, and $JDI_{pq}^{(gh)}$ are index sets depending upon this pair, FF_{gp} and FF_{hq} .

Figures 5-9 and 5-10 show semantic nets for a mating feature and a joint feature. Figure 5-11 illustrates an *assembly/joining* relation between FF_{gp} and FF_{hq}, that is "FF_{gp} and FF_{hq} are assembled subjected to MF_{r1}, JF_{r2}, and DC_{r3}".

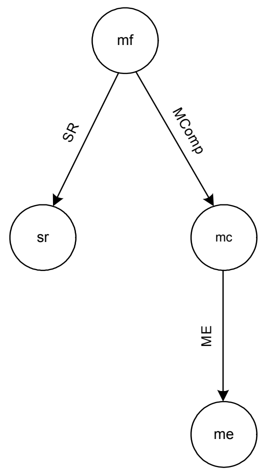


Figure 5-9 Semantic net of a mating feature

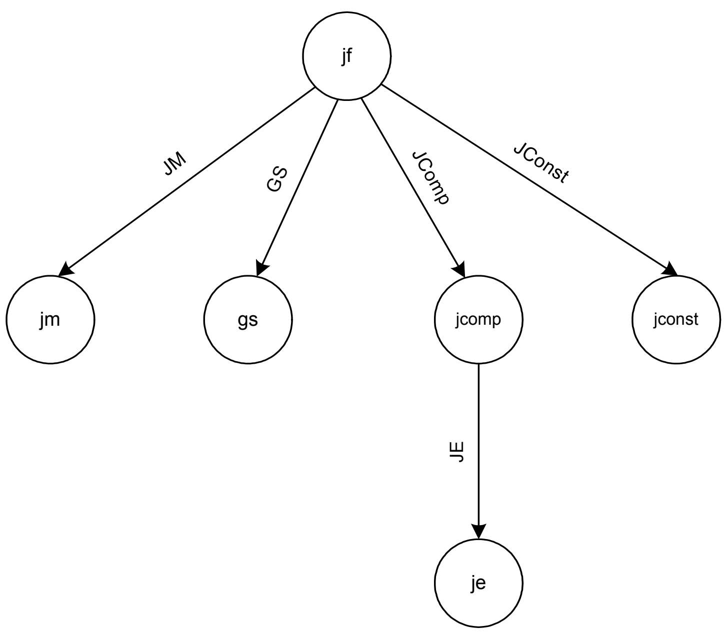


Figure 5-10 Semantic net of a joint feature

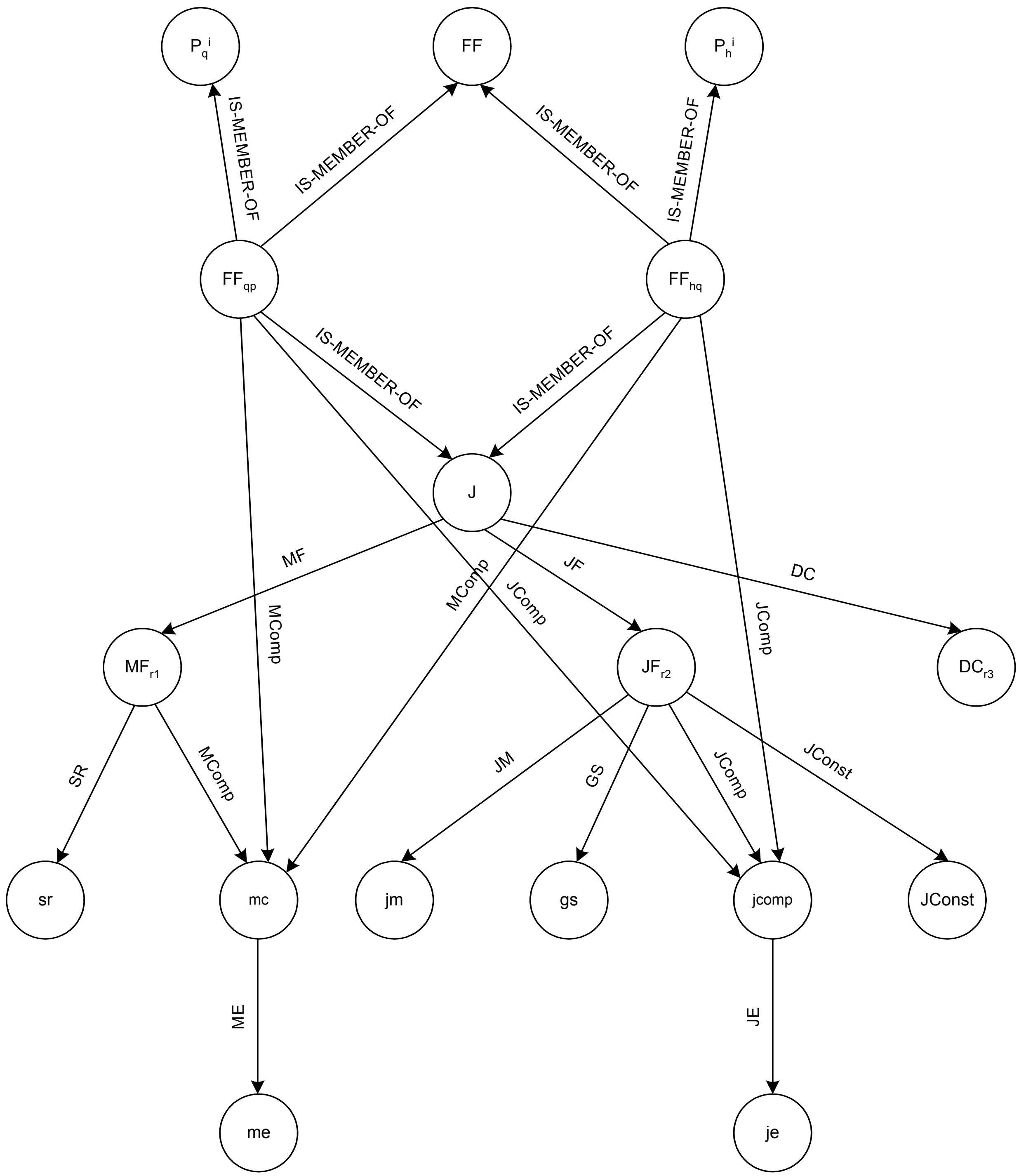


Figure 5-11 Semantic net of the assembly/joining relation

As an example of ARM, consider the simple pin assembly in Figure 5-12. Table 5-3 shows a symbolic representation of an AsD model (ARM) generated for this simple assembly. Here, P_1^1= pin_a; P_2^1 = plate_a; FF_{11} = cylinder_a; FF_{21} = block_b; FF_{22} = hole_b. Note that the designed d.o.f. in the assembly feature (AF) are inferred as {fix}. These d.o.f. are inferred from the specified joining method, such as gas metal arc welding (GMAW). In this table, the mating bonds (MB) are used to represent the engineering relationships on the entire assembly structure. From the AF, two MBs are generated for two aligned spatial relationships. Table 5-4 shows a mathematical ARM. Figure 5-13 illustrates a semantic net representing the assembly relation in Figure 5-12.

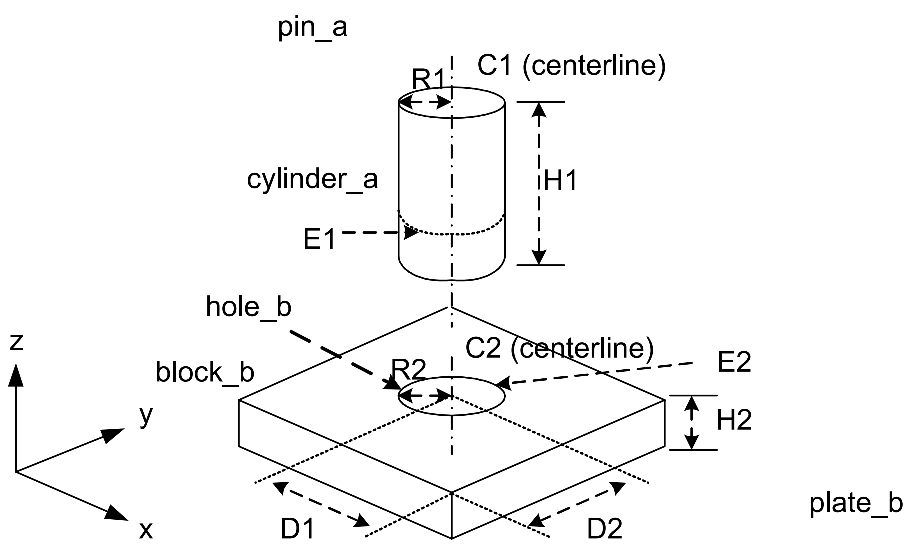


Figure 5-12 An assembly with a pin and a plate with hole

Table 5-3 Symbolic representation of ARM for Figure 5-12

Parts	Features and MB	Representation
P_1^1 & P_2^1 (pin_a & plate _b)	AF	▪ AF_1 = {mating features | mating bonds | joint features | [material] | [designed d.o.f.] | [implied constraints]} = { MF_1, MF_2 | MB_1, MB_2 | JF_1 | [Aluminum Alloy 6061 - T6] | {fix} | $\pm\Delta_1$}
	MF	▪ MF_1 = {S/R, [*mating components* (*mating entities*)]} = {aligned, [FF_{11} (C_1), FF_{22} (C_2)]} ▪ MF_2 = {aligned, [FF_{11} (E_1), FF_{22} (E_2)]}
	JF	▪ JF_1 = {*joining method* | *groove shape* | [*joining components* (*joining entities*)] | [joining constraint]} = {GMAW | single fillet | [FF_{11} (E_1), FF_{22} (E_2)] | [welding_condition], [fixture_location]}
	MB	▪ MB_1 = {mating pair ([mating features (form feature (inter-feature association, dimensional constraint))]) | mating conditions (assembly/joining relations (form features), S/R (transformed geometric constraints), d.o.f., [implied constraints])} = {MP_1 (MF_1, [C_1 (FF_{11} (I (.), {R_1, H_1})), C_2 (FF_{22} (I(I_{12}^2, RC_{12} = 0), {R2, H2}, {D1, D2}))]) | MC_1 ($\vartheta_{12}^{(12)}$(E_1, E_2), aligned ({on_line}, {parallel}), {lin_z::rot_z}, [R_1<=R_2]))} ▪ MB_2 = {MP_2 (MF_2, [E_1 (FF_{11} (I (.), {R_1, H_1})), E_2 (FF_{22} (I(I_{12}^2, RC_{12} = 0), {R2, H2}, {D1, D2}))]) | MC_2 ($\vartheta_{12}^{(12)}$(E_1, E_2), aligned ({on_line}, {parallel}), {rot_z}, [R_1<=R_2]))}

Table 5-4 Mathematical representation of ARM for Figure 5-12

Parts	Assembly engineering relationships
P_1^1 & P_2^1	$\{FF_{11} :\rightarrow P_1^1; FF_{21} :\rightarrow P_2^1; FF_{22} :\rightarrow P_2^1; FF_{21} \Leftrightarrow FF_{22} \mid RC_{12} = 0;\ FF_{11} \otimes FF_{22};$ $IDI_{12}^{(2)} = \{1\}, JMI_{12}^{(12)} = \{1, 2\}, JJI_{12}^{(12)} = \{1\}, JDI_{12}^{(12)} = \{0\}\}$

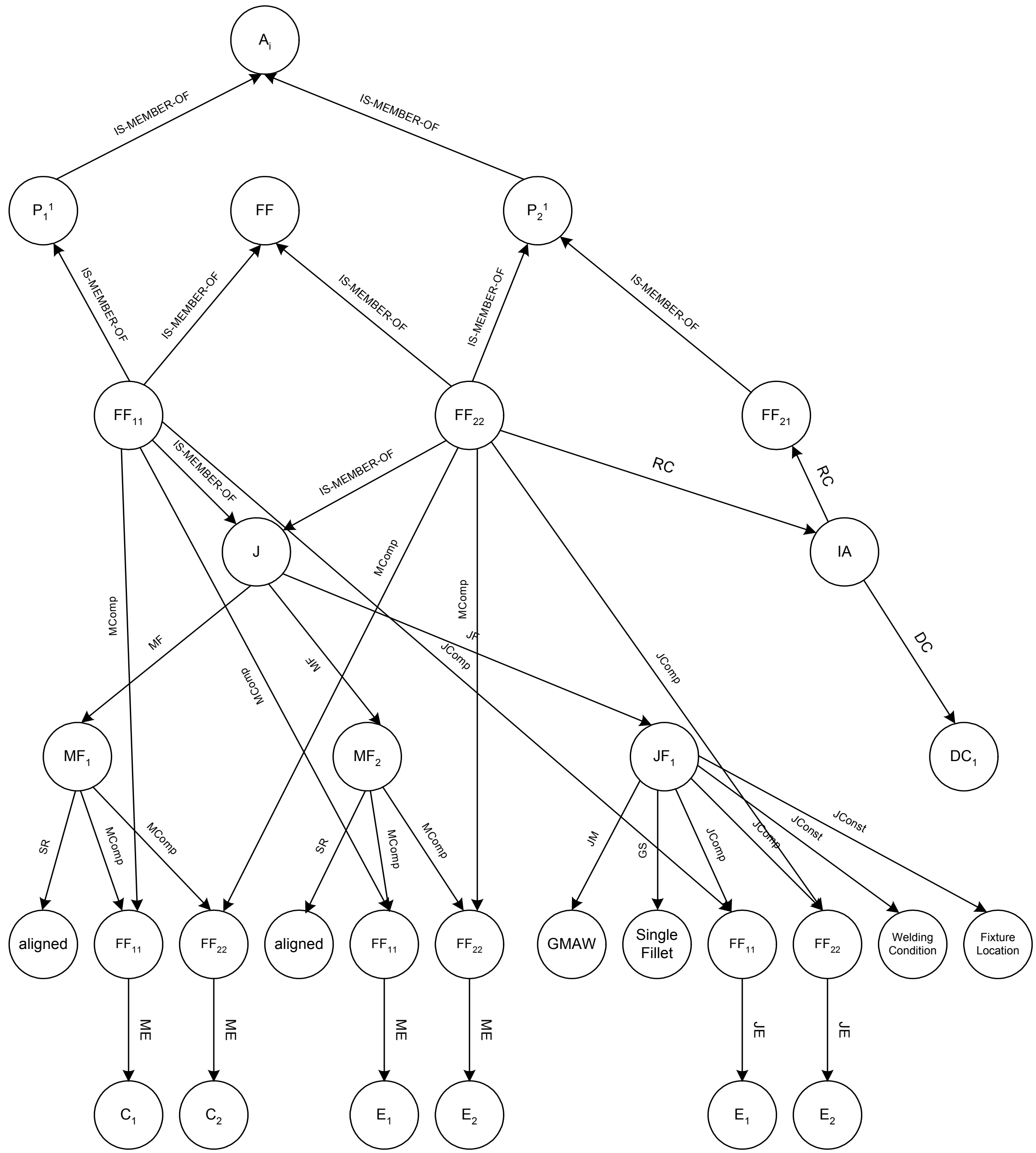


Figure 5-13 Semantic net for Figure 5-12

5.6 Hierarchical Semantic Net (HSN) Model

While AHP is a well-respected multicriteria evaluation method, it has limitations on representing evaluation knowledge and assembly design knowledge, which is inevitable in knowledge-based design decision making like ADDM. In this work, a new hierarchical semantic net (HSN) model integrating the AHP model and semantic net is introduced. Figure 5-14 illustrates the HSN model, which includes predefined criteria (i.e., design, cost, and quality criteria) and assembly design alternatives.

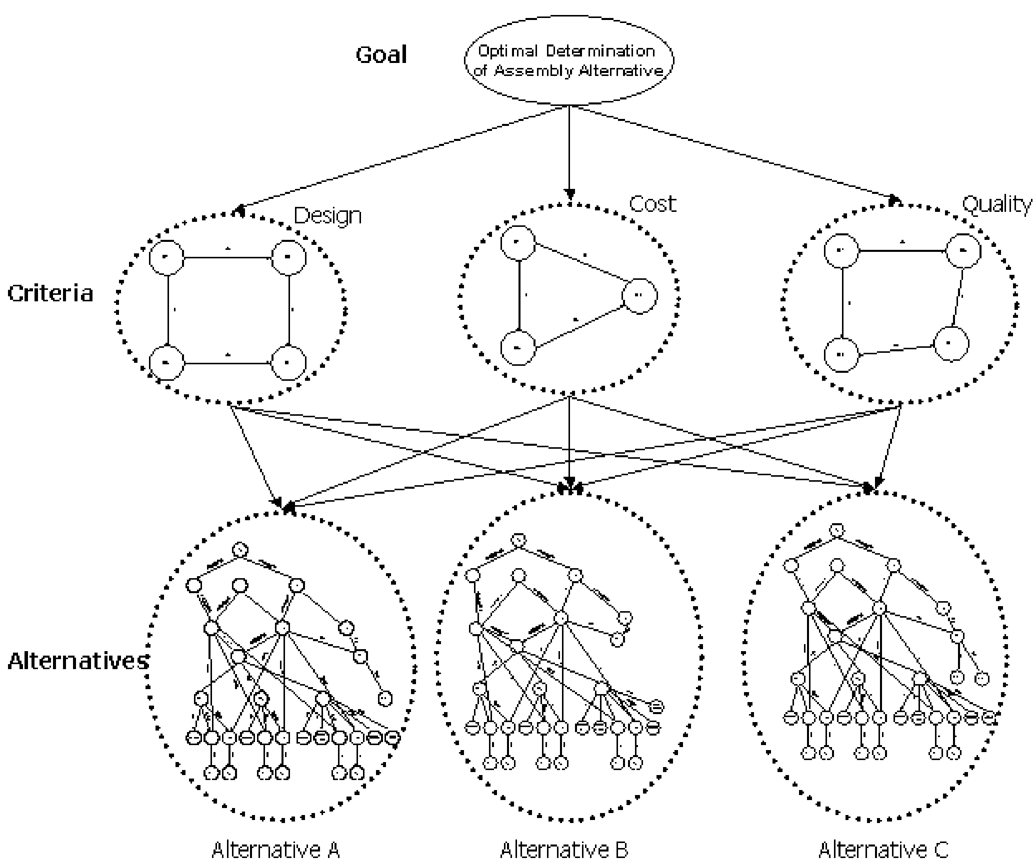


Figure 5-14 Hierarchical semantic net (HSN) model

Generally, ADD criteria have certain relations with the decision-making environment. In AHP, weights between criteria are generally determined by the user's pairwise comparison. Unlike a typical AHP, the weights of criteria can be determined by external rule bases or users. For example, if the demand for the product is high, the cost criterion will have relatively low weight in comparison to other weights (design and quality). As another illustration, if the financial situation of the company is weak and the company is willing to reduce cost, then the cost criterion will have a high weight. Rules can be built based upon domain experts' knowledge.

Table 5-5 Factors affecting ADD criteria

Criterion	Factor	Example
Design	▪ Design intent ▪ Design complexity ▪ Joinability	▪ Difference with the original design intent ▪ Difference with the standard design ▪ Sheet thickness
Cost	▪ Joining complexity ▪ Labor	▪ Difference with the standard joining process ▪ Labor requirement
Quality	▪ Physical effect ▪ Tolerance ▪ Function	▪ Distortion ▪ Fabrication tolerance ▪ Operation environment

Evaluation values of each design alternative are determined from inner knowledge (factors) of each criterion. Table 5-5 lists factors affecting ADD criteria and Figures 5-15 through 5-17 illustrate causal relations of factors and criteria. For example, if design complexity increases, joinability tends to decrease (negative relation). The reduction of joinability decreases design quality (positive relation). From these ADD factors, each design alternative is evaluated and the evaluation values in return are dynamically added to the semantic net of alternatives. By using this method, independence between evaluation knowledge and design knowledge is maintained. It enables the ADDM system to be scalable and extendable.

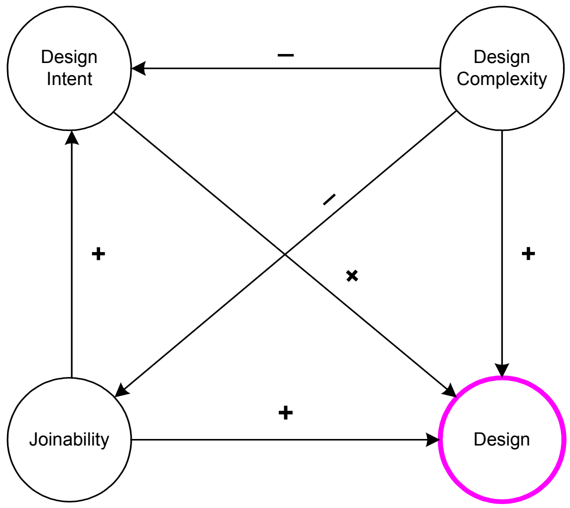


Figure 5-15 Causal model of a design criterion

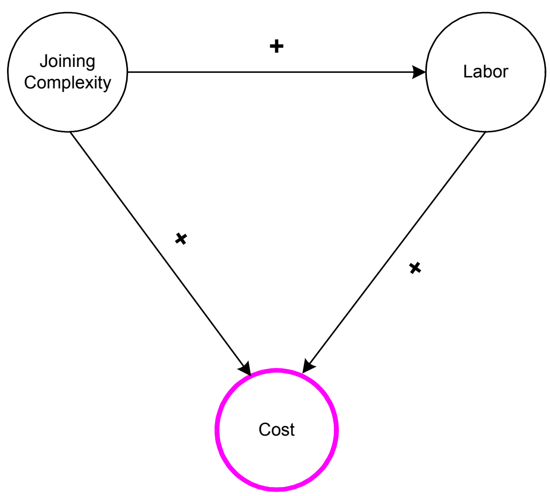


Figure 5-16 Causal model of a cost criterion

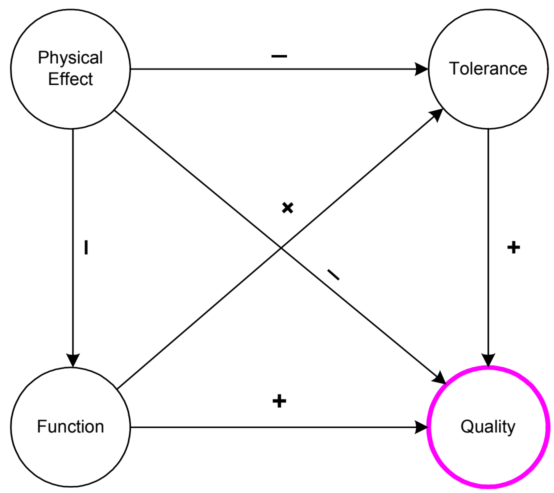


Figure 5-17 Causal model of a quality criterion

5.6.1 Alternative Evaluation Models

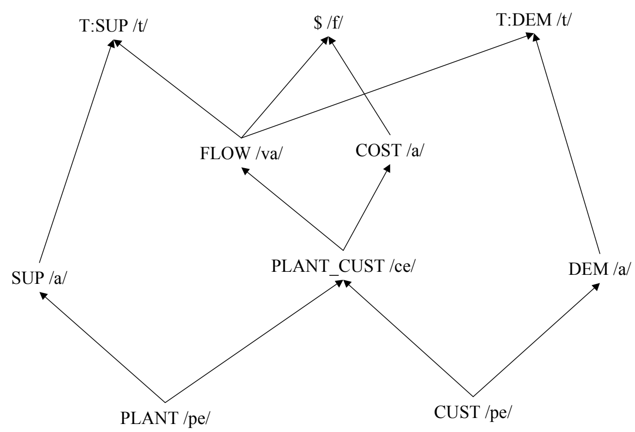
In these sub-sections, examples of alternative evaluation models are explained. A mathematical evaluation model, such as a joining cost model, can be represented by using structural modeling, which has network representations and is easily added to the semantic net of ARM.

5.6.1.1 Structured Modeling (SM). The system of structured modeling (SM) was proposed by Geoffrion (1987) in order to overcome the weaknesses of an earlier technique known as system modeling. This specific proposal maintained that each model could be viewed as a collection of distinct elements. Consequently, SM can be used as a systematic way to classify models and their potential implementations. In his work, the term "schema" indicates a logical representation framework for either an object or other entity that may denote a mathematical model. The model discussed herein serves as a mathematical evaluation model, such as a joining cost model.

Elements are categorized into five types: primitive entity, compound entity, attribute, function, and test. Dependence among these is represented as a directed acyclic graph. The theoretical foundation of SM stems from a rigorous semantic framework that deliberately avoids commitment to representational formalism. Historically, the main application domains of SM have been limited to management science and operations research problems in which a variety of mathematical decision models are required to solve a given decision-making problem. For example, SM has been widely applied to several problem domains, including graph-based modeling (Jones 1992), integration with database systems (Dolk 1988), language-directed editors (Vicuña 1990), object-oriented systems (Muhanna 1993), and model integration (Dolk and Kottemann 1993, Gagliardi and Spera 1995).

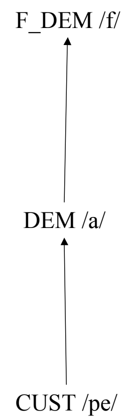
A model schema of SM is primarily defined in terms of genera that organize a set of data elements, which are based on definitional similarity. There are six types of data elements. The first is primitive entity (/pe/) that exists in nature. The second is compound entity (/ce/) which references other entities that are already defined and therefore do not require value. This is followed by attribute (/a/) and associates a certain property and value with an entity or compound entity. The variable attribute (/va/) resembles decision variables in an mathematical model. Function (/f/) allows its elements to have a value that depend on those of other functions or attributes. Finally, the test (/t/) is a function in which the value is fixed to binary values.

Each element has a calling sequence that identifies the other elements that are directly referenced. The calling sequence captures the cross-references among the model elements and can be directly derived from the graphical representation. The genus graph is one of the graphical representations of SM that captures the defined dependencies among the genera, while suppressing the details of each model instance. In the genus graph, each node indicates genera by element type, and the segment line indicates the calling sequence. Figure 5-18-a illustrates an example of a genus graph for the SM types. Figure 5-18-b presents another example of a genus graph for the SM types from the perspective of demand forecasting. As shown in these examples, the SM can be applied to simple mathematical models as well as sophisticated optimization models.



$$
\begin{aligned}
&\textit{Minimize} \\
&z = \sum_i \sum_j Cost_{ij} Flow_{ij} (\$ / f /) \\
&s.t \\
&\sum_i Flow_{ij} \geq DEM_j (T : DEM / t /) \\
&\sum_j Flow_{ij} \leq SUP_i (T : SUP / t /) \\
&\forall i, j Flow_{ij} \geq 0 \\
&i : PLANT \\
&j : CUST
\end{aligned}
$$

(a) For transportation model



$$F\_DEM = f(DEM_i)$$
$$i : CUST$$

(b) For demand forecasting model

Figure 5-18 Sample SM genus graphs

The Structured Modeling Language (SML), a non-graphical representation of SM, is a text-based notation for the genus graph. SML consists of paragraphs with a formal section for model specification and an informal section for model documentation. SML is executable, and an optimal solution can be acquired by transforming a genus graph into SML code.

5.6.1.2 Joining Cost Model. Selecting the most appropriate manufacturing process in terms of technological feasibility is one of the most important decision-making tasks; failure to get it right normally results in assemblies that are of variable quality and/or expensive to make.

In recent years a number of research groups have concentrated on the design/manufacturing interface; processes and systems for cost estimation are under development in areas, including machining (Boothroyd *et al.* 1994), powder metallurgy (Fume and Knight 1989), die casting

(Woodward and Corbett 1989) and broader techniques providing DFM (design for manufacture) and cost-related information for the designer (Shea et al. 1989, Zenger and Boothroyd, 1989, Allen and Ashley 1990).

The cost model is logically based on material volume, labor, and processing considerations. The process cost is related to the design. The process cost can be determined by considering the characteristics of the joining processes. Material costs are calculated taking into account the transformation of material to yield the final form. A general cost model for joining (J^i_C) can be formulated as:

$$J^i_c = V^i C^i_{mt} + T^i_p(C^i_l + C^i_p), \qquad \text{(Eq. 5-1)}$$

where

V^i = volume of material required in order to perform the joining process *i*

C^i_{mt} = cost of the material per unit processed

T^i_p = processing time for joining process *i*

C^i_l = labor cost for the joining process *i*

C^i_p = processing cost for the joining process *i*

The above cost model can be represented in a SM genus graph as shown in Figure 5-19.

$$JCOST^i = VOL \cdot MCOST^i + PTIME \cdot LCOST + PTIME \cdot PCOST^i \; (/f/)$$

$$= f_m \; (/f/) + f_l(/f/) + f_p \; (/f/)$$

i: JOIN

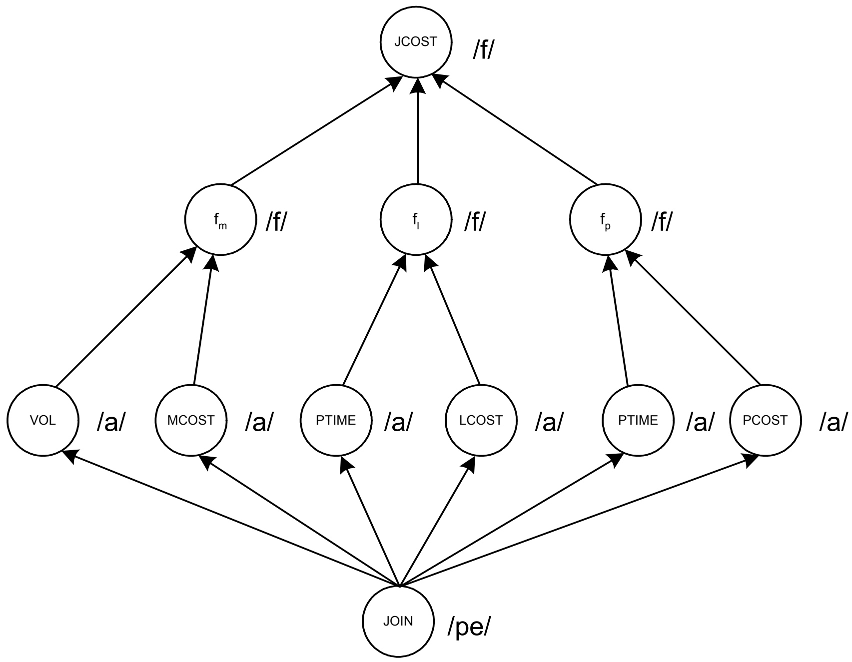


Figure 5-19 SM genus graph of the assembly design cost model

Each joining process has unique characteristics. Considering the characteristics, its cost model should be defined. In this work, practical cost models for arc welding and riveting are used.

The American Welding Society (Welding Workbook 2001) introduced the following equation to estimate direct arc welding cost.

$$C_{twp} = C_w + C_c, \qquad \text{(Eq. 5-2)}$$

where C_{twp} is total welding processing cost, C_w is total cost of the weld, and C_c is total cost of consumables, such as electrode/wire, SAW flux, and gas.

$$C_w = (C_g + C_p + C_m + C_l + C_o) \times W \times N, \quad \text{(Eq. 5-3)}$$

where

C_g is gas cost per unit weight of deposited metal (\$/g) and its equation is $G \times F/D$. A detailed description about parameters can be found in Table 5-6;

C_p is power cost per unit weight of deposited metal and its equation is $P \times V \times (A/1000) \times D$;

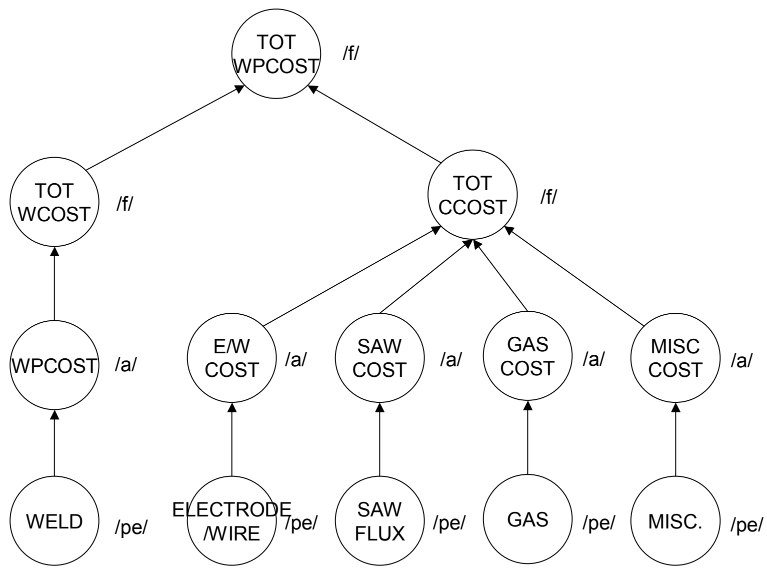
C_m is cost of materials per unit weight of deposited metal and its equation is M/E;

C_l is labor rate per unit weight of deposited metal and its equation is $L \times K/(D \times 100)$;

and C_o is overhead cost per unit weight of deposited metal and its equation is $O \times K/(D \times 100)$.

Table 5-6 Parameters for welding cost estimate

Parameters	Notation	Unit
amperes	A	A
volt	*V*	V
deposition rate	*D*	g/h
flow rate	*F*	m^3/h
unit cost of gas or flux by volume	*G*	$\$/m^3$
deposition efficiency	*E*	%
operator factor	*K*	%
labor rate	*L*	$/h
cost of materials	*M*	$/g
length of specified weld	*N*	m
overhead rate	*O*	$/h
power cost	*P*	$/kWh
total weight of weld metal	*W*	g/m



TOT_WPCOST: total welding processing cost
TOT_WCOST: total cost of weld
TOT_CCOST: total cost consumables
WPCOST: welding processing cost
E/WCOST: cost of electrode/wire
SAWCOST: cost of SAW flux
MISCCOST: cost of miscellaneous consumable goods

Figure 5-20 SM genus graph of the welding cost model

Figure 5-20 shows a SM genus graph of the welding process cost model. This graph can be integrated to the semantic net of ARM and translated to an executable code (Appendix D.1).

Estimating riveting cost is much simpler than welding. The cost model consists of labor cost and rivet cost. The riveting cost is mainly affected by the number of rivets to apply.

$$C_{trp} = C_r + C_{cr}, \qquad \text{(Eq. 5-4)}$$

where C_{trp} is total rivet processing cost, C_r is labor cost, and C_{cr} is total cost of rivets applied.

$$C_r = L \times T_r, \qquad \text{(Eq. 5-5)}$$

where L is labor rate (\$/h); T_r is total riveting time and its equation is N_r/K; N_r is the number of rivets applied; and K is operator factor (%).

5.6.1.3 Design Model. A design can be evaluated by determining how much more expensive the design will be to assemble components with more demanding features than the "ideal design." Swift and Booker (1997) considered shape complexity to estimate manufacturing cost considering a design. In this work, a design complexity model is used as a design criterion evaluation model. The design complexity model is based upon a design complexity index (*Dc*). *Dc* is defined as relative cost associated with assembling components of varying geometrical complexity with different joining methods. The design complexity index is obtained by using a form feature-based classification system, which enables the important design/assembly issues to be taken into account. The basic shape category is divided into three classes (Table 5-7). *Class A* is a set of solids generated by revolution. *Class C* is a set of flat or thin-walled sections with specific contours; Thin bars belong to *Class B*. Generally, component geometry closed to a contact region is an important consideration for assembly design. A component shape has a relationship with a joining method. For example, if a *Class A* component and a *Class B* component are assembled by welding (assembly I), this design will have a higher complexity than assembly between *Class B* components (assembly II). In this case, *Dc* of assembly I will be higher than *Dc* of assembly II. The *Dc* can be determined by domain experts. Tables 5-8 and 5-9 show examples of design complexity indexes for welding and riveting, which was determined based upon the discussion with domain experts. The design complexity index is given a value between 1 to 10 and a low value means that the assembly configuration is relatively simpler than an assembly with a high value with regard to joining configurations and methods. For riveting, the number of riveting adds some degrees of design complexity. Thus, in this work each rivet adds a value of “0.05” to the basic *Dc* value. For example, if two thin wall sections (*Class C*) are joined with two rivets, the *Dc* of the assembly will be “1.6.”

Table 5-7 Basic shape category (Swift and Booker 1997)

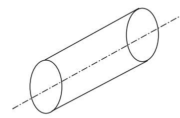
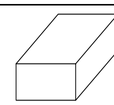
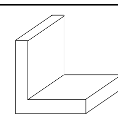
Class	Description	Geometry
A	Solid of revolution	
B	Prismatic solid	
C	Flat or thin-walled section	

Table 5-8 Design complexity index for welding

Class		A (Solid of revolution)		B (Prismatic solid)	C (Flat or thin-walled section)
		Planer face	Cylindrical face		
A (Solid of revolution)	Planer face	1.2	3	1.5	2
	Cylindrical face	3	10	10	10
B (Prismatic solid)		1.5	10	1	1.5
C (Flat or thin-walled section)		2	10	1.5	1.5

Table 5-9 Design complexity index for riveting

Class		A (Solid of revolution)		B (Prismatic solid)	C (Flat or thin-walled section)
		Planer face	Cylindrical face		
A (Solid of revolution)	Planer face	8	10	2	3
	Cylindrical face	10	10	7	4
B (Prismatic solid)		2	5	1	1.5
C (Flat or thin-walled section)		3	7	1.5	1.5

5.6.1.4 Physical Effect Simulation Model. Predicting the physical effects of joining requires specific analysis methodology and procedures. For example, a thermo-structural analysis is used to understand the thermal and structural behavior of the welding operation. As another example, structural analysis is employed to predict various structural phenomena of the riveting operation. In this work, the Virtual Assembly Analysis (VAA) method is used to simulate physical effects of joining and the obtained results are used as physical effect values.

5.6.2 Knowledge-Based Dynamic HSN Model

As described above, design alternatives are evaluated based upon the evaluation models and the obtained values in return are dynamically added to the semantic net of alternatives. Figure 5-20 illustrates the semantic net example (Figure 5-13) including evaluation values. The obtained evaluation values, such as JOIN, PEFFECT, and DCOMPLEXITY, are dynamically added to the object j (joining method). As shown in Figure 5-21, the semantic net based ARM can easily include the obtained evaluation values, such as joining cost, design complexity, and physical effects. These values are used to determine weights of each alternative in the context of each criterion. After the weights are decided, the values are discarded and a new evaluation process is triggered. This method allows independence between evaluation knowledge and design knowledge. In other words, the addition of new criterion will not affect the evaluation of design alternatives. It also enables the ADDM system to be scalable and extendable.

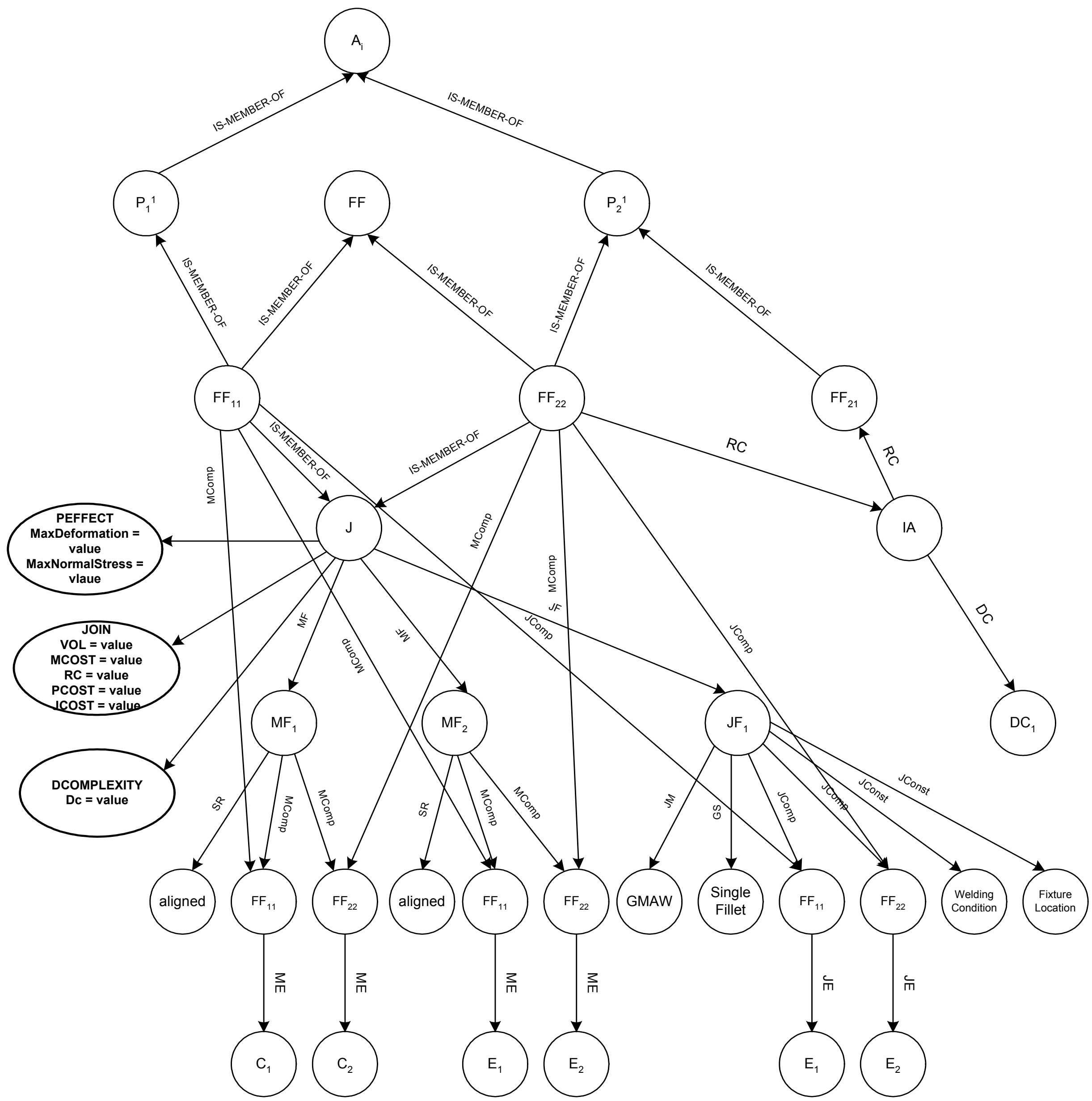


Figure 5-21 Semantic net including evaluation values

5.7 The Assembly Design Decision Making (ADDM)

The purpose of ADDM is to propose assembly alternatives to the designer by considering assembly implication information and assembly/joining knowledge. A hierarchical semantic net (HSN) model is introduced as a core model to represent evaluation knowledge and assembly design knowledge. Figure 5-22 shows the overall concept of the ADDM and detailed procedures are described below.

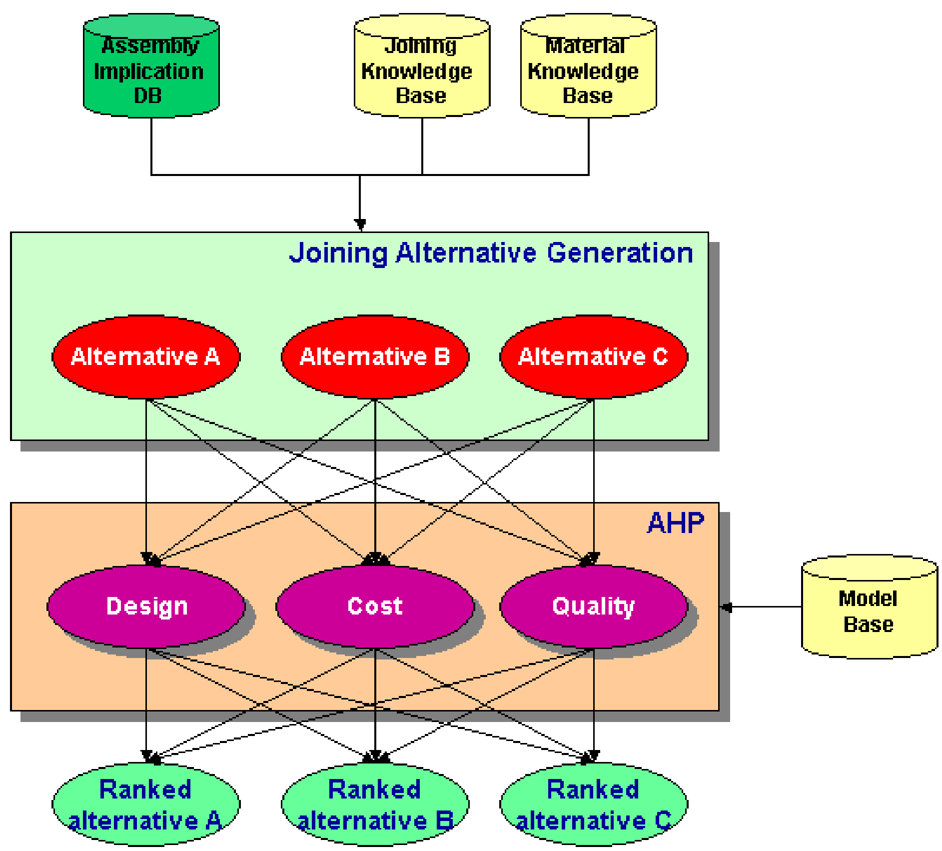


Figure 5-22 Assembly design decision making

ADDM Procedures

STEP 1: Indicate environmental change, such as violation of assembly specification.

STEP 2: Generate alternatives based on assembly/joining knowledge and material knowledge.

STEP 3: Construct a set of pairwise comparison matrices for the criteria. The elements of the matrix can be rules obtained from the external rule base or the decision maker. It represents any factor that may affect evaluation of the AHP model (e.g., user characteristics, financial condition, market situation). Based upon the rules, make all the pairwise comparisons. Use a fundamental scale of absolute numbers from 1 to 9 to indicate the relative dominance with respect to a given property of one criterion over another used as the unit of the paired comparison in a cluster of homogeneous elements.

STEP 4: Construct a set of pairwise comparison matrices for the alternatives. For each criterion, evaluate alternatives using evaluation models and update the evaluation values to alternatives. Make all the pairwise comparisons by comparing the evaluation values.

STEP 5: Check for consistency of the comparisons.

STEP 6: Synthesize the comparisons to get the priorities of the alternatives with respect to each criterion and the weight of each criterion with respect to the goal.

STEP 7: The obtained local priorities and weights are confirmed by the decision maker.

STEP 8: Local priorities are then multiplied by the weights of the respective criterion and the results are summed up to get the overall priority of each alternative.

STEP 9: Determine the optimal alternative.

5.7.1 Obtaining Weights for Each Criterion

In the ADDM, weights for each criterion can be obtained by the following methods. Let C_1, C_2, …, C_n be the set of criteria. The pairwise comparison on the criteria, C_i and C_j, are represented by an n-by-n matrix.

$$A = [a_{ij}] \text{ , where } i, j = 1, 2, \ldots, n. \qquad \text{(Eq. 5-6)}$$

The entities a_{ij} are defined by the external knowledge base or by the decision maker. Here, if $a_{ij} = \alpha$, then $a_{ji} = 1/\alpha$, $\alpha \neq 0$. If C_i is judged to be of equal relative importance as C_j, then $a_{ij} = 1$, $a_{ji} = 1$; in particular, $a_{ii} = 1$ for all i. Thus, the matrix A has the form

$$A = \begin{bmatrix} 1 & a_{12} & \cdots & a_{1n} \\ \frac{1}{a_{12}} & 1 & & a_{2n} \\ \vdots & \vdots & & \vdots \\ \frac{1}{a_{1n}} & \frac{1}{a_{2n}} & \cdots & 1 \end{bmatrix} \qquad \text{(Eq. 5-7)}$$

When a_{ij} is determined by the external knowledge base, a_{ij} can be represented in Horn sentences (Russell and Norvig 1995) as follows. Here, EQ is an *equal* function. For example, the expected quality level of an assembly is high, quality is more important than cost; $a_{ij} = 3$, where i = quality criterion and j = cost criterion.

$$P_1 \wedge P_2 \wedge \ldots \wedge P_n \Rightarrow EQ(a_{ij}, \alpha) \qquad \text{(Eq. 5-8)}$$

Weights for each criterion can be obtained from the pairwise comparison matrix A by using the following method (Winston 1993). Suppose there are n criteria. Let w_i = the weights given to criteria i. Let's suppose the pairwise comparison is done consistently. Then, the pairwise comparison matrix should be of the following form:

$$A = \begin{bmatrix} \frac{w_1}{w_1} & \frac{w_1}{w_2} & \cdots & \frac{w_1}{w_n} \\ \frac{w_2}{w_1} & \frac{w_2}{w_2} & & \frac{w_2}{w_n} \\ \vdots & \vdots & & \vdots \\ \frac{w_n}{w_1} & \frac{w_n}{w_2} & \cdots & \frac{w_n}{w_n} \end{bmatrix} \qquad \text{(Eq. 5-9)}$$

For example, suppose that $w_1 = \frac{1}{2}$ and $w_2 = \frac{1}{6}$. Then criterion 1 is three times as important as criterion 2, so $a_{12} = \frac{w_1}{w_2} = 3$.

Now suppose that a consistent pairwise comparison matrix A of the form (Eq. 5-9) is given. The vector $\boldsymbol{w} = [w_1 \ w_2 \ \ldots \ w_n]$ can be recovered from A. Consider the system of n equations

$$A\boldsymbol{w}^T = \Delta \boldsymbol{w}^T, \quad \text{(Eq. 5-10)}$$

where Δ is an unknown number and $\boldsymbol{w}^T$ is an unknown n-dimensional column vector. For any number Δ, equation (5-10) always has the trivial solution $\boldsymbol{w} = [0 \ 0 \ \ldots \ 0]$. It can be shown that if A is the pairwise comparison matrix of a perfectly consistent decision maker (that is, if A is of the form (Eq. 5-9)) and we do not allow $\Delta = 0$, then the only nontrivial solution to (Eq. 5-10) is Δ

$= n$ and $\boldsymbol{w} = [w_1 \ \ w_2 \ \ \ldots \ \ w_n]$. This shows that for a consistent pairwise comparison, the weights w_i can be obtained from the only nontrivial solution to (Eq. 5-9). Now suppose that the pairwise comparison is not perfectly consistent. Let Δ_{max} be the largest number for which (Eq. 5-9) has a nontrivial solution (call this solution $\boldsymbol{w}_{max}$). If the comparisons do not deviate very much from perfect consistency, we would expect Δ_{max} to be close to n and $\boldsymbol{w}_{max}$ to be close to $\boldsymbol{w}$. Saaty (1996) verified that this intuition is indeed correct and suggested approximating $\boldsymbol{w}$ by $\boldsymbol{w}_{max}$. Saaty also proposed measuring the decision maker's consistency by looking how close Δ_{max} is to n. In what follows, a simple method that can be used to approximate Δ_{max} and $\boldsymbol{w}_{max}$ and an index of consistency are addressed.

To approximate $\boldsymbol{w}_{max}$, we use the following two-step procedure:

STEP 1: For each of A's columns, divide each entry in column i of A by the sum of the entries in column i. This yields a new matrix (call it A_{norm}, for normalized) in which the sum of the entries in each column is 1.

STEP 2: To find an approximation of $\boldsymbol{w}_{max}$ (to be used as our estimate of $\boldsymbol{w}$), estimate w_i as the average of the entries in row i of A_{norm}.

Intuitively, each entry in row i shows a relative importance between two criteria. For example, w_2/w_3 says the weight given to criterion 2 comparing to criterion 3. Thus, we can say that the average to obtain w_i represents in some way a measure of the total weight attached to criterion i.

5.7.2 Checking for Consistency

In this work, the following four-step procedure (Winston 1993) is used to check for the consistency of the pairwise comparisons.

STEP 1: Compute Aw^T.

STEP 2: Compute

$$\frac{1}{n}\sum_{i=1}^{i=n}\frac{i\text{th entry in } Aw^T}{i\text{th entry in } w^T}$$

STEP 3: Compute the consistency index (CI) as follows:

$$\text{CI} = \frac{(\text{STEP 2 result}) - n}{n-1}$$

STEP 4: Compare CI to the random index (RI) for the appropriate value of *n*, shown in Table 5-10.

For a perfectly consistent decision maker, the *i*th entry in $A\mathbf{w}^T = n$ (*i*th entry of $\mathbf{w}^T$). This implies that a perfectly consistent pairwise comparison has CI = 0. The values of RI in Table 5-10 give the average value of CI if the entries in *A* were chosen at random, subject to the constraint that all diagonal entries must equal 1 and $a_{ij} = \dfrac{1}{a_{ji}}$.

If CI is sufficiently small, the pairwise comparisons are probably consistent enough to give useful estimates of the weights for the criterion. If CI/RI < 0.10, the degree of consistency is satisfactory, but if CI/RI > 0.10, serious inconsistencies may exist, and AHP may not yield meaningful results.

Table 5-10 Values of the random index (RI) (Winston 1993)

n	RI
2	0
3	0.58
4	0.90
5	1.12
6	1.24
7	1.32
8	1.41
9	1.45
10	1.51

5.7.3 Obtaining Local Priorities for Alternatives

Similar to obtaining weights for criteria, local priorities of an alternative are determined based upon the evaluation models with respect to each criterion. Before continuing, the types of criteria should be considered. There are two different types of criteria, that is, criteria with positive relations (*type I*) and with negative relations (*type II*). The types are determined by evaluation models. An typical example of type II is cost criteria. If the cost evaluation value of an alternative is high, then the alternative is not preferable and has low local priority. Element, a_{ij}, of a pairwise comparison matrix is obtained by the following steps.

STEP 1: Calculate ρ to compare evaluation values of alternative i and alternative j.

$$\rho = 10 \cdot \left(\frac{m_i - m_j}{m_{\max} - m_{\min}} \right),$$

where $m_{\max}$ and $m_{\min}$ are maximum and minimum evaluation value between alternatives.

STEP 2: Obtain integer numbers, ρ_r

If $\rho \geq 0$, obtain ρ_r by rounding up. To follow AHP's fundamental scale, 1 to 9, if $\rho_r > 9.5$, ρ_r is 9 and if $\rho_r < 0.5$, ρ_r is 1.

If $\rho < 0$, obtain ρ_r by rounding down. For the fundamental scale, if $\rho_r < -9.5$, ρ_r is -9 and if $\rho_r > -0.5$, ρ_r is -1.

STEP 3: Obtain a_{ij} from ρ_r

For type I:

If $\rho_r > 0$, $a_{ij} = \rho_r$ and $a_{ji} = 1/\rho_r$

If $\rho_r < 0$, $a_{ji} = \rho_r$ and $a_{ij} = 1/\rho_r$

For type II:

If $\rho_r > 0$, $a_{ij} = 1/\rho_r$ and $a_{ji} = \rho_r$

If $\rho_r < 0$, $a_{ji} = 1/\rho_r$ and $a_{ij} = \rho_r$

Sometimes, multiple models are required to evaluate each alternative with respect to a certain criterion. For example, maximum stress and maximum displacement can be used to evaluate each alternative's quality. In this work, the following weighted sum method is used to obtain an aggregated evaluation value, $m^a{}_i$ of alternative i.

$$m^a{}_i = \alpha_1 . m_{1,i} + \alpha_2 . m_{2,i} + \ldots + \alpha_r . m_{r,i} = \sum_{p=1}^{r} \alpha_p m_{p,i} \ , \forall i, \quad \text{(Eq. 5-11)}$$

where r is the number of evaluation models applied. α_p is a weight normalizing $m_{p,i}$ and

$\alpha_p = \dfrac{1}{\sum_{i=1}^{k} m_{p,i}}$, $\forall p$; k is the number of alternatives.

5.8 Implementation

Figure 5-23 illustrates the architecture of the AsA engine. The VAA tool predicts physical effects of assemblies and thus, potential assembly problem can be indicated. Once the assembly problem is shown, the AsA engine generates AsD alternatives based upon external knowledge bases, including joining and material knowledge bases. It also generates weights of criteria and local priorities of alternatives. To obtain quality evaluation value, that is, stress and displacement, the AsA engine communicates interactively with the VAA tool. The obtained alternatives, weights, and local priorities are inputs for the AHP tool. The weights and local priorities are confirmed by the decision maker before starting the AHP process. The AHP tool proposes overall priorities of the AsD alternatives and the decision maker makes design decision based upon the priorities.

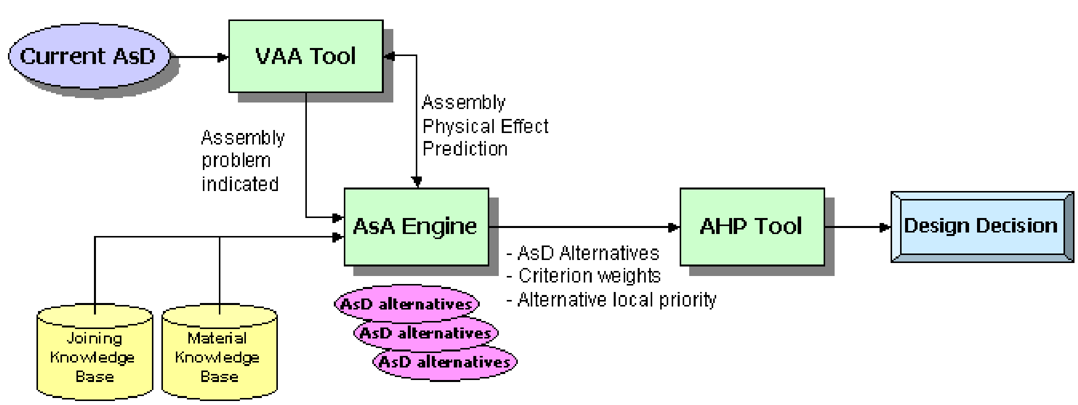


Figure 5-23 Architecture of the AsA engine

The AsA engine checks whether the specified joining method is feasible within nominal geometry. Three important properties common to the analysis of any assembly are stiffness, strength, and deflection of deformed shape. Stiffness is a measure of the force required to produce a given deflection, and strength refers to the force, or force intensity, necessary to cause failure. A criterion for failure is required in order to determine the strength of a structure, and this depends upon the particular application. A well-known criterion is that failure can be defined when a stress (internal force intensity) exceeds the yield stress of the material (this is called *material criterion*). Another is that failure can mean excessive displacements, which occur during buckling (this is called *tolerance criterion*). The stiffness and strength of a structure depend on its geometrical configuration, connections, and the stiffness and strength of the material from which it is made.

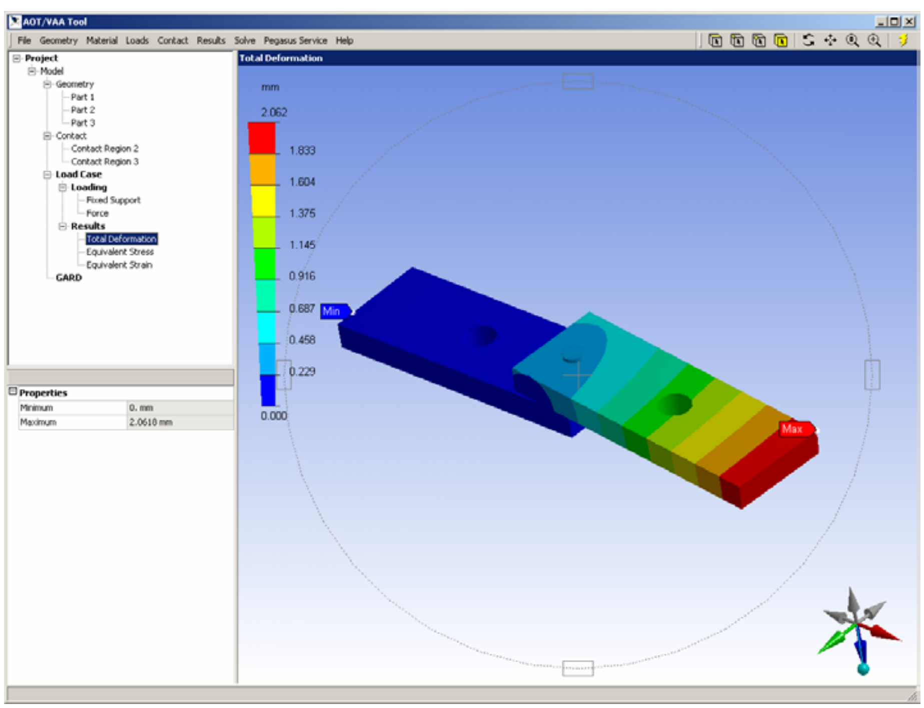


Figure 5-24 VAA result of a rivet joint

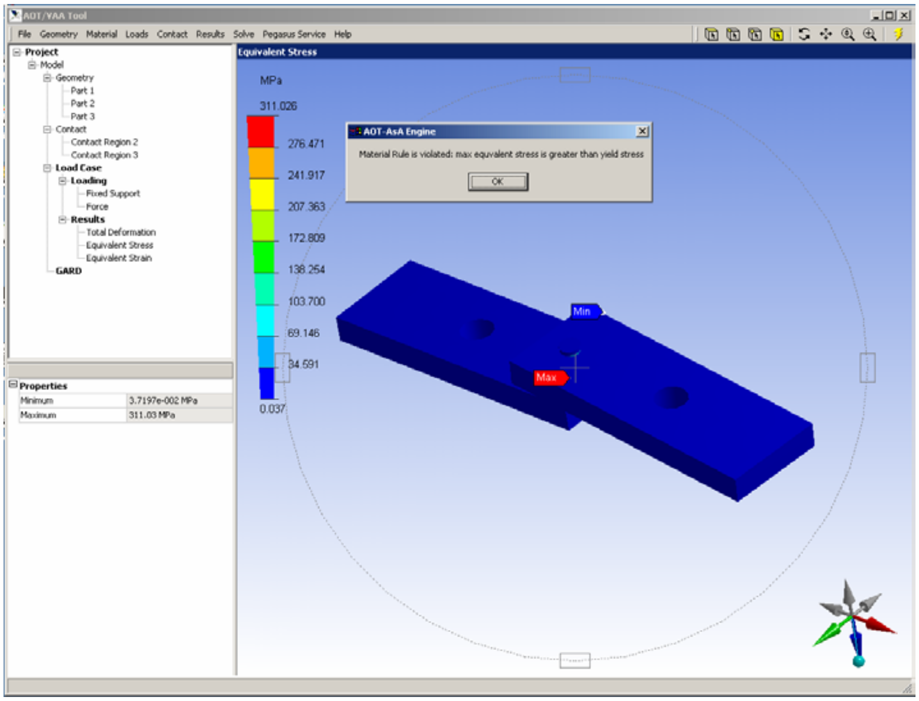


Figure 5-25 AsA engine indicating an assembly problem

A simple assembly with a rivet joint is used to explain the functionality of the AsA engine. Let's assume that the two aluminum plates are joined by a single aluminum rivet in the initial design (Figure 5-24). VAA predicts a potential assembly problem (Figure 5-25) violating the material criterion. It illustrates a case of that max equivalent stress beyond yield stress of the aluminum-alloy 6063 (280 MPa); the max equivalent stress is 311.026 MPa. Like this case, once any AsD problem is found, the AsA engine generates relevant AsD alternatives and guides the designer to make a proper AsD decision. In this work, three alternatives are generated by the rules in Table 5-11. Here, σ is stress; σ_{yield} is yield stress; u is total displacement; τ_s is allowed straightness. Rule 1 generates a design alternative by changing the joining condition. Rule 2 changes the current joining method to another joining method. A comparison index (Appendix E.1) can be used as a criterion to select possible joining methods. In this work, the index of strength is used to select joining alternatives. Selecting a better joining method is a decision making problem and the characteristics of each joining method and their relationship with multiple disciplines should be investigated. In this work, arc welding and riveting processes are investigated as a case study. Rule 3 generates a design alternative by changing material. There are certain sets of material, which can be joined by a joining method. By using the joining and material knowledge shown in Appendices E.2 and E.3, alternative material can be selected and considered for a given joining method.

Table 5-11 Alternative generation rules

Rules	Condition	Implication		
		Principle	Welding	Riveting
Rule 1	max $\sigma < \sigma_{yield} \wedge$ max $u < \tau_s$	Change joining condition	Reduce welding temperature	Increase the number of rivets
Rule 2	max $\sigma < \sigma_{yield} \wedge$ max $u < \tau_s$	Change joining method	Use other possible joining method (with lower value in Appendix E.1)	Use other possible joining method (with lower value in Appendix E.1)
Rule 3	max $\sigma < \sigma_{yield} \wedge$ max $u < \tau_s$	Change material	Select other weldable material (Appendix E.2)	Change to a material with higher allowable stress (Appendix E.3)

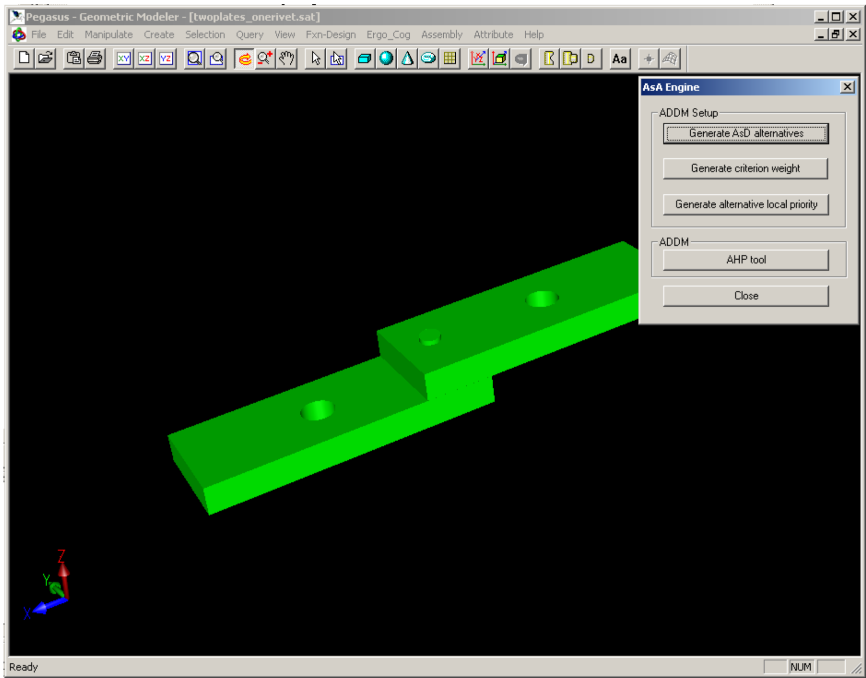


Figure 5-26 AsA engine

Figure 5-26 shows the user interface of the AsA engine. By using the AsA engine, AsD alternatives are generated and weights of criteria and local priorities of the alternatives are determined. In this implementation, three criteria, (i.e., design, cost, and quality) are used. The weights of AsD criteria can be determined from external rule base or users. As a demonstration purpose, a pairwise comparison matrix for criteria is generated as shown in Table 5-12. The cost criterion received the highest weight and the quality criterion was ranked next highest.

Table 5-12 Pairwise comparison matrix for criteria

Criterion	Design	Cost	Quality
Design	1	1/4	1/3
Cost	4	1	1
Quality	3	1	1

Table 5-13 Assembly design alternatives and evaluation values generated by the AsA engine

Design Alternatives	Material	Joining	**Design Evaluation**	**Cost Evaluation ($)**	Max Deformation (mm)	Max Stress (MPa)	**Quality Evaluation**
A	AA 6063	Two structural rivet	1.10	1.38	0.48	43.88	0.39
B	AA 6063	GMAW	1.00	3.86	0.30	17.14	0.22
C	Structural Steel rivet	One structural rivet	1.05	1.18	0.97	312.19	1.39

Table 5-13 shows three AsD alternatives and their evaluation values. The design evaluation values are determined by the design complexity model. Since the assembly components are from the same shape category (category B), alternative B received the value of “1”. This complexity model only considers the complexity of the shape of assemblies. Thus, the welded design is considered not requiring any additional components. Alternative A received the value of “1.10”, because the shape requires two rivet geometries. Similarly, alternative C received the value of “1.05”. The joining costs of each alternative are determined by the joining cost model. Table 5-14 shows cost components and values required to estimate riveting cost. To estimate the riveting cost, the cost of an aluminum rivet, R, is assumed to be 0.5 ($/each), operator factor, K, to be 80 (%), and labor rate, L, to be 15 ($/h). The cost of a steel rivet is assumed to be 0.4 ($/each). Similarly, welding cost is estimated based upon the equations and parameters presented in Tables 5-15 and 5-16.

Table 5-14 Riveting cost estimation

Cost	Notation	Unit	Equation	Value
number of rivet required	N_r			2
riveting time per unit rivet	T_{re}	h	1/K	0.01
total riveting time	T_r	h	$N_r \times T_{re}$	0.03
labor cost	C_r	$	$L \times T_r$	0.38
rivet cost	C_{cr}	$	$N_r \times R$	1.00
total riveting processing cost	C_{trp}	$	$C_r + C_{cr}$	**1.38**

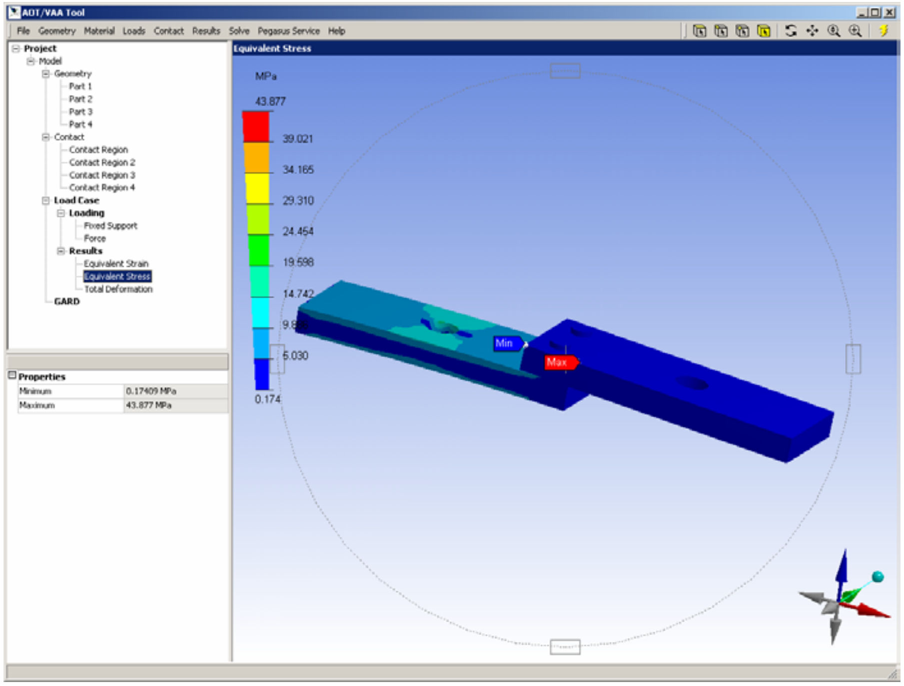
Table 5-15 Welding cost estimation

Cost	Notation	Unit	Equation	Value
gas cost per unit weight of deposited metal	C_g	$/kg	$G \times F/D$	5.00
power cost per unit weight of deposited metal	C_p	$/kg	$P \times V \times (A/1000) \times D$	0.44
cost of materials per unit weight of deposited metal	C_m	$/kg	M/E	0.03
labor rate per unit weight of deposited metal	C_l	$/kg	$L \times K/(D \times 100)$	30.00
overhead cost per unit weight of deposited metal	C_o	$/kg	$O \times K/(D \times 100)$	22.50
total cost of weld per unit weight of deposited metal	C_{wd}	$/kg	$C_g+C_p+C_m+C_l+C_o$	57.97
total cost of weld per unit length of joint	C_{wj}	$/m	$C_{wd} \times S$	0.02
total cost of weld	C_w	$	$C_{wd} \times W \times N$	**2.57**
total welding time	T_w	h	$W/(D \times K)$	923.33
total weight of weld metal	W_w	kg	$S \times N \times C$	0.886
welding time per unit length for a specific joint	T_{wj}	h	$W_w+(D \times K/100)$	0.92
electrode or wire	R_{ce}	kg	W_w+D	1.09
SAW flux	R_{cs}	kg	$1.5 \times W_w/E$	0.01
gas	R_{cg}	m^3	$(F \times T_w)/E$	0.005
required electrode or wire cost	C_{ce}	$	$EC \times R_{ce}$	1.19
required SAW flux cost	C_{cs}	$	$SC \times R_{cs}$	0.015
required gas cost	C_{cg}	$	$GC \times R_{cg}$	0.084
total consumables cost	C_c	$	$C_{ce}+C_{cs}+C_{cg}$	1.29
Total welding processing cost	C_{twp}	$	C_w+C_c	**3.86**

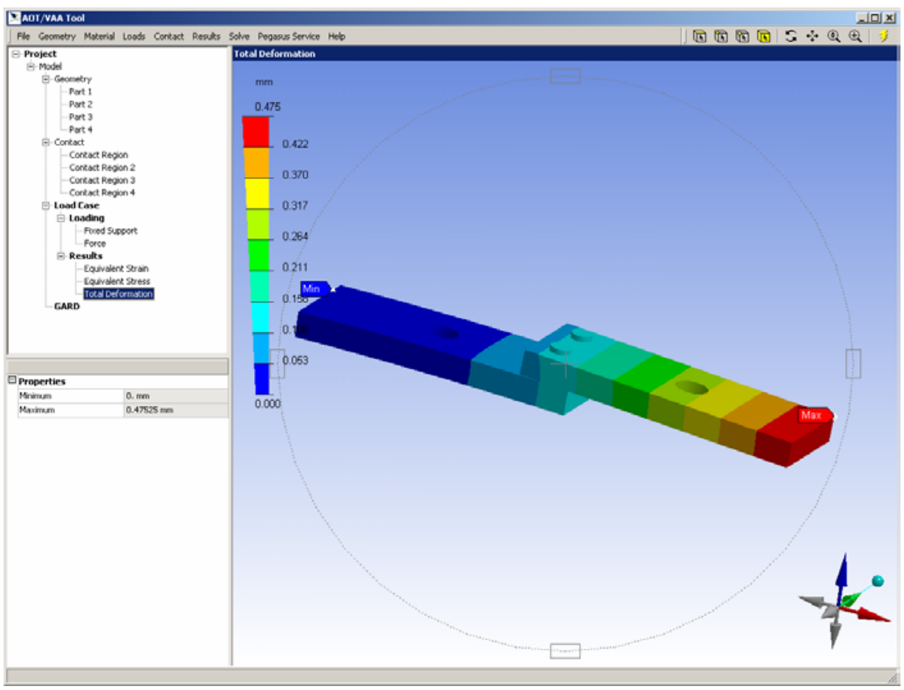
Table 5-16 Parameters used to estimate welding

Parameters	Notation	Unit	Value
amperes	A	A	5
volt	V	V	220
specific gravity of metal	C	kg/m^3	55360
deposition rate	D	kg/h	0.2
flow rate	F	m^3/h	0.5
unit cost of gas or flux by volume	G	$\$/m^3$	2
deposition efficiency	E	%	95
operator factor	K	%	15
labor rate	L	$/h	40
cost of materials	M	$/kg	2.65
length of specified weld	N	m	0.04
overhead rate	O	$/h	30
power cost	P	$/kWh	2
total weight of weld metal	W	kg/m^3	2770
cross-sectional area of weld joint	S	m^2	0.0004
electrode or wire cost	EC	$/kg	1.1
SAW flux cost	SC	$/kg	1.1
gas cost	GC	$\$/m^3$	18

To evaluate quality, the max deformation and max stress are used as the evaluation model. Those analysis values are obtained by VAA. Figures 5-27 to 5-29 illustrate VAA results (total deformation and equivalent stress) of the three AsD alternatives. In this case study, only effects of the joining processes are simulated. This work concentrates on studying the physical effects of different joining processes. Although the max total deformation of the alternatives A and B satisfies the material criterion, it is still difficult to make a decision. For example, alternatives A and B have good quality-evaluation results and alternative C is best in the cost aspect. In this situation, determining assembly design decision requires an analytical process to justify the decision.

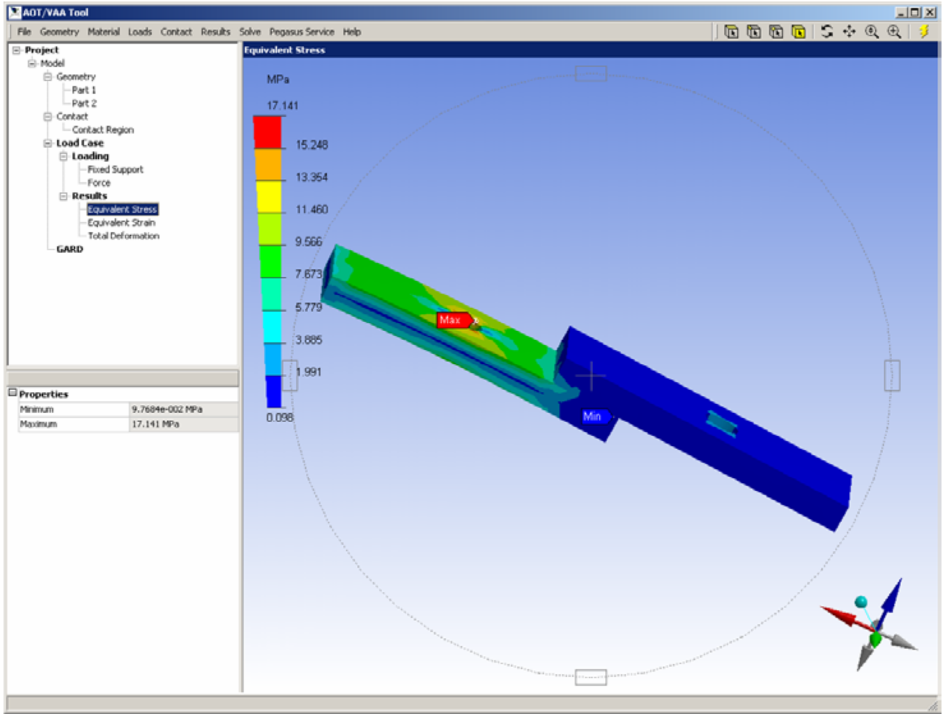


a) Equivalent stress

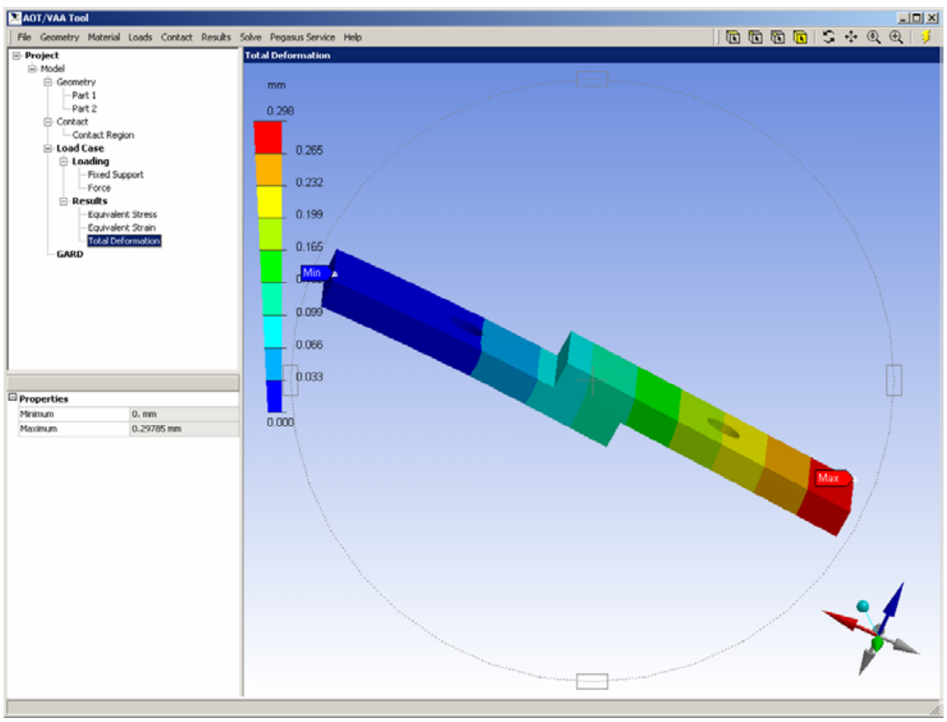


b) Total deformation

Figure 5-27 VAA result - Alternative A

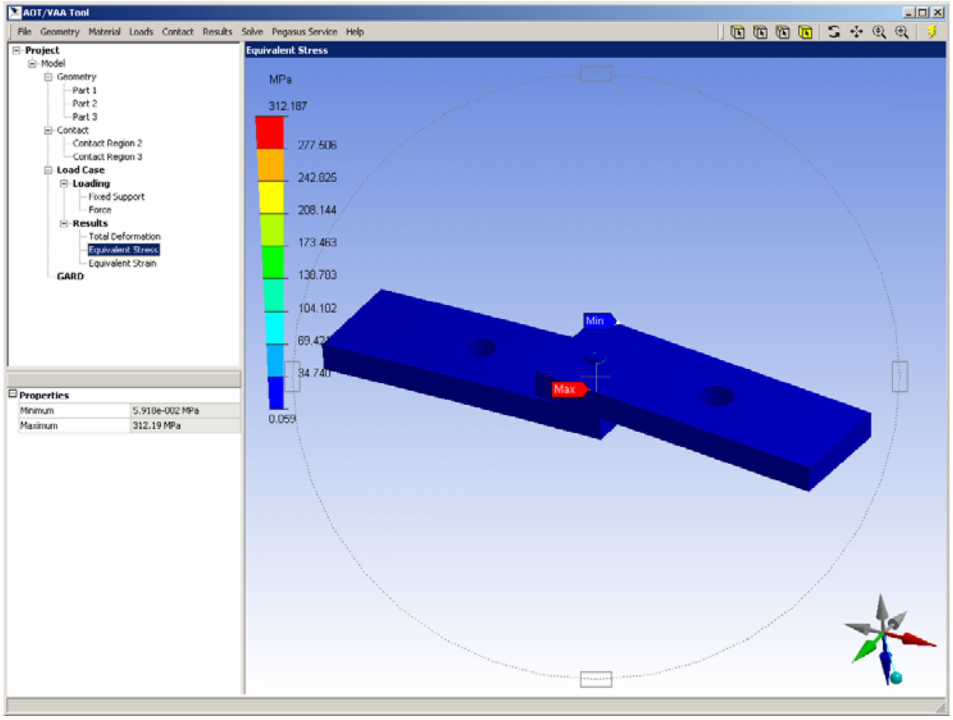


a) Equivalent stress

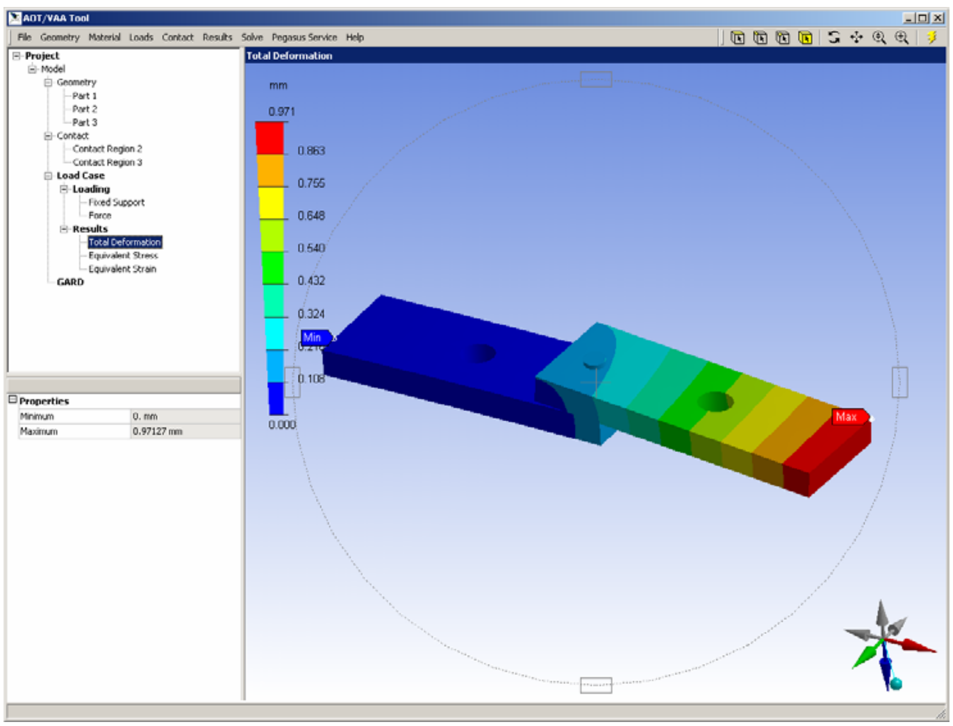


b) Total deformation

Figure 5-28 VAA result - Alternative B



a) Equivalent stress



b) Total deformation

Figure 5-29 VAA result - Alternative C

Table 5-17 Pairwise comparison matrixes for alternatives generated from the evaluation

Criterion	Pairwise comparison matrix			
Design criterion	Alternatives	A	B	C
	A	1.00	0.11	0.20
	B	9.00	1.00	5.00
	C	5.00	0.20	1.00
Cost criterion	Alternatives	A	B	C
	A	1.00	9.00	1.00
	B	0.11	1.00	0.11
	C	1.00	9.00	1.00
Quality criterion	Alternatives	A	B	C
	A	1.00	0.50	9.00
	B	2.00	1.00	9.00
	C	0.11	0.11	1.00

Figure 5-30 AHP setup file – XML data

Table 5-17 shows pairwise comparison matrixes automatically generated from the evaluation values. After the alternatives, weights, and local priorities are obtained, an AHP setup file is generated in a XML format (Figure 5-30). Appendix B.2 shows details of an example of the AHP setup file. The XML file includes joining information of three AsD alternatives and the AHP setup. The AHP setup includes weights of AsD criteria and local proprieties of AsD alternatives. Once the AHP setup is done, the AHP tool is triggered. The obtained alternatives, weights, and local priorities are input of the AHP tool. In this work, the Expert Choice software of Expert Choice, Inc. is employed as the AHP tool. In this implementation, the AsA engine and the AHP tool are integrated locally. With benefits of XML, those tools can be integrated remotely while keeping "plug and play" modularity.

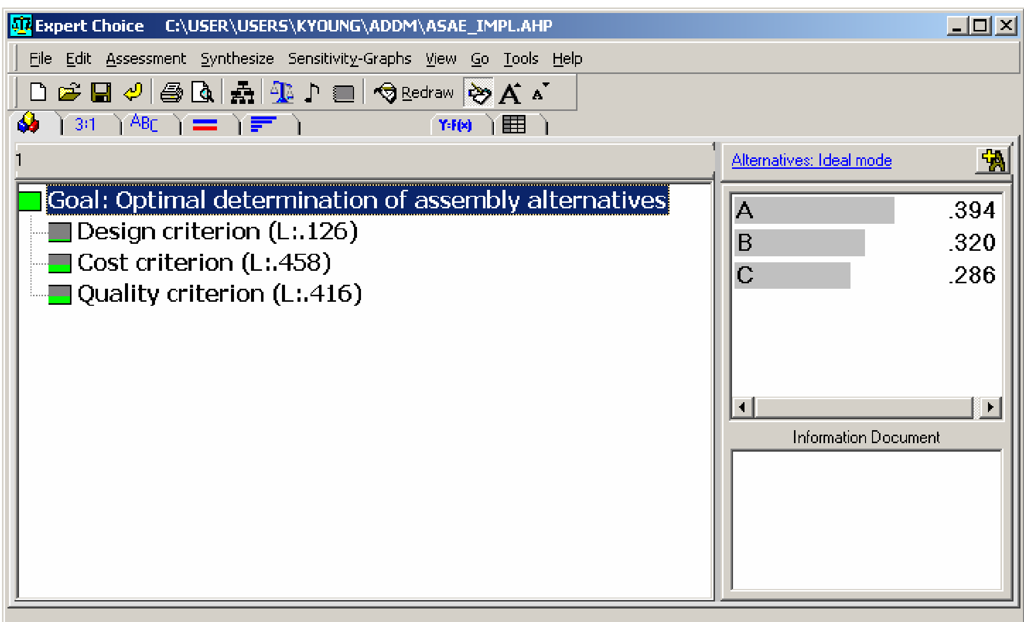


Figure 5-31 AHP result

Figure 5-31 shows the AHP tool and the result obtained. By using this tool, the weights and local priorities are confirmed by the decision maker (designer). From this AHP analysis, AsD alternative A using two structural rivets obtains the highest evaluation value and alternative B, in which welding is used, has the second highest evaluation value. The more detail functions of this AHP tool will be explained in the next chapter to validate the developed concepts and tools.

5.9 Summary

Appropriate assembly design should be determined by considering mechanical and mathematical implication information and assembly/joining knowledge. Finding an optimal assembly design from a very large design alternative set, a proper design of experiments methodology should be developed. In this work, a new framework of an assembly design decision-making procedure is developed. By following the developed procedure, dominant design alternatives are investigated and their potential problems are predicted by analytical experiments. Eventually a designer is guided to a right path to experiment for making assembly design decision.

The aim of this chapter is to introduce a new method to resolve the assembly design decision (ADD) problem, called ADDM. Assembly plays a very important role in manufacturing industries. Appropriate joints, which are inevitable to assembly structure, should be determined by considering mechanical and mathematical implication information and assembly/joining knowledge. In this work, the hierarchical semantic net (HSN) model is introduced as a core model to represent evaluation knowledge and assembly design knowledge, which is unavoidable in the multicriteria and knowledge-based ADD problem. In the HSN model, the semantic net capturing assembly relations is embedded in the assembly design alternative, which is a component of the AHP model. The ADDM framework developed in this work is implemented in an assembly advisory engine. The contributions of this work are summarized as follows:

Contributions

1. The semantic net-based assembly relation model (ARM) can efficiently capture assembly design knowledge and convey the knowledge to downstream activities, such as ADDM.
2. The captured assembly design knowledge can be seamlessly transformed into an AHP-hierarchy (HSN) for ADDM.
3. Designers can impose preferences on the ADDM using AHP.
4. The ADDM can manage interactions between alternatives and also between criteria.
5. Evaluation values of each design alternative are determined from inner knowledge (factors) of each criterion. The evaluation values in return are dynamically added to the semantic net of alternatives. By using this method, independence between evaluation knowledge and design knowledge is maintained. It enables the ADDM system to be scalable and extendable.

6.0 VALIDATION

The principles developed in this work are tested and validated using a case study: the design of a sub-assembly of an automotive space-frame. The primary activity involved in the validation of the work is answering the question: "Do the developed concept and methods accomplish the research objectives defined at the beginning of this work?" To answer this question, it is important to re-emphasize the objective of an assembly design in general and the objective of the assembly operations tools developed in this work. How is an assembly design judged good or bad? In this work, a design is considered good if it successfully satisfies the design specifications and/or the designers (decision makers). The assembly operation tools developed in this work are used to generate an assembly design, evaluate the generated design, and support an assembly design decision. Hence, the developed assembly design framework can be validated by evaluating how efficiently an assembly design can be generated and whether the assembly operation tools can guide a designer into the right direction and eventually improve the current assembly design.

To evaluate the assembly design framework, the following procedure is used.

- Select a case study comprising an assembly design that requires the use of joining.
- Generate an assembly design with aid of the AsD engine.
- Evaluate the assembly design using the AsI engine.
- Generate assembly alternatives with aid of the AsA engine, after potential problems of the current assembly design are indicated.

- Compare the design alternative and the current assembly design, highlighting the benefits and disadvantages of the new design.

6.1 The Architecture of The Assembly Operation Tools

The developed AsD formalism and three core engines (AsD engine, AsI engine, and AsA engine) are integrated in an architecture of assembly operation tools, as illustrated in Figure 6-1. The AsD model is generated from the AsD engine capturing prescribed joining operations. The generated AsD model is sent to the AsI engine and assembly implications are extracted from the AsD model. The assembly implication is displayed through an assembly graphic engine to help a designer make better decisions. At the same time, the implication is equally sent to the AsA engine, which considers various constraints and built-in knowledge and suggests design alternatives if required.

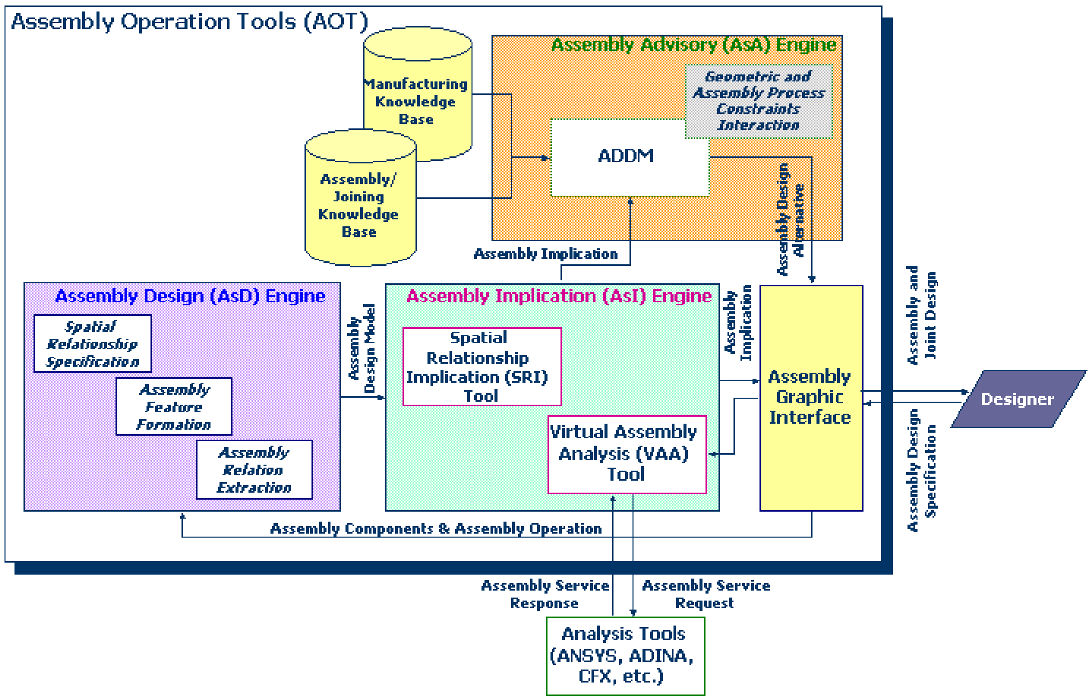


Figure 6-1 Architecture of assembly operation tools

6.2 The Assembly Design Procedure

Before applying the new assembly design concepts, the general assembly design steps identified above are expanded to illustrate the specific assembly design actions that are involved when using the methods and computer tools developed in this research. The process flow associated with these steps is illustrated by the flow chart of Figure 6-2. This assembly design procedure is described below:

1. Share assembly component design among design participants and identify the needs of assembly.
2. Load the selected assembly components in the AsD engine.
3. Specify spatial relationships to generate the assembly.

4. Specify joining method and conditions.
5. Validate spatial relationship implication with the aid of the SRI tool.
   - If it violates the designer's intents, go to step 3. If not, continue.
6. Generate a XML formatted AsD model and GARD to share the assembly design with other design participants.
7. Generate an AsAM with aid of the VAA tool and trigger the VAA.
8. Check whether the current AsD violates any assembly specification with aid of the AsA engine.
9. If any specification is not violated, keep the current AsD. If the any specification is violated, the AsA engine generates design alternatives and sets up the ADDM.
   - To predict physical effects of alternative joining, repeat step 7.
10. Trigger the AHP tool and evaluate the design alternatives.
11. If the alternative AsD satisfies design specification and/or the designer, then keep the alternative AsD. If the alternative AsD doesn't satisfy design specification, repeat steps 9 and 10 with the best alternative AsD.

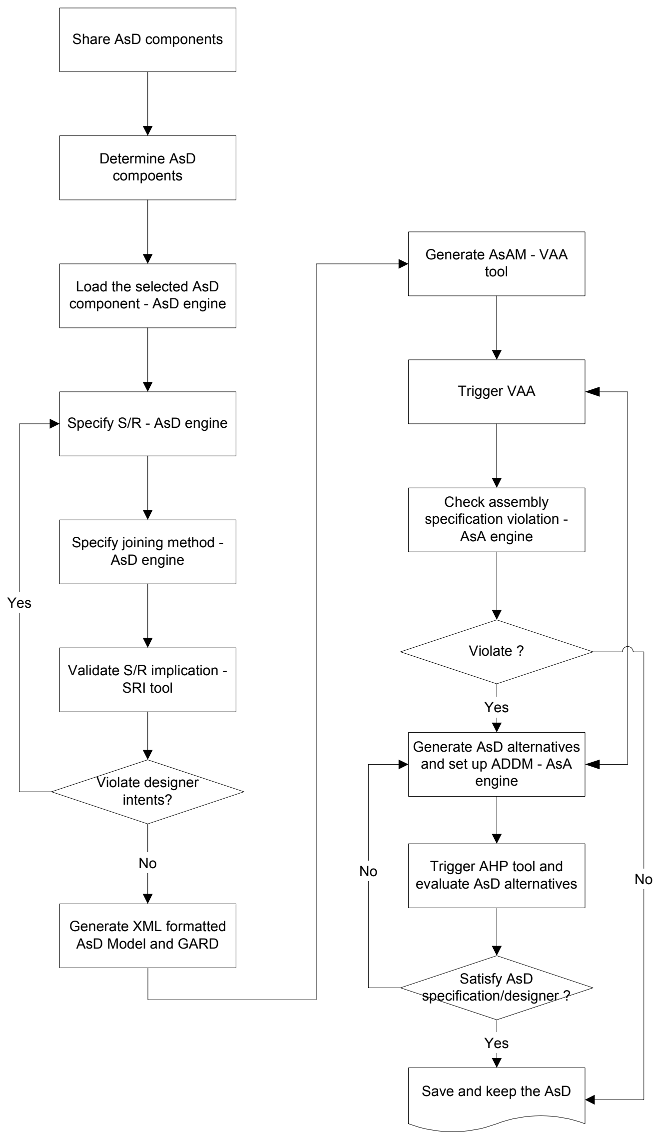


Figure 6-2 Assembly design flow diagram

6.3 Assembly Design of an Automotive Space-Frame Sub-Assembly

In this work, a realistic example is used as a case study, which is an aluminum space frame assembly for an automobile (Figure 6-3). The welded frame (Buchholz 1999) is made up of thin walled aluminum beams with rectangular sections and flat planer sections. Aluminum alloy (such as 6061 or 6063) extrusions have been considered as materials. Moreover, recent emphasis on lightweight environmentally sound car design has opened up the possibility of substituting lower-density corrosion-resistant recyclable aluminum for steel in car bodies (Ashley 1994). However, the high distortion of aluminum alloy is a difficult problem to overcome to achieve precision manufacturing. For example, aluminum alloys 6061-T6 and 6063-T6 have a deformation index of 0.01 (worse) against an index of 1.0 for mild steel (Radaj 1992).

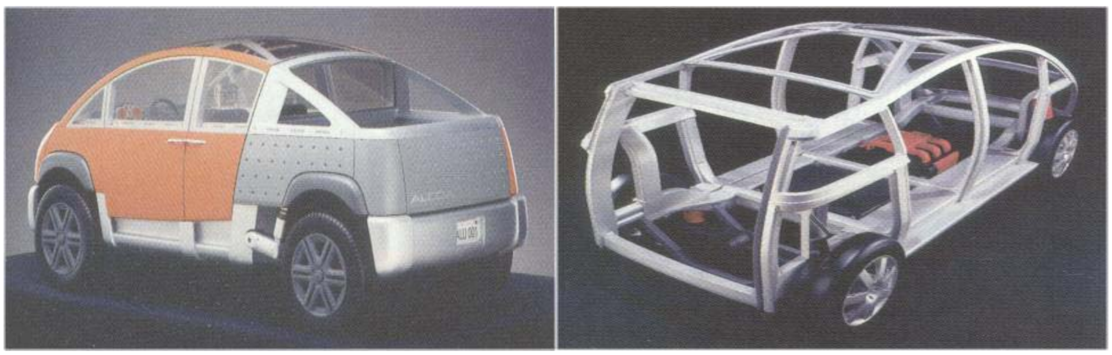


Figure 6-3 Aluminum concept car and body frames (Buchholz 1999)

For validation purposes, a base-frame sub-assembly (Figure 6-4) of the body frames is used. The sub-assembly can be found on the front, bottom frame of the body. The assembly scenario is that a crossing extrusion is attached to a side extruded beam sub-assembly by welding. The two extrusions of the extruded beam sub-assembly were joined by welding. The welding on this kind of structure typically generates distortion and residual stress, which can cause a fitting distortion problem. Examples of a fitting distortion problem can be found in the door, bumper, and window assemblies. The distortion on the front bumper area can seriously weaken structural impact performance. The kind of problem can be predicted transparently by using the developed AOT. Figure 6-5 illustrates the dimensions of the joint. Note that each dimension is reduced to keep a confidentiality agreement, while maintaining aspect ratios.

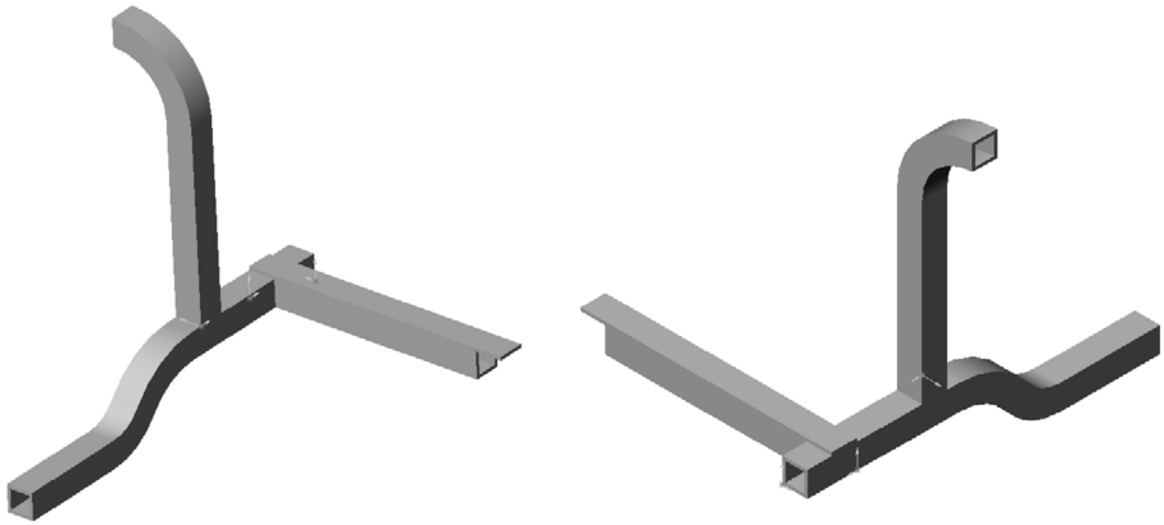


Figure 6-4 Base-frame assembly and joint of extruded beams: 3D solid view

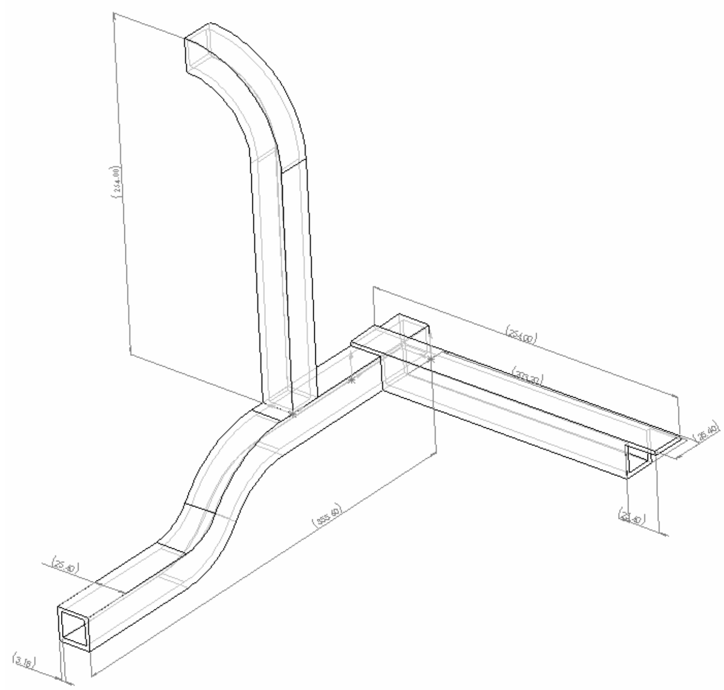
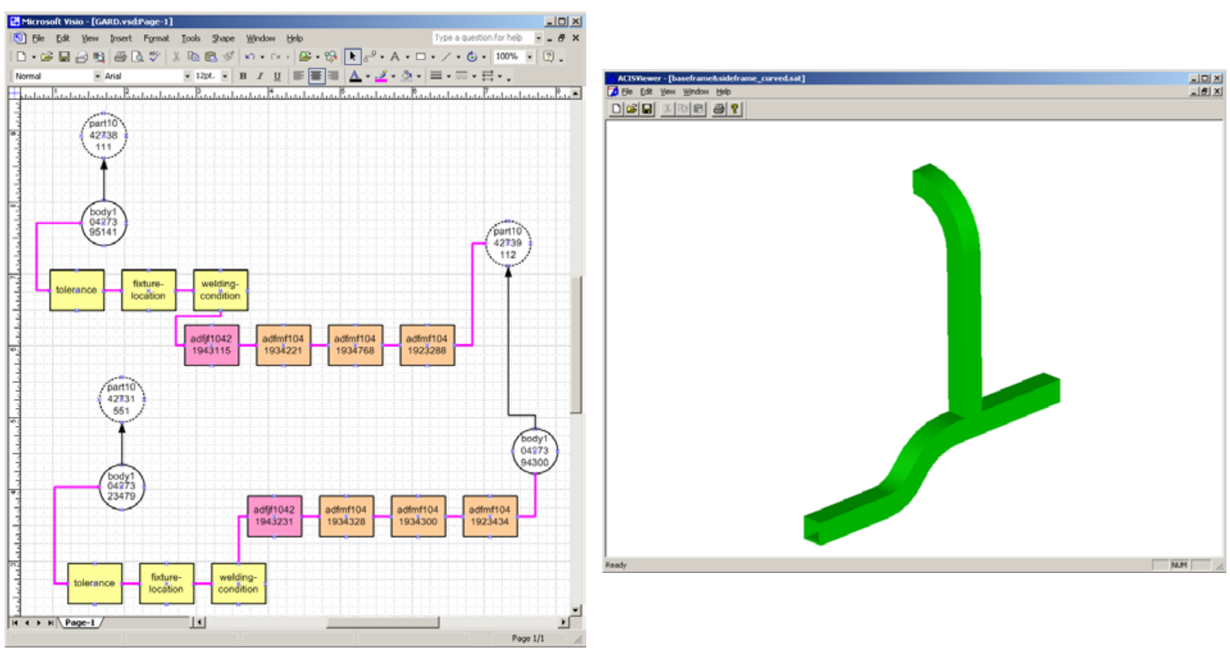


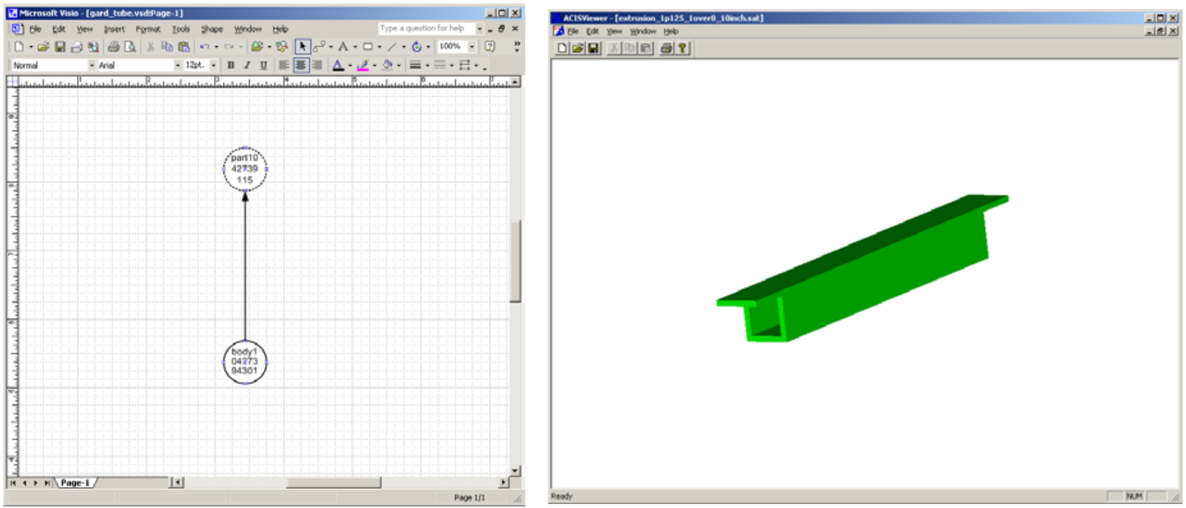
Figure 6-5 Base-frame assembly and joint of extruded beams: 3D wire-frame view (mm)

6.4 Product Data Sharing

Assembly design collaborators (*e*-designers) can share an AsD model interacting with different CAD systems in the service-oriented architecture. Figure 6-6 shows GARDs illustrating assembly relations and relevant ACIS models of each assembly component. In this particular GARD, the bottom beam (part1042739112) of the side extruded beam assembly is already joined to an extruded beam (part1042738111) by welding (see Figure 6-6-a). Through the service-oriented architecture, AsD models of assembly components can be provided remotely to the system integrator and the system integrator can generate an assembly. In this case, the two components (i.e., the side extruded beam assembly and the bottom extruded beam) are selected for joining.



a) Side extruded beam assembly



b) Bottom extruded beam

Figure 6-6 ACIS models and GARDs used for PDS

6.5 Spatial Relationship Specification

Once AsD models of each assembly component are acquired, they are loaded into the AsD engine and the designer (system integrator) can specify spatial relationships between the components. By interactively assigning spatial relationships, the designer can assemble components together to make final products and infer the d.o.f. remaining on each of the components. In this case, one against relationship between faces and two aligned relationships between edges are assigned. Figure 6-7 shows the last specification of the aligned relationship. Note that the inferred (designed) d.o.f. are fixed.

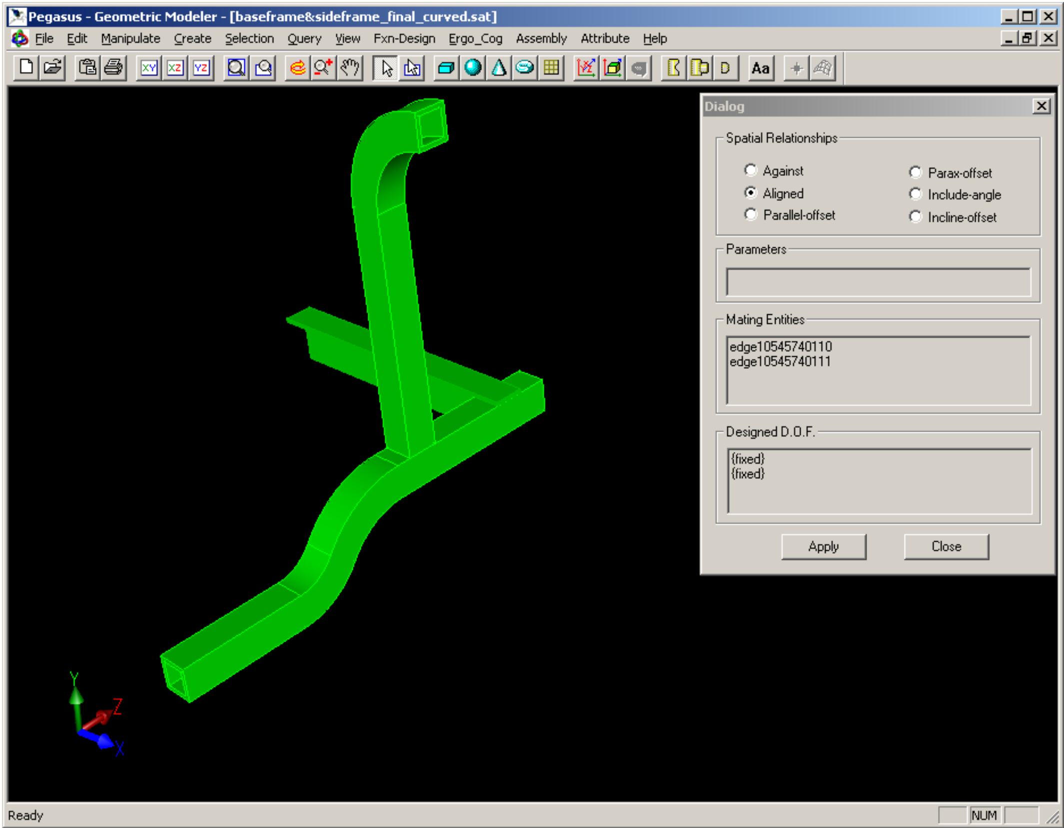


Figure 6-7 Spatial relationship specification on the base-frame joint

6.6 Joining Method Specification

Actual assembly is realized by joining, whereas joining method specification is a core process to finalize an assembly design. The developed AsD engine currently provides the capability to specify two categories of joining methods (i.e., welding and riveting). Joining categories can be extended. Generally, joining processes happen on the mating entities, such as weld seams. If a designer specifies a geometric entity, which does not belong to the mating features as a joining entity, then it violates the validity of joining. As shown in Figure 6-8, a gas metal arc welding process is selected as a joining method and a weld seam is specified on a selected edge. Using the AsD engine, joining conditions can be specified. Alternatively, detailed joining conditions can be specified with the VAA tool. Once finishing this process, a joint feature is internally generated capturing the specified information.

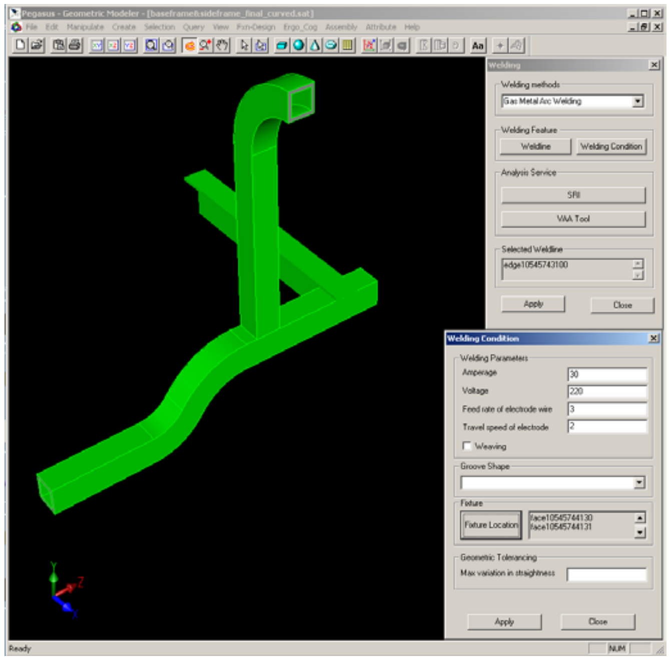


Figure 6-8 Joining method specification on the base-frame joint

6.7 Spatial Relationship Implication Validation

In assembly design, the desired spatial relationships, which are inferring the designed d.o.f., are realized and maintained (or enforced) in the physical assembly by joining. Once the designer specifies a joining method, the SRI tool validates whether the designed d.o.f. are satisfied by the specified joining method. Since welding processes restrict all d.o.f. (fixed), the designed d.o.f. are fully satisfied by welding (see Figure 6-9).

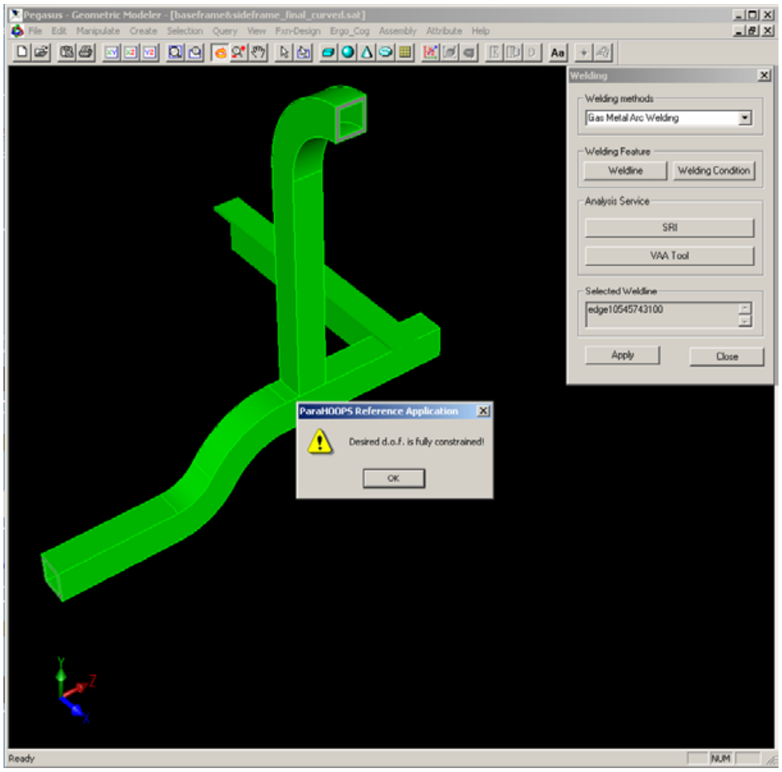


Figure 6-9 SRI tool indicating that welding satisfies the designed d.o.f.

6.8 AsD Model Generation

The internally generated AsD model can be exported in XML format with aid of the AsD engine. The sole purpose of XML is to carry data. The AsD model is translated to the defined XML-AsD format to store and exchange AsD information. XML is a cross-platform, software and hardware independent tool for transmitting information. This makes it particularly applicable to represent AsD data that may be exchanged between different CAD platforms and systems. The XML formatted AsD model generated from the above procedures can be found in Appendix B.1. Figures 6-10 and 6-11 illustrate detailed ACIS entity ID's of the geometric model. The

XML-AsD model can be displayed in a pictorial format (GARD) with the aid of the GARD tool. Figure 6-12 illustrates the GARD of the AsD model of the base-frame joint. Note that the detailed information is linked to each element of the GARD.

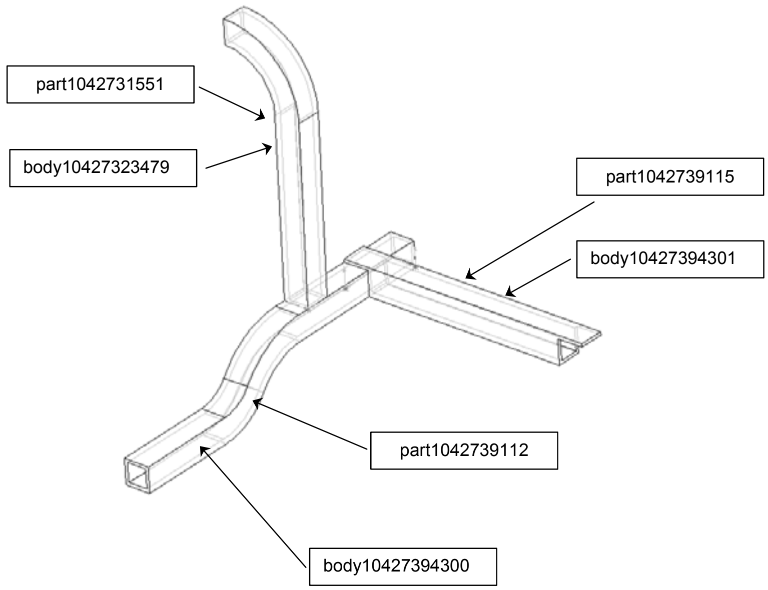


Figure 6-10 Base frame joint with ACIS entity IDs

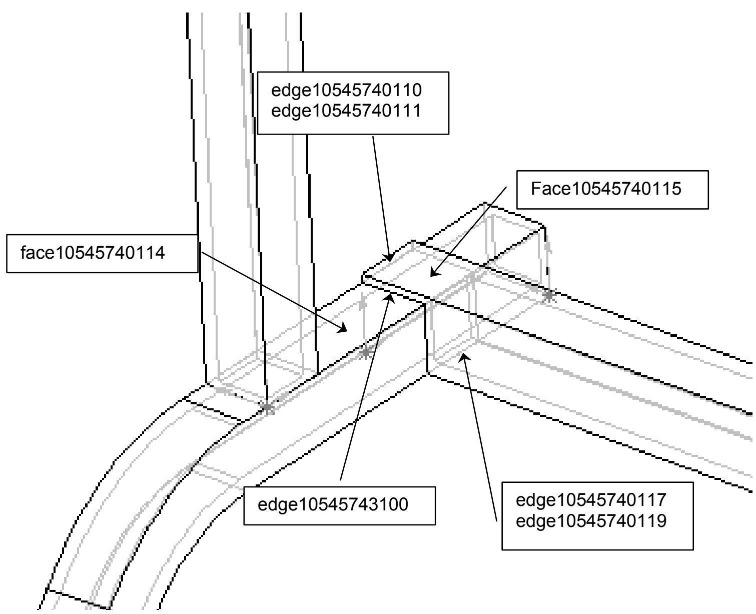


Figure 6-11 Base-frame joint with ACIS entity IDs: zoomed view

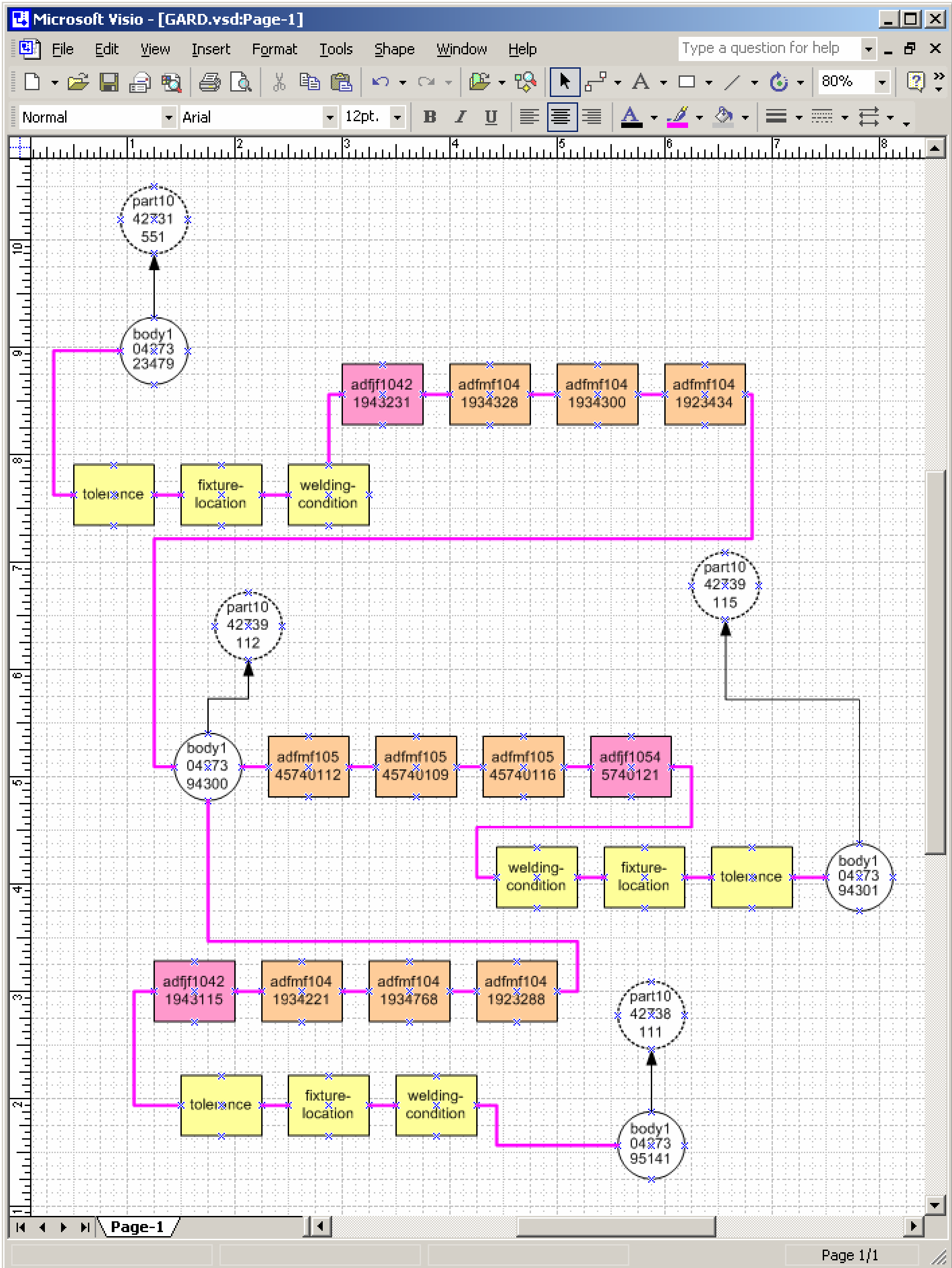


Figure 6-12 GARD of the base-frame joint

6.9 VAA Setup and Process

To integrate assembly design and assembly operation analysis, the AsD models can be translated to an assembly analysis model (AsAM). To perform VAA, the assembly/joining information necessary to assembly operation analysis can be extracted from the given AsD information. Figure 6-13 shows an AsAM of the base frame joint in the VAA tool. In addition to the given conditions, the designer can impose more conditions using the VAA tool. Once an AsAM is generated, the VAA process is ready to go. The VAA service is requested of the Pegasus service manager. The service manager dynamically determines an available VAA service provider and relevant analysis input is generated. In this case study, the ANSYS analysis solver is used as a VAA service provider. Appendix F.1 describes detailed analysis scenarios and results.

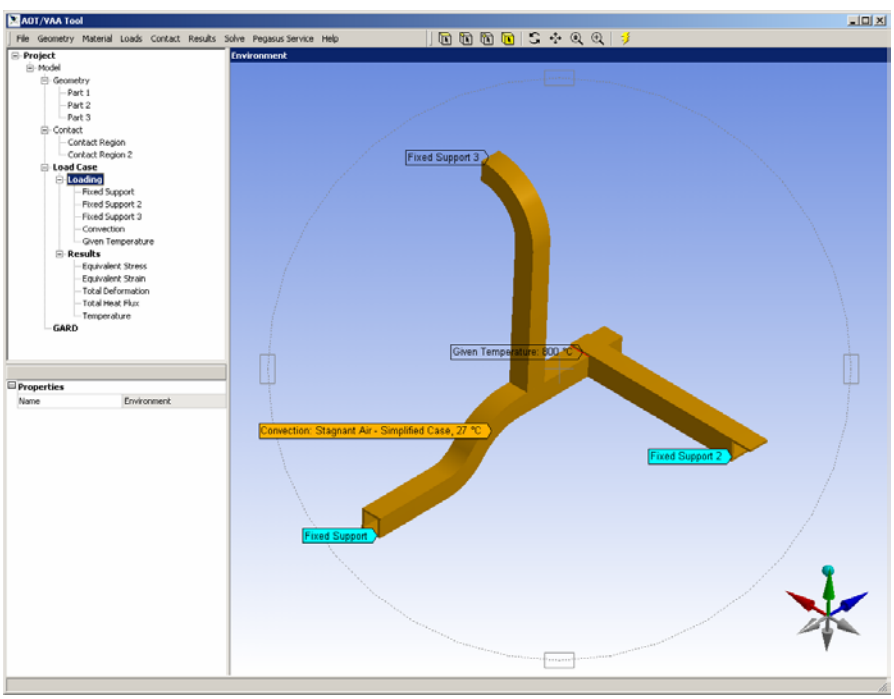


Figure 6-13 AsAM of the base frame joint

Figures 6-14 and 6-15 illustrate the VAA results for the base frame joint. The result clearly shows deformation of this structure concentrated at the welded joint. Based on this result, any AsD problem can be indicated and the AsA engine guides the designer to make a proper AsD decision within this nominal geometry.

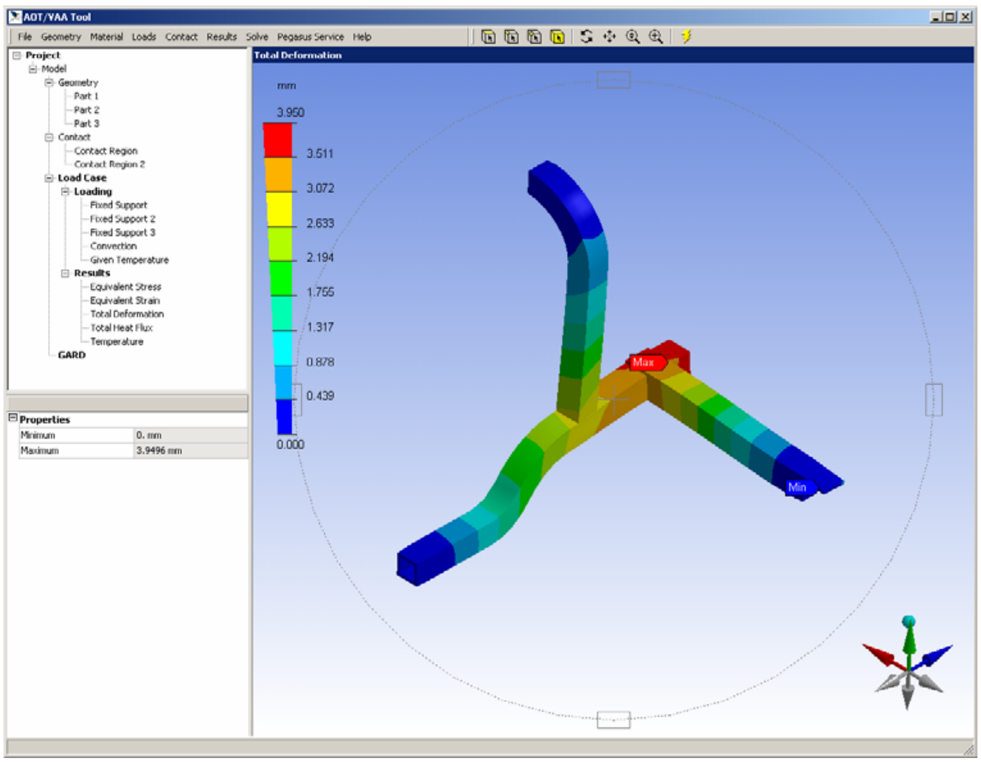


Figure 6-14 VAA result: total deformation

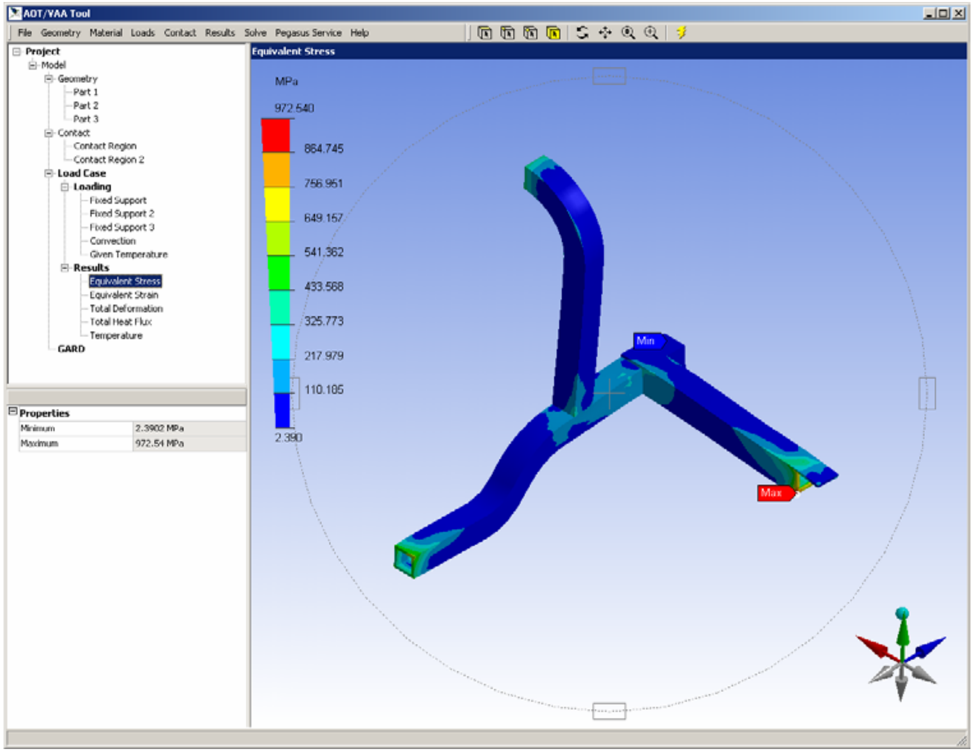


Figure 6-15 VAA result: Equivalent Stress

6.10 AsA Engine and ADDM

The AsA engine checks whether the specified joining method is feasible within nominal geometry. Figure 6-16 illustrates a case of the tolerance criterion (i.e., total deformation beyond allowed tolerance (2 mm)) being violated; the maximum total displacement is 3.950 mm. Like this case, once any AsD problem is found, the AsA engine generates relevant AsD alternatives and guides the designer to make a proper AsD decision.

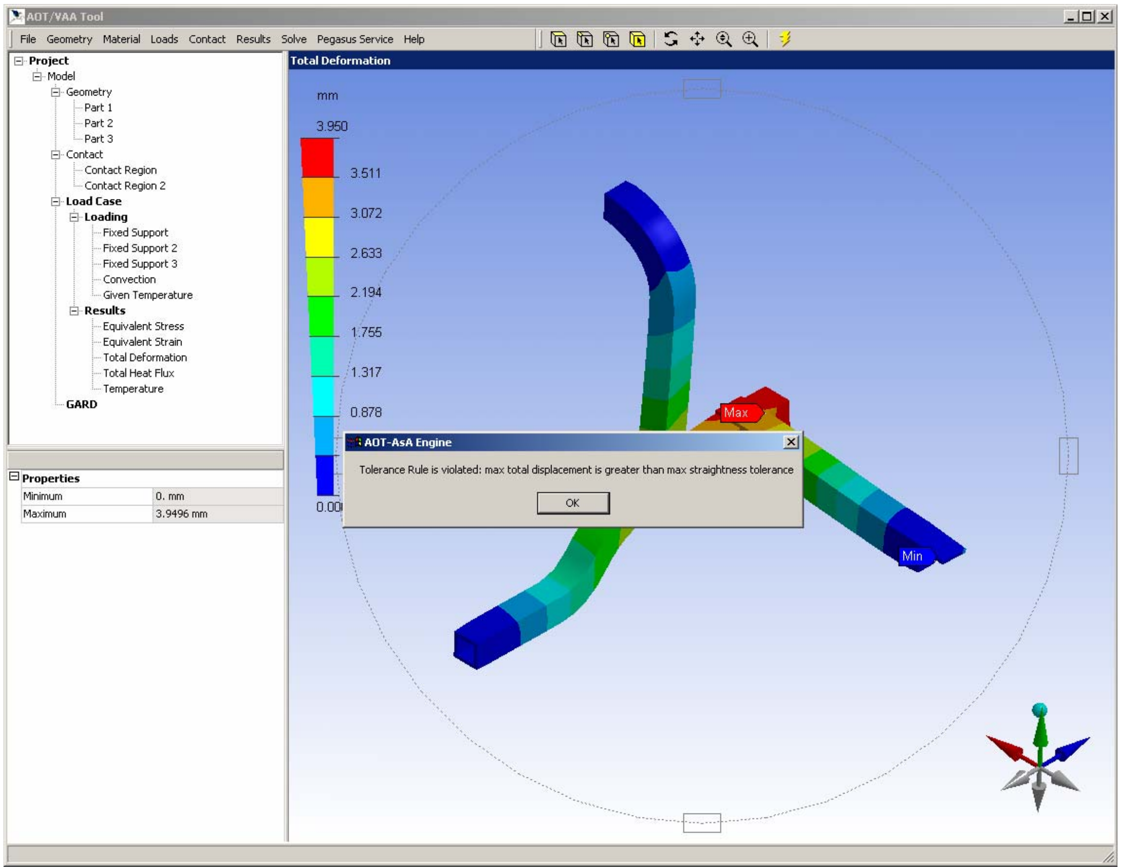


Figure 6-16 AsA engine indicating tolerance criterion violation

Figure 6-17 shows the user interface of the AsA engine and a XML file including information for three AsD alternatives and the AHP setup. The AHP setup includes weights of AsD criteria and local proprieties of AsD alternatives. The weights of AsD criteria can be determined from external rule bases or users. In this case study, the design and quality criteria are assumed to be more important than the cost criterion. Table 6-1 shows three AsD alternatives and their evaluation values. Two design evaluation criteria (i.e., design evaluation model I and design evaluation model II) are employed. Design evaluation model I represents design complexity and design evaluation model II shows overall weight of the assembly. Design evaluation model II is included to capture the emphasis on lightweight sound car design. Also, the quality criterion includes two evaluation models (i.e., quality evaluation model I and quality evaluation model II) to predict effects of joining and structural performance. Quality evaluation model I measures physical effects from joining and quality evaluation model II measures impact on the front bumper area. The max deformation and max stress are used as an evaluation model for the quality criteria. Those values are obtained by the VAA. This work concentrates on studying the physical effects of different joining processes. With the advantage of the dynamic HSN, additional evaluation models can be included for various considerations without affecting the whole system. Figures 6-18 through 6-23 illustrate the VAA results (total deformation and equivalent stress) of the three AsD alternatives. Although the max total deformations of all design alternatives satisfy the tolerance criterion, it is still difficult to make a decision from the AsD alternatives. While alternative B has good cost-evaluation and quality-evaluation I results, other alternatives (A and C) received good score in the design criteria. Figures 6-24 through 6-29 illustrate impact test results. As shown in Table 6-1, the alternative C has the best quality-evaluation result from the impact test; but it receives the worst values for design evaluation II. In this situation, which alternative should be selected? The following ADDM process will resolve these conflicts. Appendices F.2 to F.5 describe detailed analysis scenarios and results for each alternative. Once the AHP setup is done, the AHP tool is triggered. The obtained alternatives, weights, and local priorities are inputs of the AHP tool.

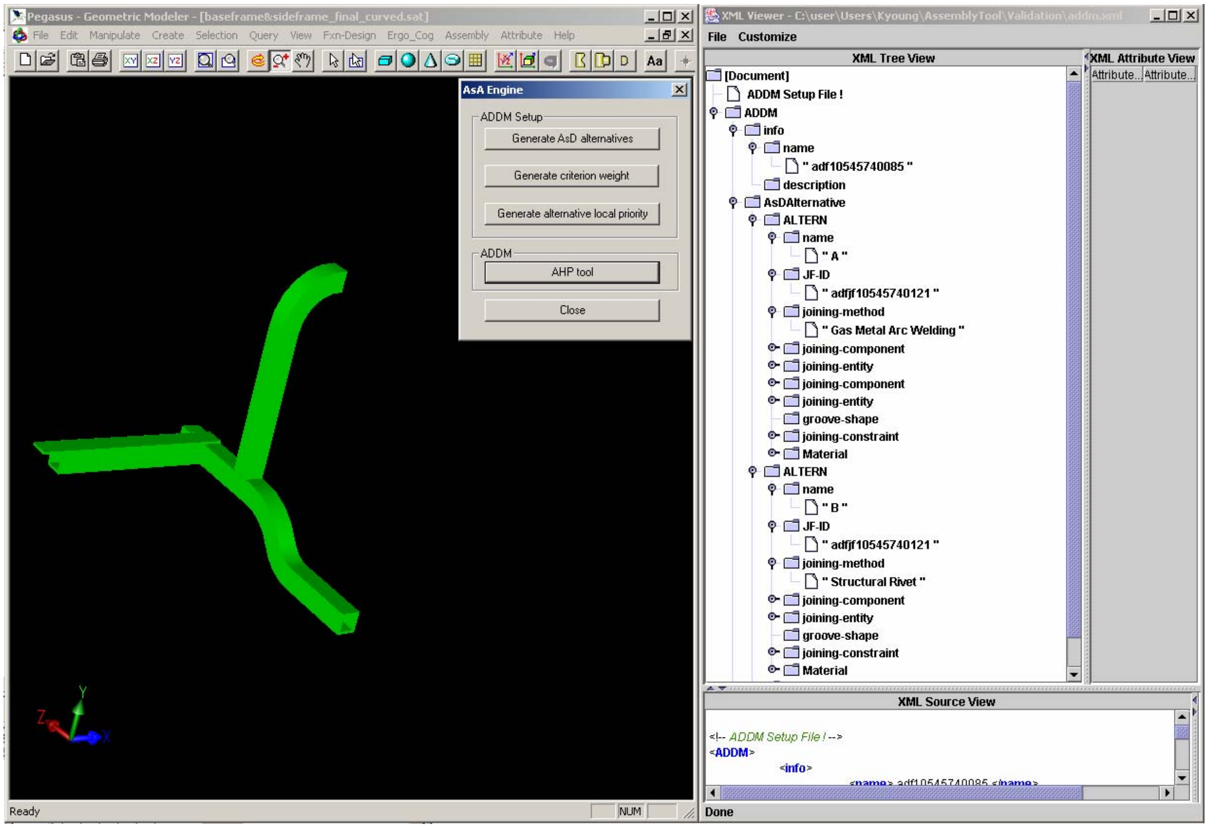


Figure 6-17 AsA engine generating assembly design alternatives

Table 6-1 Assembly design alternatives and evaluation values generated with aid of the AsA engine

Design Alternatives	Material	Joining	Design Evaluation I	Design Evaluation II	Cost Evaluation ($)	Max Deformation (mm)	Max Stress (MPa)	Quality Evaluation I	Max Deformation (mm)	Max Stress (MPa)	Quality Evaluation II
A	AA 6063	GMAW	1.5	0.69	2.52	1.93	474.36	0.78	11.89	4041.00	0.76
B	AA 6063	Structural Rivet	1.6	0.69	1.38	0.03	203.02	0.12	11.49	4091.00	0.75
C	Structural Steel	GMAW	1.5	1.95	2.33	1.84	1064.70	1.10	4.23	4080.00	0.49

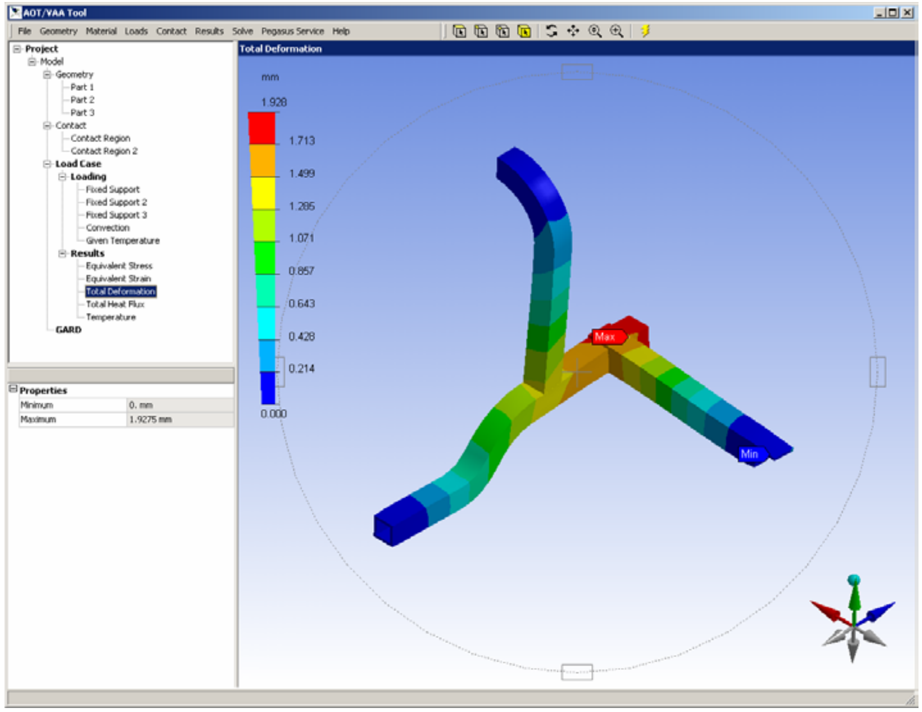


Figure 6-18 VAA result for alternative A (welding) – total deformation

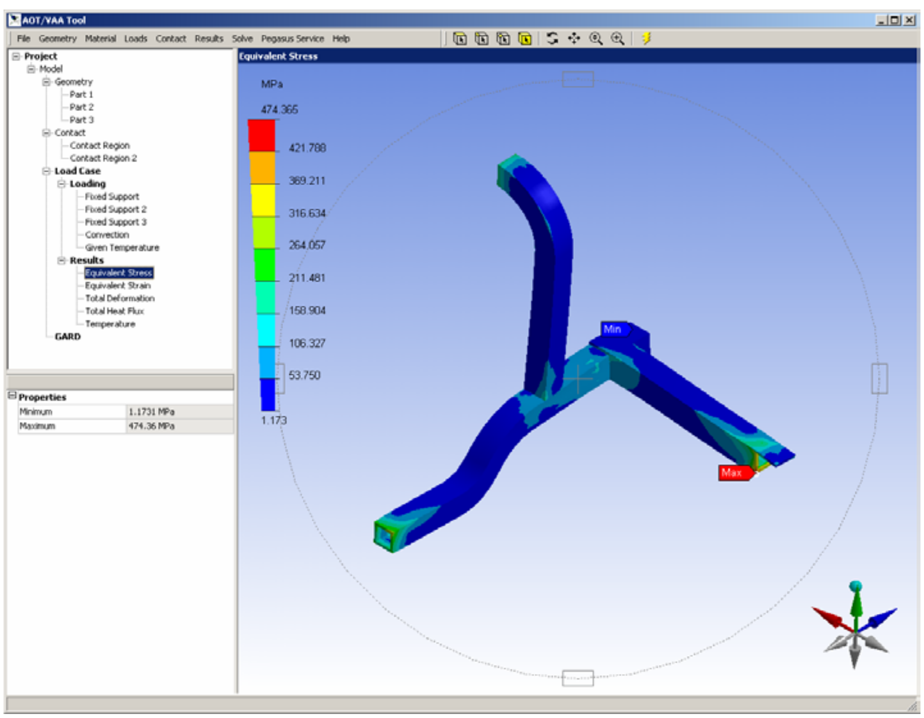


Figure 6-19 VAA result for alternative A (welding) – equivalent stress

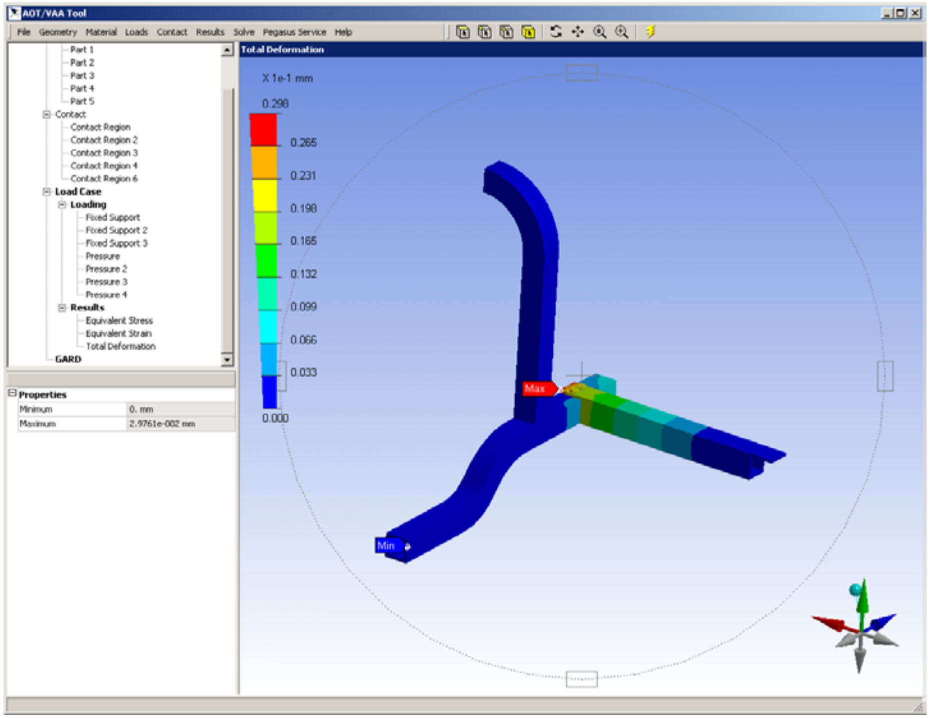


Figure 6-20 VAA result for alternative B (riveting) – total deformation

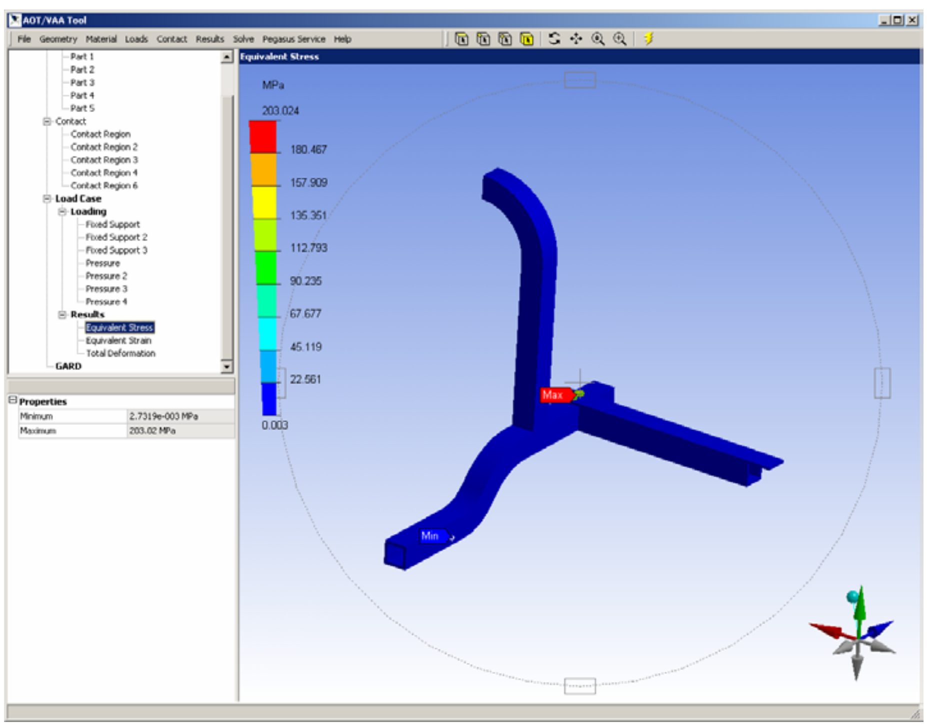


Figure 6-21 VAA result for alternative B (riveting) – equivalent stress

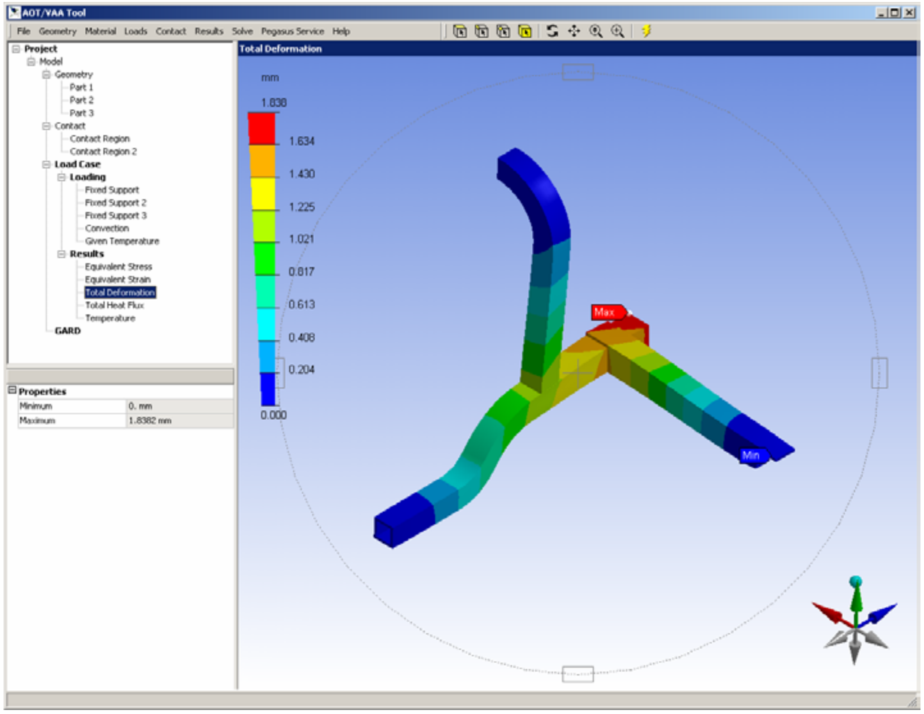


Figure 6-22 VAA result for alternative C (welding) – total deformation

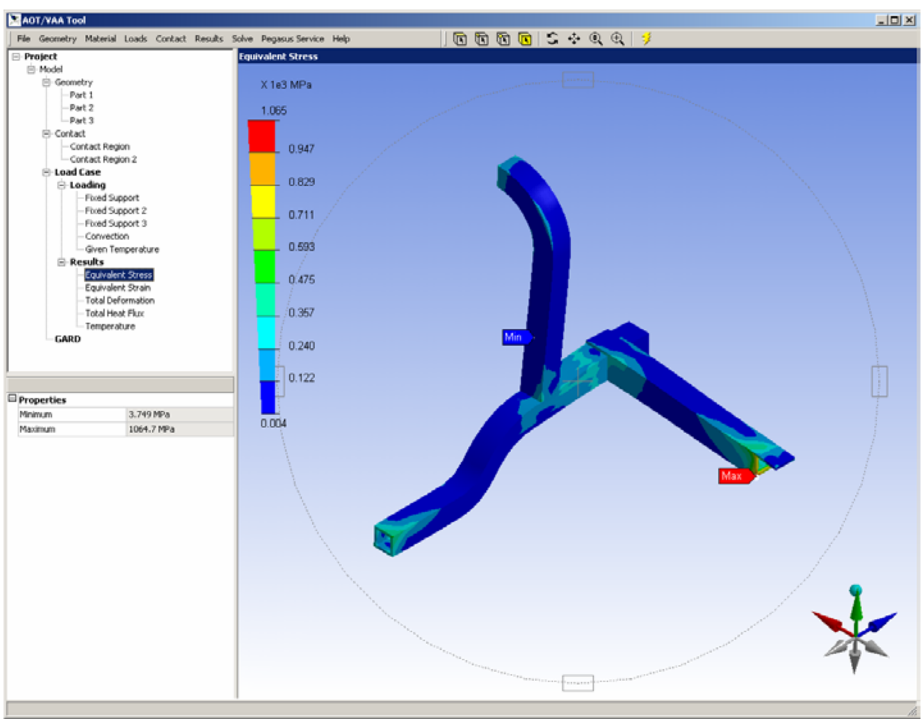


Figure 6-23 VAA result for alternative C (welding) – equivalent stress

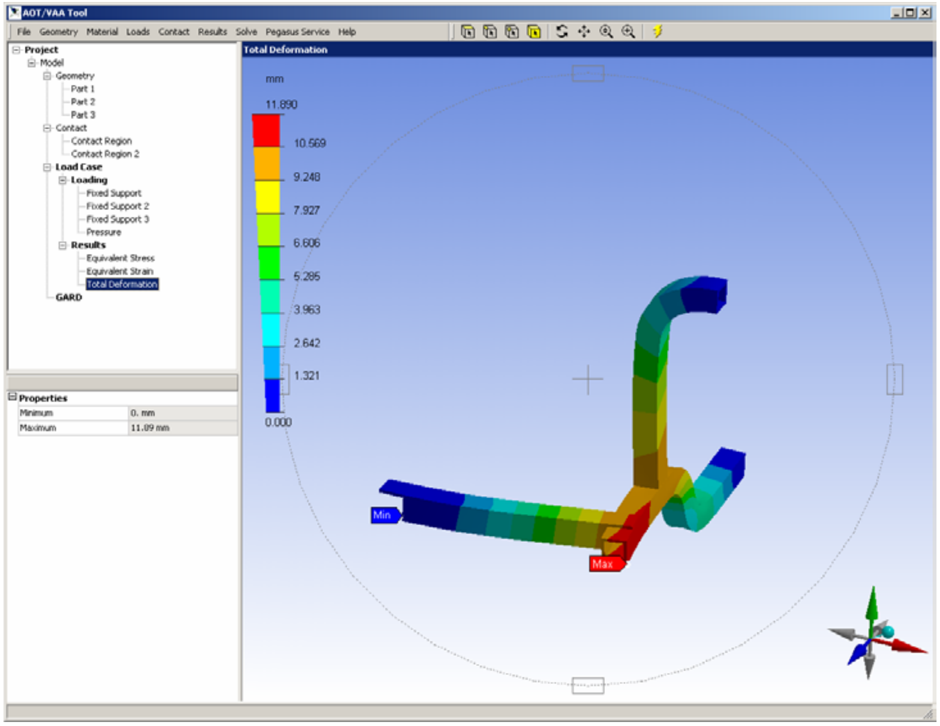


Figure 6-24 Impact test result for alternative A (welding) – total deformation

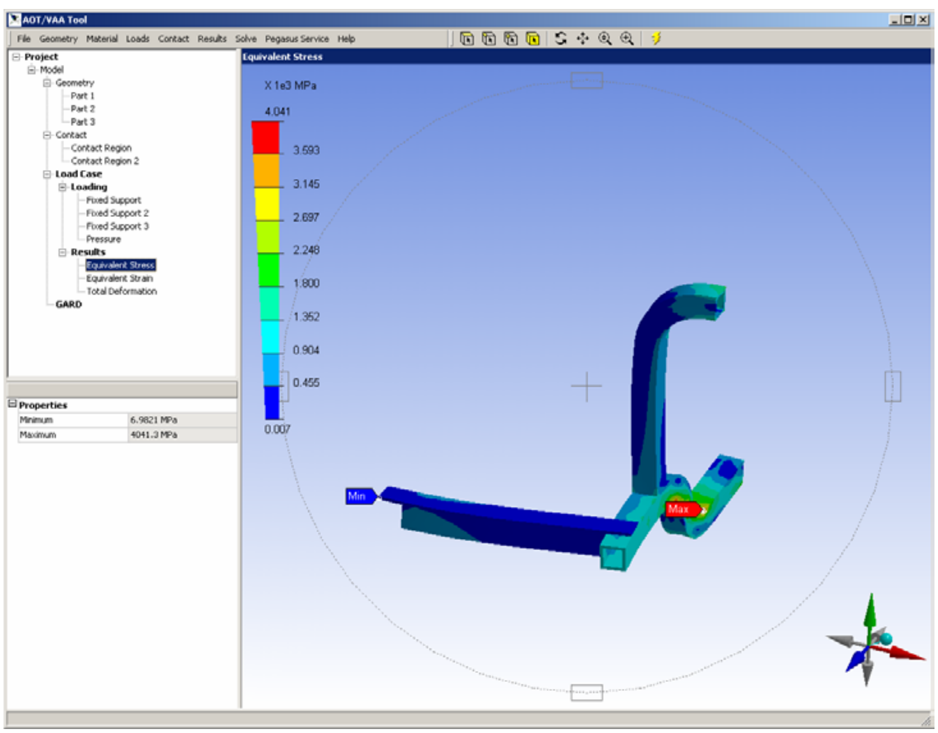


Figure 6-25 Impact test result for alternative A (welding) – equivalent stress

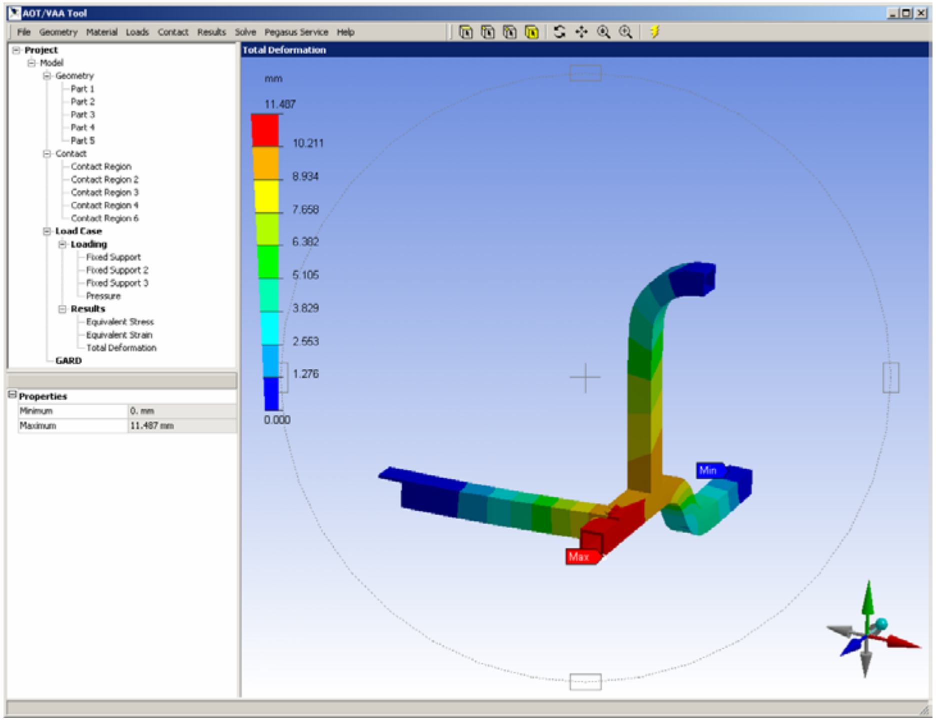


Figure 6-26 Impact test result for alternative B (riveting) – total deformation

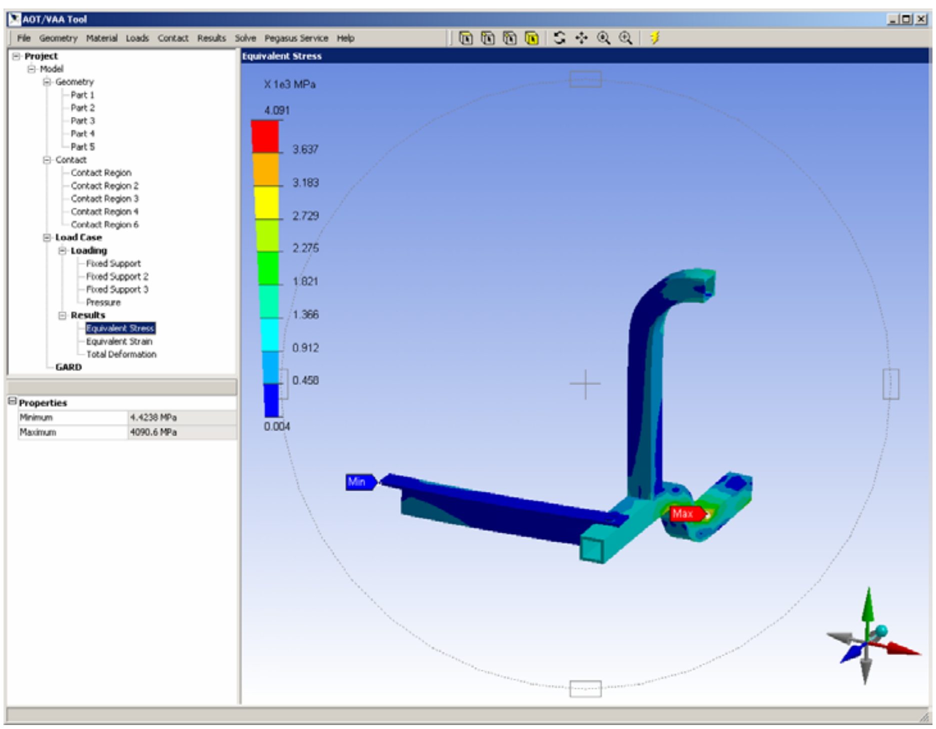


Figure 6-27 Impact test result for alternative B (riveting) – equivalent stress

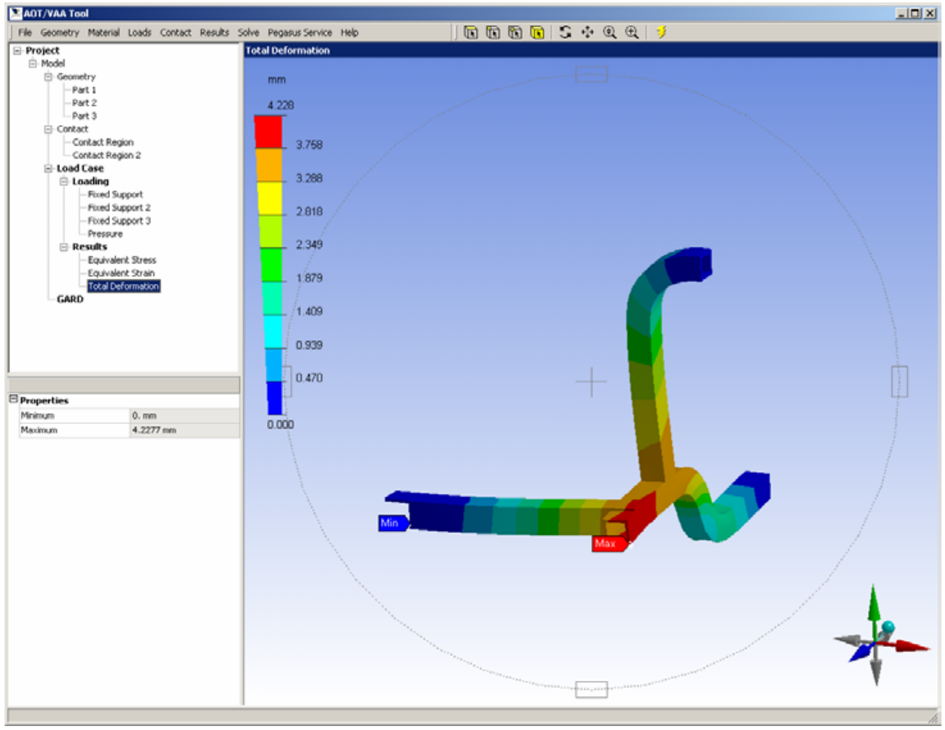


Figure 6-28 Impact test result for alternative C (welding) – total deformation

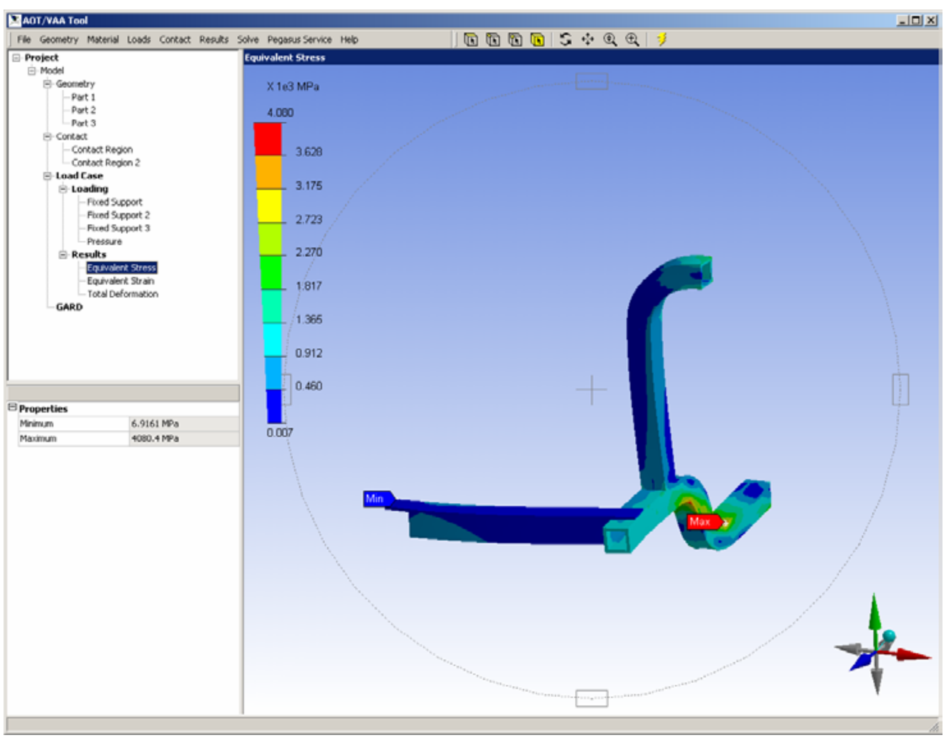


Figure 6-29 Impact test result for alternative C (welding) – equivalent stress

Figure 6-30 shows the AHP tool and the obtained result. The weights and local priorities are confirmed by the decision maker (designer). From this AHP analysis, AsD alternative B, using structural rivets, obtains the highest evaluation value (0.429) and alternative A, in which the welding condition is changed, has the second highest evaluation value (0.311). The alternative C, using different material, receives the lowest evaluation value (0.261). As shown in Figure 6-31, although the alternative B received the highest evaluation values for the cost criterion and quality criterion I, it received the lowest evaluation value in design criterion I (0.053) and in quality criterion II (0.091). Note that alternative A also received a quite competitive score (0.311). Figures 6-32 illustrates sensitivity analyses on different situations. For example, if the weights of the criteria are changed and the design criterion II is made most important, then the best alternative is changed to the alternative A (see Figure 6-32-b). By using this sensitivity analysis, the designer can consider various situations before making an AsD decision. Finally, the decision maker (designer) can make the following decision based upon this ADDM process:

1) Select alternative B as the best alternative and change the joining method from arc welding to riveting;
2) Consider alternative A as reasonable and select alternative A. In this case, a VAA process can be performed to improve the quality of the assembly (i.e., reduction of total deformation by controlling welding conditions);
3) Or, the designer can continue the ADDM process with alternative B as the current AsD.

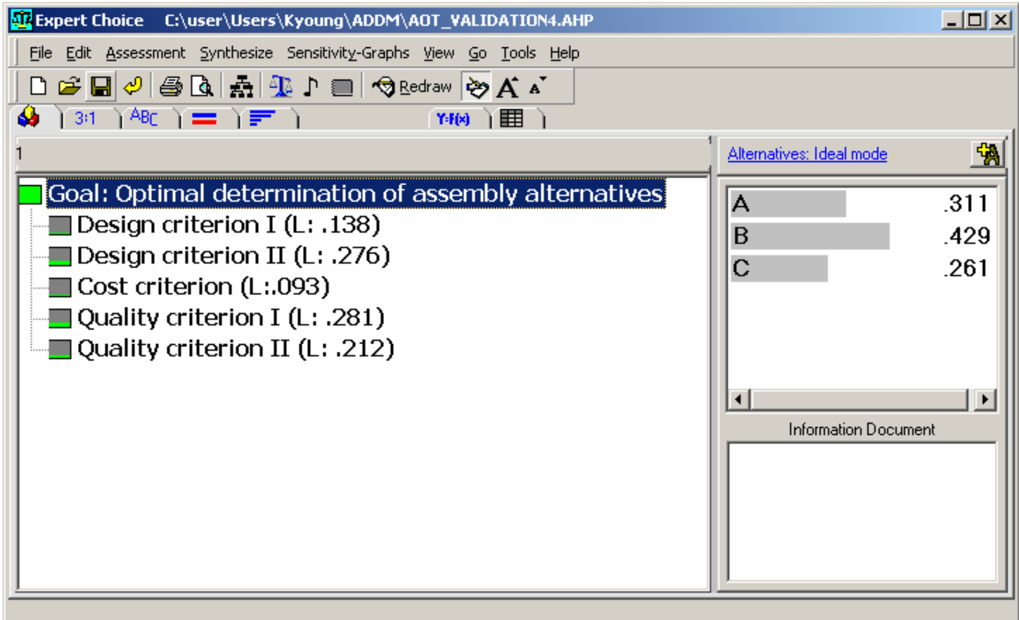
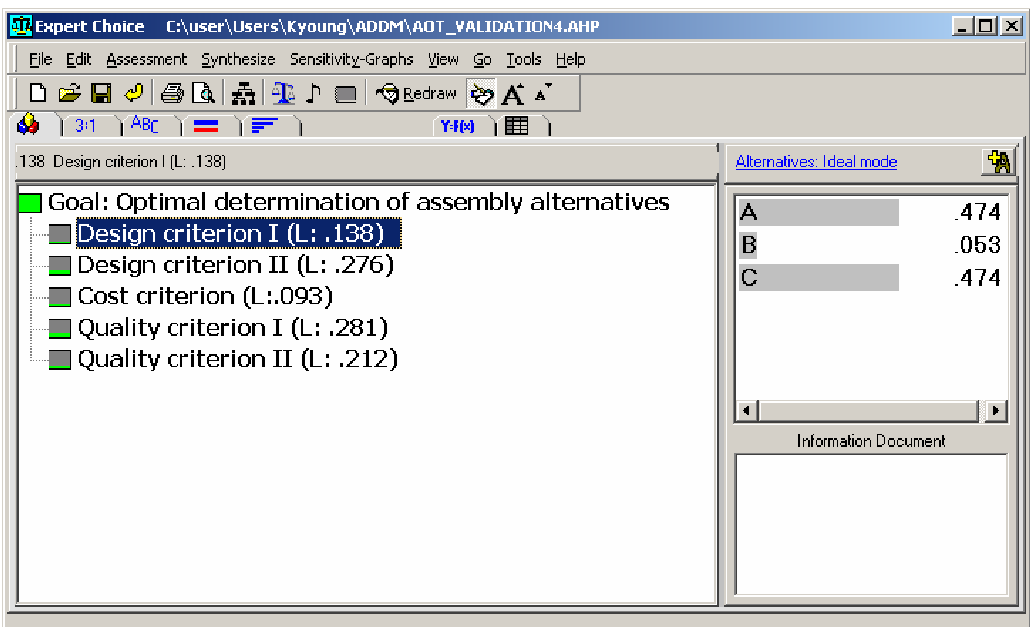
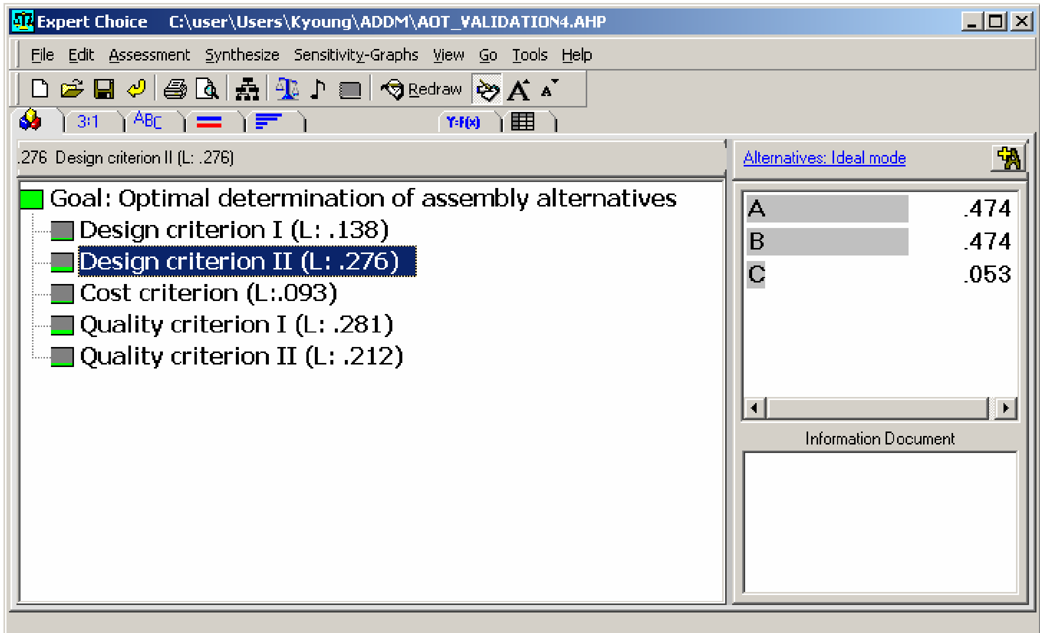


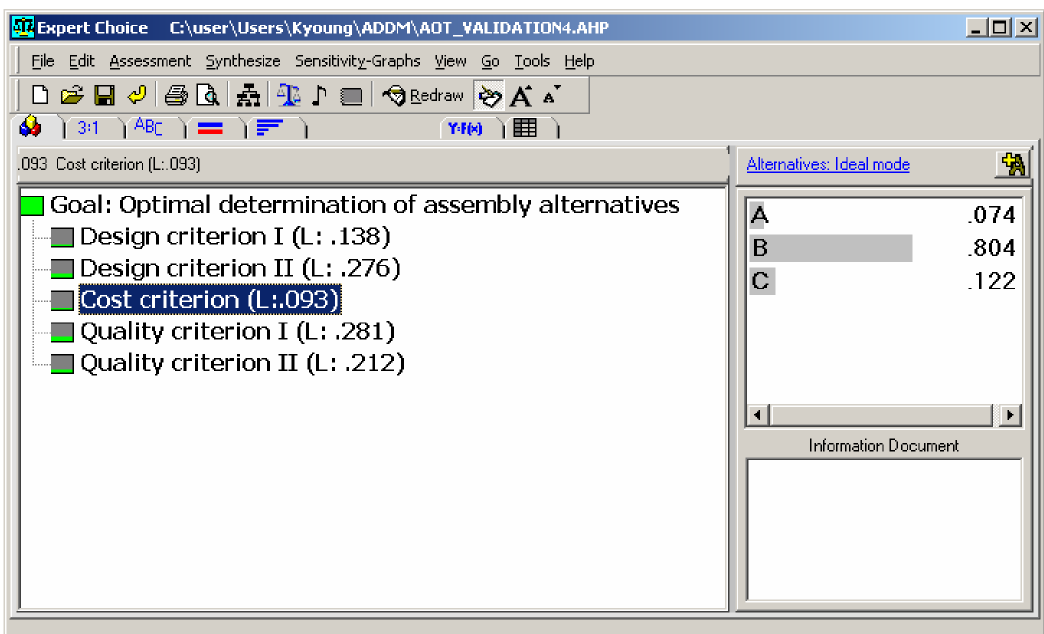
Figure 6-30 AHP result used for ADDM



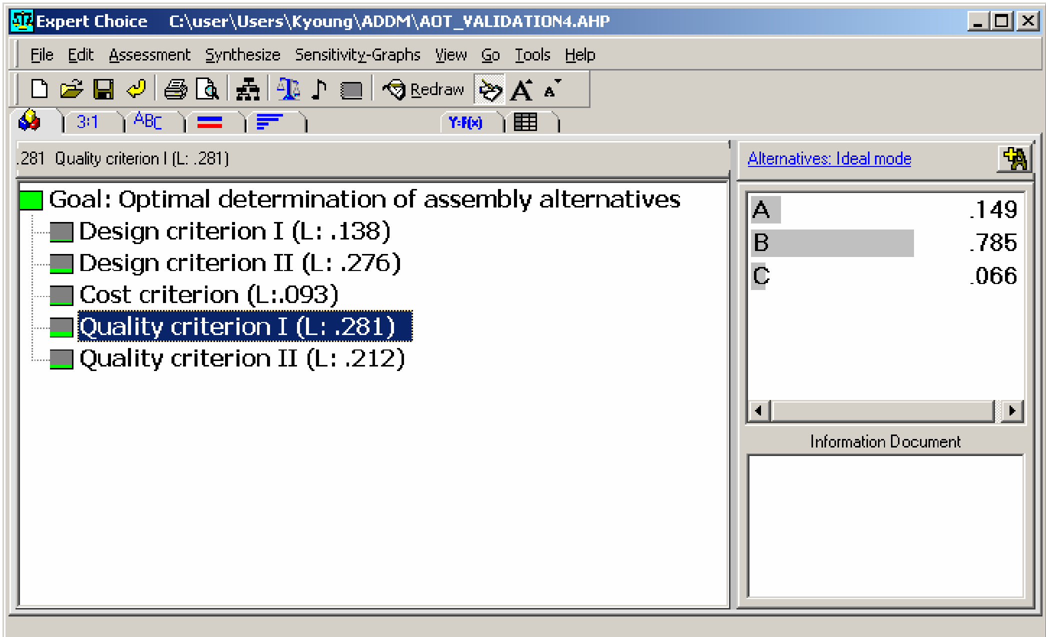
a) Design criterion I



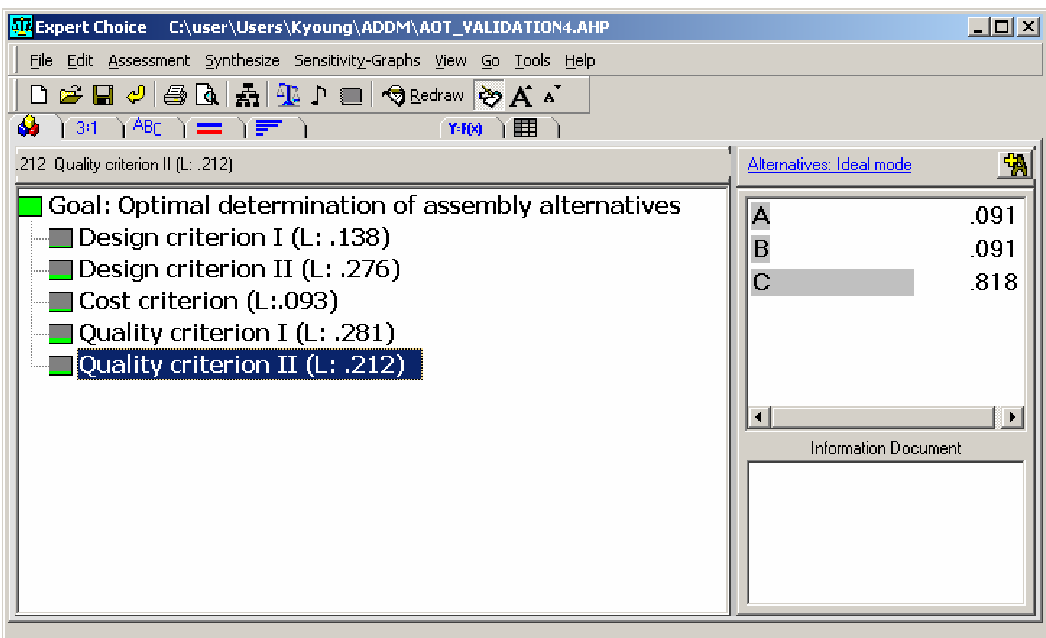
b) Design criterion II



c) Cost criterion

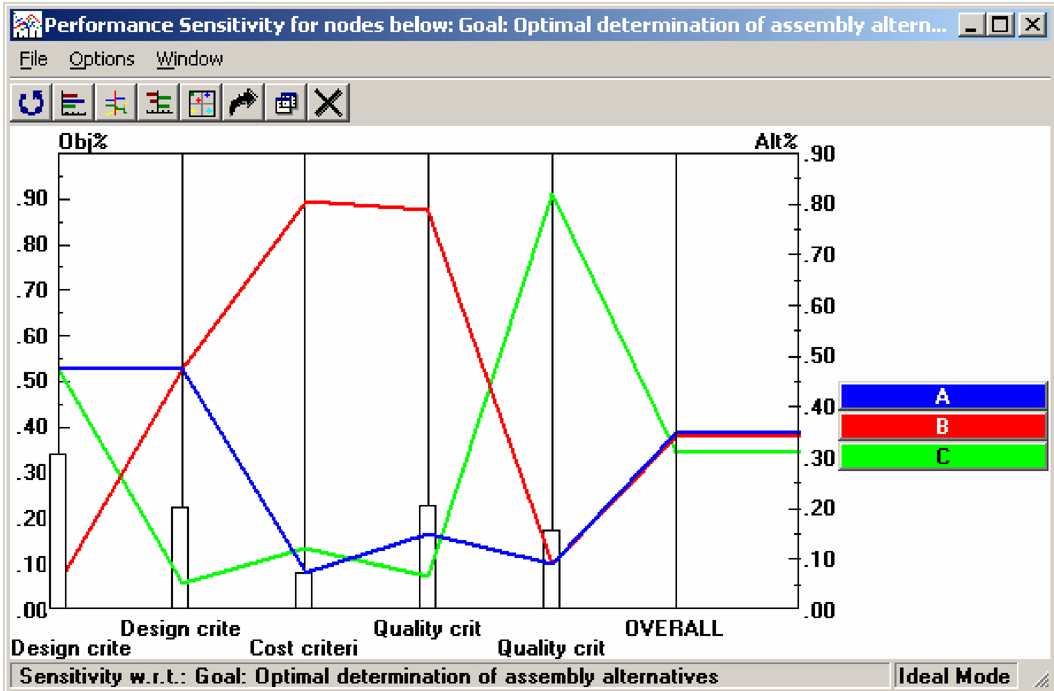


d) Quality criterion I

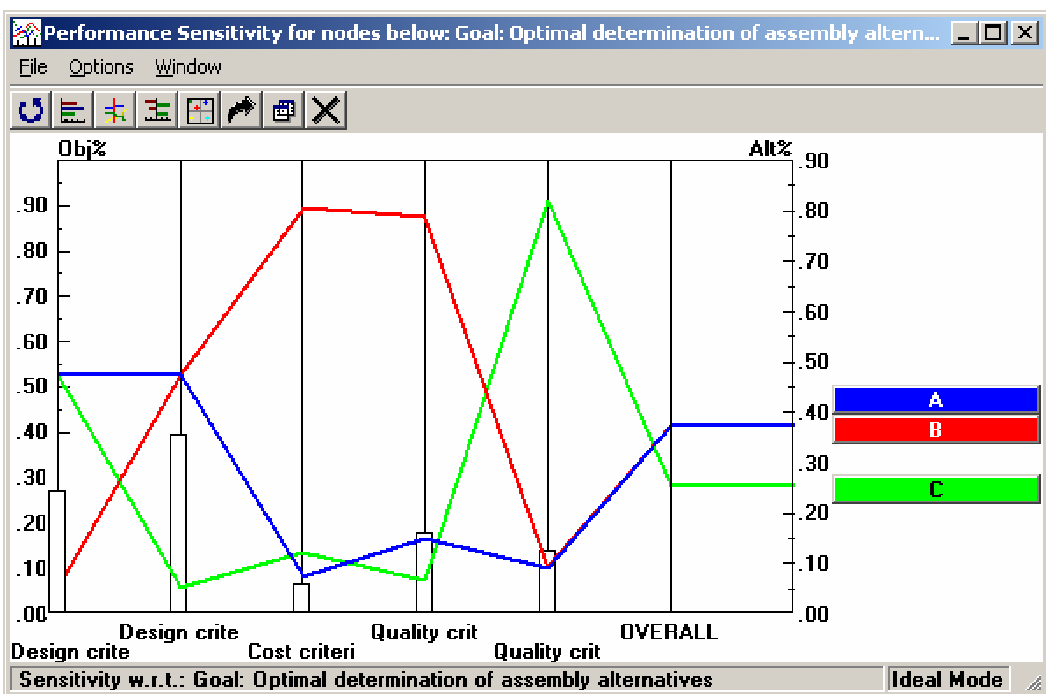


e) Quality criterion II

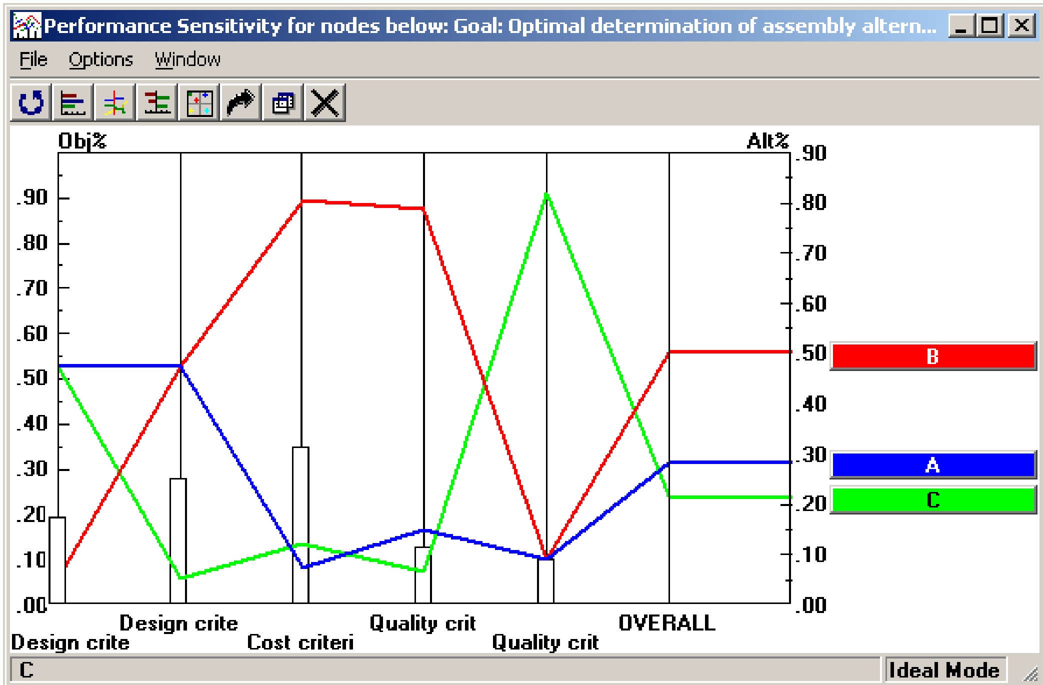
Figure 6-31 Local priority of alternatives



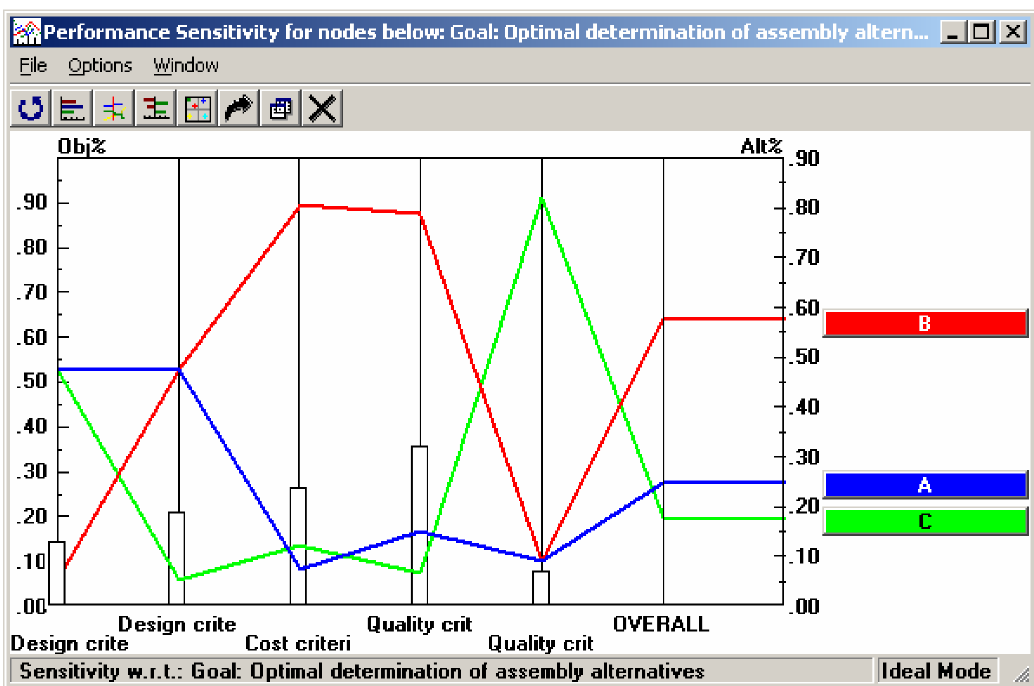
a) when design criterion I is most important



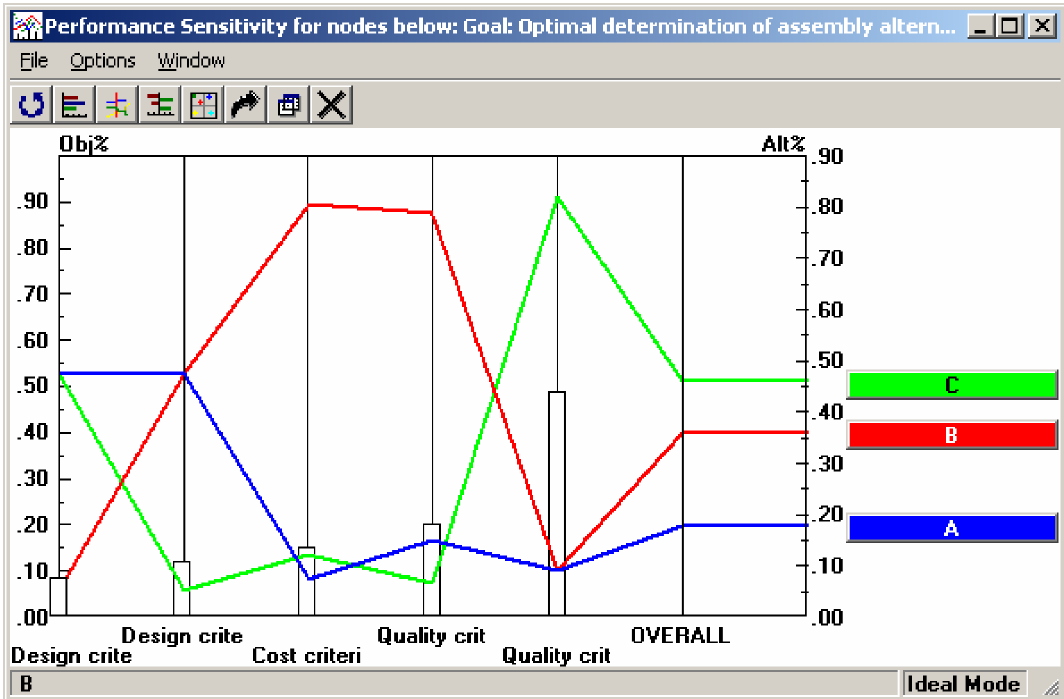
b) when design criterion II is most important



c) when cost criterion is most important



d) when quality criterion I is most important



e) when quality criterion II is most important

Figure 6-32 Sensitivity analysis

6.11 Experimental Study to Validate the VAA

In this work, the VAA concept and framework are validated by comparing the predicted VAA results and actual physical effects of the joints. The assembly scenario is that the thin bar is attached to a large extruded beam by welding and the rectangular beam is mounted on the top of the thin bar. Figure 6-34 illustrates the dimensions of the joint. Note that each dimension is reduced while keeping the aspect ratios to easily acquire test pieces for physical experiments and to keep a confidentiality agreement. Figure 6-35 shows boundary and loading conditions of the VAA and these conditions are applied to a physical test. Figures 6-36 and 6-37 show the predicted effects for the weld and rivet joints. To compare with the analytical results, actual aluminum alloy joints are fabricated as shown in Figures 6-38 and 6-39. In the physical test, a known mass is used to apply a force, which is the same as in the VAA, on the test-pieces. A CMM machine is used to measure deformation before loading and after loading. As shown in Table 6-2 the max deformations observed from the physical test and predicted by the VAA are quite similar.

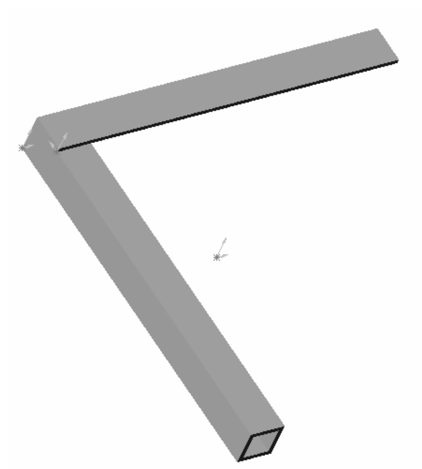


Figure 6-33 Elbow joint of extruded beams: 3D solid view

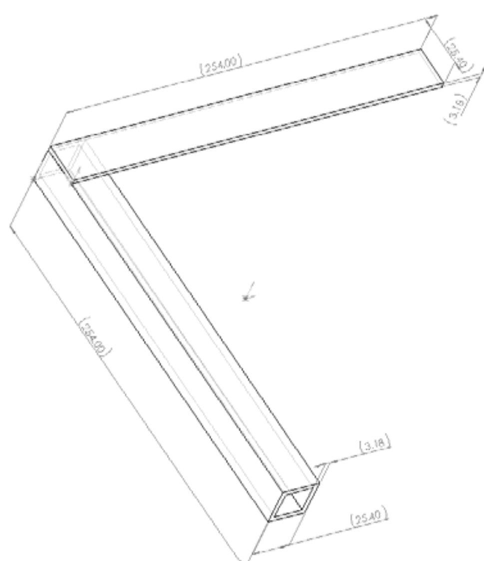


Figure 6-34 Elbow joint of extruded beams: 3D wire-frame view (mm)

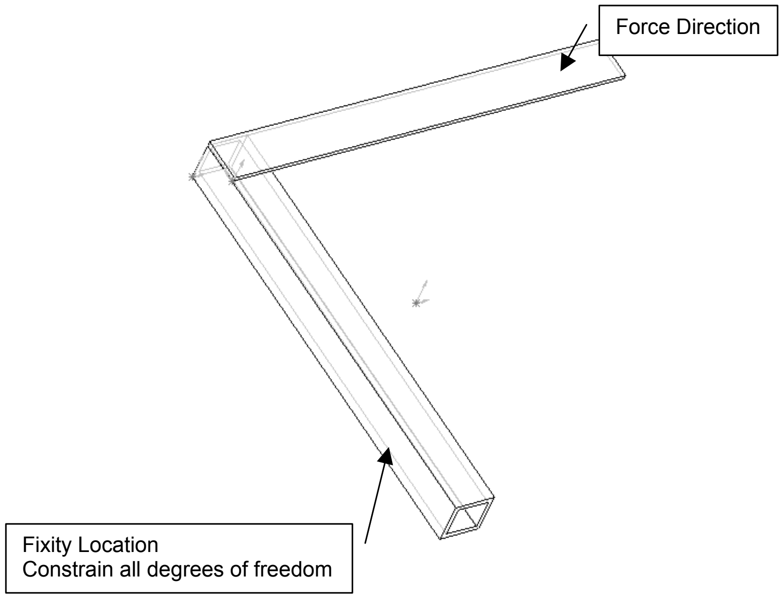
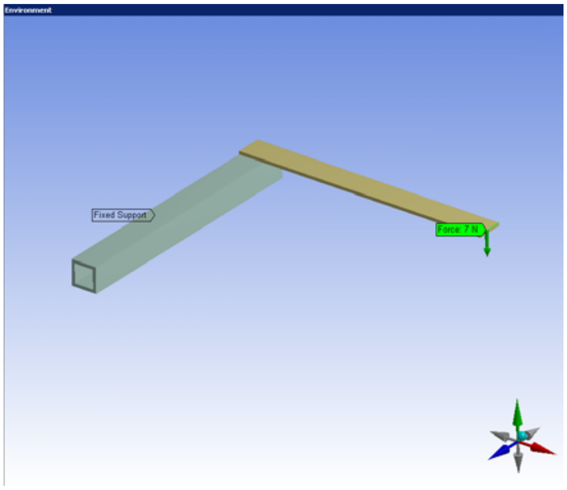
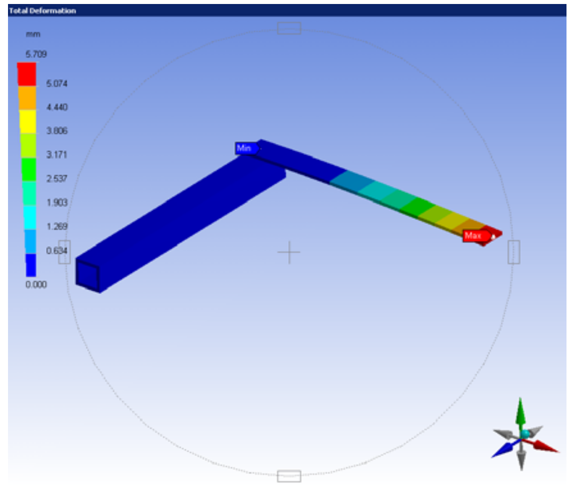


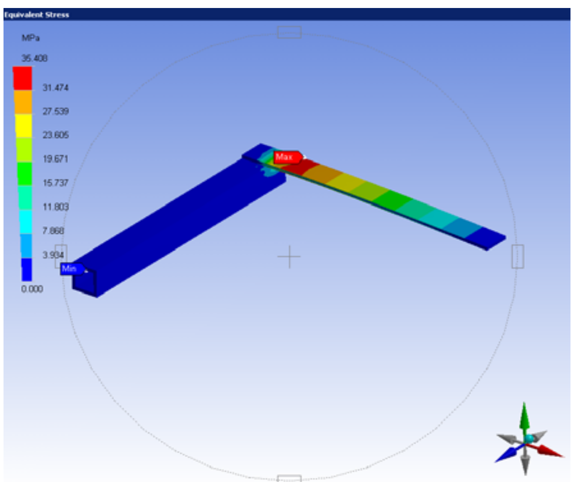
Figure 6-35 Analysis set-up



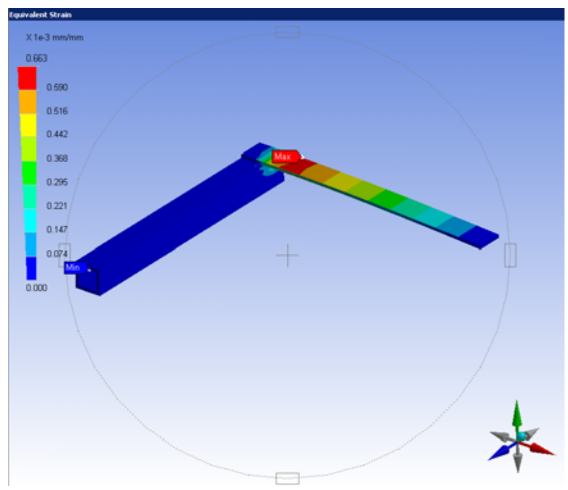
(a) AsAM



(b) Total deformation

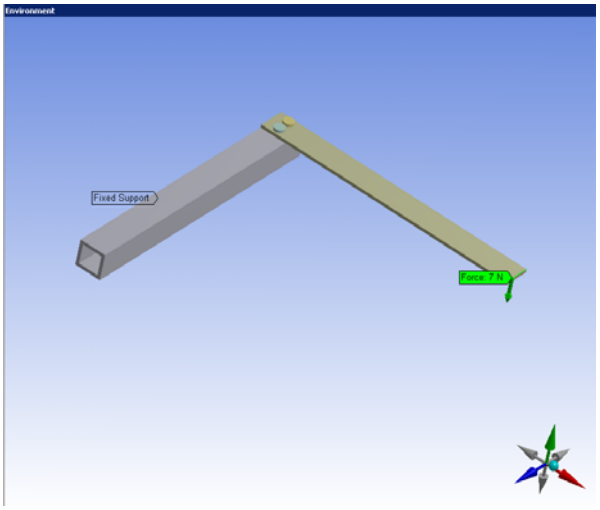


(c) Equivalent stress

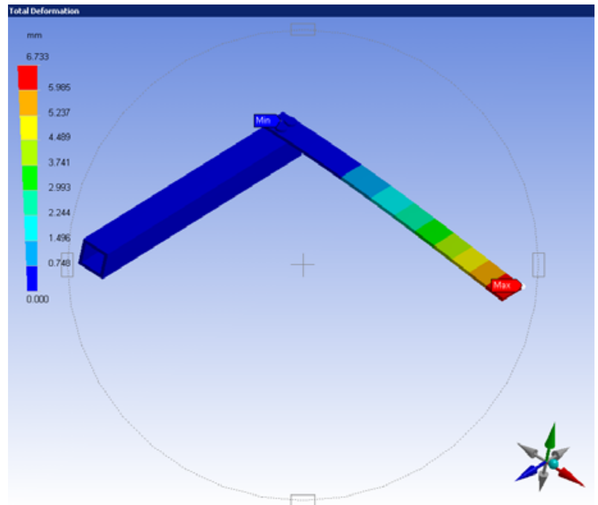


(d) Equivalent strain

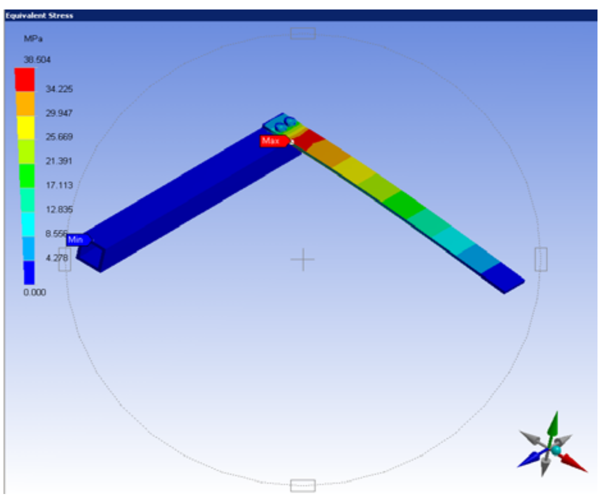
Figure 6-36 Predicted physical effects of the weld joint



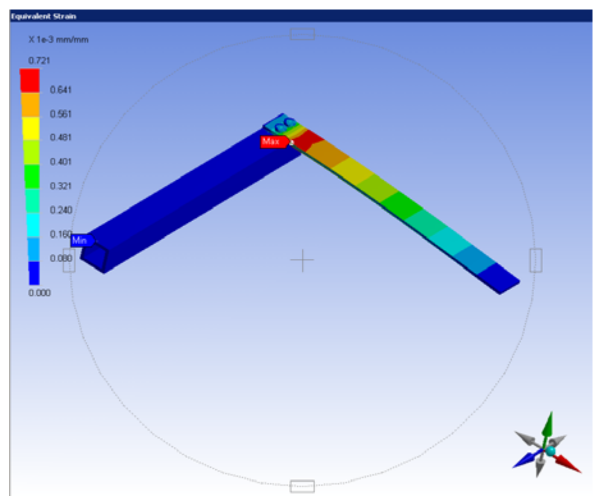
(a) AsAM



(b) Total deformation



(c) Equivalent stress



(d) Equivalent strain

Figure 6-37 Predicted physical effects of the rivet joint

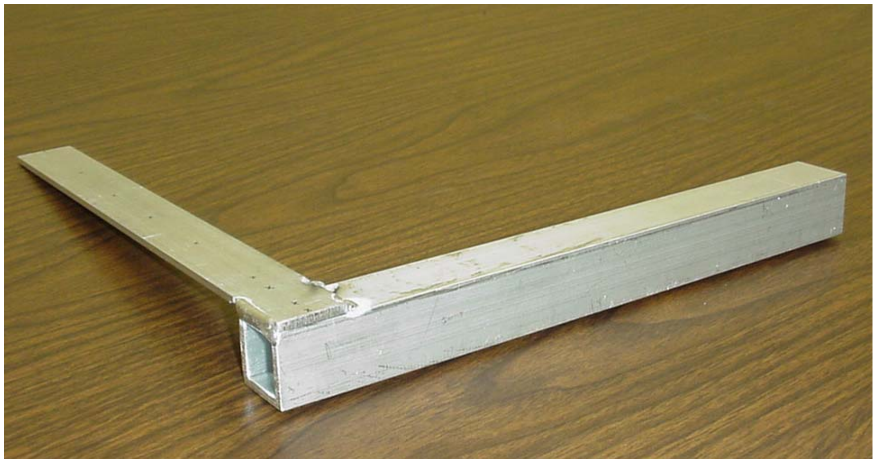


Figure 6-38 Welded joint for physical test

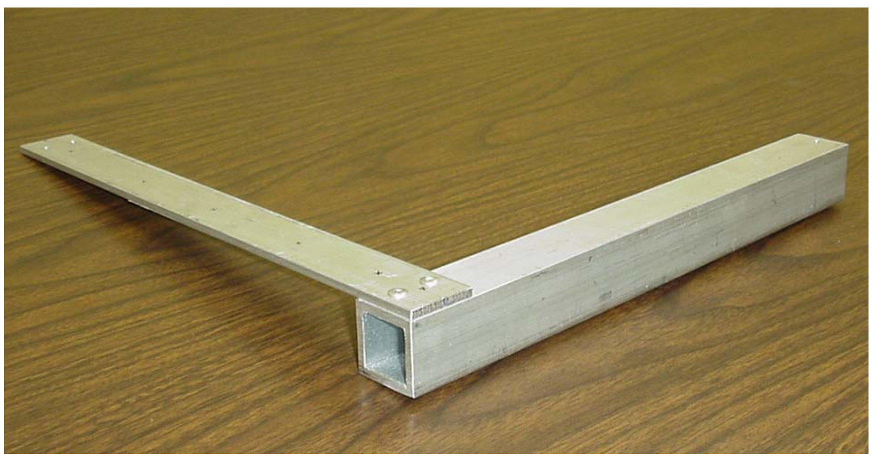


Figure 6-39 Rivet joint for physical test

Table 6-2 Comparison between VAA and physical test results

Joint	Expected max deformation (mm)	Actual max deformation (mm)
Weld joint	5.709	5.676
Rivet joint	6.733	6.794

6.12 Benefits Compared to Commercial CAD Packages

Commercial CAD systems have evolved into powerful designer aids in the development of mechanical products. Common CAD systems including AutoCAD, SolidWorks, Pro/Engineer, and CATIA, are considered state-of-art in the product design community. These CAD systems are compared to the capability of assembly operation tools as outlined in Sections 3 to 5 for evaluation and validation of the assembly design framework and methodology. The result of this comparison is summarized in Table 6-3. The tabulation shows that the developed assembly operation tools provide an environment in which assembly/joining relations are imposed on an AsD model and various core design activities including analysis, design intent analysis, and decision making are integrated in a collaborative design environment. The existing CAD systems can generate assemblies, but joining relations are not fully captured. Although some systems have massive tools for Product Data Management (PDM), it is still not clear how assembly/joining information can be captured persistently and concisely. Recently, commercial CAD companies including PTC, SolidWorks, and IBM have shown strong interest in the integration of CAD and CAE environments. As shown in Table 6-3, PTC's Pro/Engineer Simulation software and SolidWorks' COSMOS/Works provide a locally integrated analysis environment. CATIA's Tolerance Analysis of deformable Assembly (TAA) workbench presents technology integrating assembly design and joining analysis for some joining methods (CATIA 2003). Nonetheless, these workbenches still require all tools to reside in house and has a limitation to realize a transparent and remote analysis for collaborative assembly design and analysis. SolidWorks, Pro/Engineer, and CATIA have their own integrated analysis tools.

Table 6-3 Assembly operation tools capability versus existing commercial CAD systems

Comparison Measure	Commercial CAD systems				Assembly Operation Tools
	AutoCAD	SolidWorks	Pro/Engineer	CATIA	
Assembly/joining relation capture	Not available	Limited support	Limited support	Limited support	Supported
Lean assembly/joining information exchange	Not available	Limited support (SMARTEAM as PDM solution)	Limited support (Winchill CAD Integrations as PDM solution)	Limited support (ENOVIA as PDM solution)	Supported (through GARD and XML data)
Designer's intent analysis on SRI	Not available	Not available	Not available	Not available	Supported
Transparent and remote analysis	Not available	Limited support (COSMOS/Works provides locally integrated analysis)	Limited support (Pro/Engineer Simulation software provides locally integrated analysis)	Limited support (TAA Workbench provides locally integrated analysis)	Supported
Assembly/joining knowledge capture	Not available	Not available	Not available	Not available	Supported
Assembly design decision support	Not available	Not available	Not available	Not available	Supported

7.0 CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

The aim of this work is to develop a set of assembly design and virtual analysis tools and innovative concepts for joining to be used in an *e*-product design and realization environment. An intelligent assembly design system should be able to assist a designer during the product joining design process by predicting expected assembly design problems and providing alternative suggestions. Such a system should have the capability of employing spatial relationships and joining protocols that result in the physical realization of an assembly. Traditional solid modeling systems for assembly components, while adequate for visualization purposes, do not support downstream lifecycle activities. Furthermore, existing CAD systems are unable to show the physical and mechanical implications of an assembly operation, such as the thermo-mechanical effect of a weld. Currently, the effect of joining is analyzed upon completion of assembly modeling. This sequential process is arduous and time-consuming.

In this dissertation work, a set of engineering tools is developed to improve product assembly design processes. Through this work, an assembly design formalism and assembly operation tools were developed to enable an IT-enabled collaborative product assembly design environment. The assembly operation tools include the *AsD engine*, *AsI engine*, and *AsA engine*. The AsD formalism captures product assembly/joining information and allows the specification of the joining relations symbolically; it enables transparent information flow in an overall assembly product development lifecycle. This formalism was implemented in the AsD engine. The spatial relationship implications of joining were mathematically captured for realistic assembly designs. The relationships between designed d.o.f. and implied d.o.f. of selected joining methods are investigated and implemented in the SRI tool. A new VAA framework was developed to integrate assembly design and analysis transparently and remotely. The VAA tool and a service-oriented architecture were developed and implemented in the VAA framework. The VAA processes are validated by physical tests. To support assembly design decision making, a new decision-making method, ADDM, is developed. ADDM proposes assembly alternatives to the designer by considering assembly design and implication information, assembly/joining knowledge, and material knowledge. A HSN model was proposed as a core model to represent evaluation knowledge and assembly design knowledge, which is inevitable in knowledge based design decision making. This ADDM framework was implemented in the AsA engine.

This work provides a set of assembly operation tools that supports the mechanical product industry in fast and efficient product design. This set of tools serves as a "plug-and-play" module in an *e*-designer system (Pegasus) under development at the Center for *e*-Design at the University of Pittsburgh. This research extends current assembly design to include realistic joint design and realization by considering physical effects/implications.

7.2 Future Work

This research provides the basic frameworks and methodologies for assembly design considering joining. As a case study, a set of joining methods (i.e., arc welding and riveting) are selected and investigated. Future research will extend these methodologies to include all joining methods. It is also possible to extend the results of this work to the design of other engineering products (other than the current restriction to mechanical assemblies). Details of future research are described below.

7.2.1 Integration with Existing CAD Systems

In the developed assembly operation tools architecture, information is transferred and exchanged among system components by using XML formats. Each assembly operation tool may be integrated into existing CAD systems by the use of XML geometric ID tags. The geometric ID tags are unique ID numbers assigned to each geometric entity, which are involved in AsD model generation, AsAM generation, and ADDM setup file generation.

In this work, the ACIS kernel is used as a demonstration of the integration with commercial CAD formats. The implementation of this work in the ACIS kernel requires the use of special customization features provided in the ACIS architecture to support imposition of assembly information. ACIS is an object-oriented 3D geometric modeling engine from Spatial Technology Inc. It is designed for use as the geometric foundation within virtually any end user 3D modeling application. The ACIS model representation consists of various geometric and topologic entities, as well as attributes that may be attached to the entities. The model is implemented in C++ using a hierarchy of classes. All geometric entities specified in the XML data are linked to a solid model. In an ACIS solid model, the attribute ID is used as a linkage tag. This AsD model's XML data goes together with geometric data (solid model) in AsD data transitions. It allows functionality information to be persistently captured in a CAD design environment. Similarly, other CAD kernel formats need to be investigated for the integration with the assembly operation tools. In the future, the Parasolid kernel will be studied and integrated.

7.2.2 Integration of Assembly Design and Analysis

In this work, pre-determined FEA procedures are used to predict the physical effects, such as displacement and residual stress, of the joints. Determining an appropriate joining analysis procedure is very important for obtaining realistic analysis results. Detailed and realistic analysis methodology can provide in-depth information of joining behavior to a designer. As a case study, a set of joining methods (i.e., arc welding and riveting) are selected and investigated in this work. To realize VAA for additional joining methods, various issues need to be investigated, such as mapping between the assembly design model and the assembly analysis model, and determination of efficient and realistic joining analysis methods considering the joining characteristics. Especially various actual situations in joining need to be considered. For example, in arc welding, welding directions, welding sequences, and weld layers are important considerations. The developed service-oriented architecture provides an environment, in which new analysis procedures are easily acquired from remote analysis procedure service providers. In future research, more analysis procedures will be investigated and acquired to provide more analysis flexibility. Remote, comprehensive libraries of joining analysis procedures will be developed.

7.2.3 Extension of ADDM

In this work, a new method to resolve the assembly design decision (ADD) problem, called ADDM was introduced. The ADD problem is a multicriteria and knowledge-based problem in nature. As a core model to represent evaluation knowledge and assembly design knowledge, a hierarchical semantic net (HSN) model is introduced. In the HSN model, design alternatives are evaluated based upon the evaluation models and the obtained values in return are dynamically added to the semantic net of alternatives. This method allows independence between evaluation knowledge and design knowledge. In other words, the addition of new criterion will not affect the evaluation of design alternatives. It also enables the ADDM system to be scalable and extendable. In the current validation, two design specifications (i.e., max total deformation and yield stress), three AsD criteria (i.e., design, cost, quality), and several corresponding evaluation models are considered. In the future validation, more complex and realistic design specifications and criteria will be employed. For example, design evaluation models considering ergonomics and safety can be inserted for AsD evaluation. Functional requirements can be used as AsD specifications to ensure the fulfillment of the functionality.

The ADDM framework can be extended to various domains. For example, as an extension of ADDM, a framework of multimedia decision support for assembly design can be developed. In collaboration, multimedia information/data acts as a very important tool to share product design and engineering knowledge between project participants. Existing collaboration tools (FIPER 2001, OneSpace 2003) have employed multimedia for an efficient communication protocol. In contrast to text-based presentations, multimedia presentations empirically have been shown to reduce the influence of first impression bias (Lim *et al*. 2000).

With the advancement in multimedia technology, it has become popular and efficient for collaborators to share information using multimedia in product design. However, current multimedia in collaboration provides only a communication protocol and has a limitation to capture any engineering information imposed on product design. Although some research including this work has been done to integrate product design tools seamlessly, it has not fully addressed how to extract and exchange engineering design information imposed in multimedia. To design and analyze a new product from scratch is knowledge-intensive and hence, very costly. For example, the result of a mechanical analysis stored as an image file, such as JPEG format, is stripped of all engineering information regarding the product and analysis procedures/outcomes. Hence, reusing past design knowledge, if any, may improve the productivity of the engineering design decision-making. Future ADD system needs to handle images, engineering data, and models as one form, such as a graphical form. By using this graphical from, any type of users such as customers, suppliers, designers or modelers can receive and send their own information with a unique and standard form, which may increase the sharability that is crucial for collaborative product design.

In the future, a new framework of multimedia decision support for assembly design will be developed. In this framework, design participants can collaborate with each other with the aid of multimedia. The multimedia can be linked to remote and distributed engineering information/data, which are core sources of assembly design decision-making. In doing so, rather than directly interfacing and sharing engineering information/data, the new framework will provide the design participants with a seamless (and user-friendly) way of collaboration. In other words, while design participants are interfacing with multimedia formats, the participant does not need to understand details about engineering data, such as data format and internal logic.

7.2.4 Extension to Commercial Product Level

The work in this research has focused on the development of assembly/joint design concepts necessary for considering joining and its effects in an *e*-product design and realization environment. The implementation has been restricted to the demonstration of the concepts developed in this research. To advance the assembly operation tools to a commercial product level, the following additional tasks need to be performed. Once these technologies are fully realized, a commercial level software providing functionality of the AOT is expected to be shown in market within a three or four year time span.

- Develop an ontology server to handle the consistent use of XML tags as means of transferring/exchanging assembly design information. XML data is a universal media to exchange assembly design information. The ontology server can manage consistency of XML syntax for collaboration.
- Standard ways of the World Wide Web Consortium (W3C) can be considered to specify the structure of AsD models and the data type of each element and attribute in XML. Currently, two standard ways are widely used: Data Type Definition (DTD) that is inherited from Standard Generalized Markup Language (SGML), and Schema. The DTD and Schemas are defined according to W3C's documentations (W3C DTD 2000, W3C Schema 2001).
- Support various CAD kernel formats and proprietary CAD formats for AsD modeling generation and AsAM generation.

- Extend joint features for various other joining methods, including adhesive bonding.
- Develop an analysis procedure library as a service provider for various joining methods and conditions.
- Interface with commercial database and knowledge base systems to obtain various domain information and knowledge including material, manufacturing, assembly, cost model, ergonomics, and safety.
- Obtain APIs to generate AsAM needed to allow the ability to seamlessly impose various constraints on various entities in AsAM. Currently, the VAA tool is based on the AI*Workbench architecture. With close collaboration with ANSYS, Inc., the interface among the VAA tool based on the AI*Workbench architecture and external CAD kernels can be improved to prevent information distortion.
- Various output display modes including animation and image formats need to be provided to allow for easy, user-friendly visualization of VAA results.
- Develop advanced mechanisms for acquiring external knowledge, which represents environmental situations including corporate policy, market conditions, and federal regulations for ADDM.
- The computer graphic capability of the GARD tool needs to be changed by implementing the tool as an independent package (that is outside of the Visio software).

The usability of the developed AOT technologies needs to be tested in actual industry situation. In the future, companies supporting the Center for e-Design at the University of Pittsburgh will provide test-sites for the developed frameworks and AOT.

APPENDICES

APPENDIX A

AsD CLASSES

Appendix A.1 Assembly Class

```
class CADFAssembly
{
public:
	CADFAssembly();
	virtual ~CADFAssembly();

	char assy_id[20];

	CADFPart adfPrt[10];
	CADFAssemblyFeature adfAF[10];
};
```

Appendix A.2 Part Class

```
class CADFPart
{
public:
	CADFPart();
	virtual ~CADFPart();

	char *prt_id;

	CADFFormFeature adfFF[10];

};
```

Appendix A.3 Assembly Feature Class

```
class CADFAssemblyFeature
{
public:
        CADFAssemblyFeature();
        virtual ~CADFAssemblyFeature();

        char af_id[20];
        char af_partID[10][20];
        char af_material[10][30];
        char af_srDOF[30];
        char af_impConstraint[30];

        CADFMatingFeature adfMF[5];
        CADFMatingBond adfMB[5];
        CADFJointFeature adfJF[5];
};
```

Appendix A.4 Form Feature Class

```
class CADFFormFeature
{
public:
	CADFFormFeature();
	virtual ~CADFFormFeature();


	char *ff_id;
};
```

Appendix A.5 Mating Feature Class

```
class CADFMatingFeature
{
public:
	CADFMatingFeature();
	virtual ~CADFMatingFeature();

	char mf_id[20];
	char mf_mComponent[2][20];
	char mf_mEntity[2][20];

	CADFSpatialRelationship adfSR;

};
```

Appendix A.6 Joint Feature Class

```
class CADFJointFeature
{
public:
	CADFJointFeature();
	virtual ~CADFJointFeature();

	char jf_jId[20];
	char jf_jType[10];
	char jf_jMethod[40];
	char jf_jGrooveShape[20];
	char jf_jComponent[10][20];
	char jf_jEntity[10][20];
	char jf_jConstraint[20][20];

};
```

Appendix A.7 Mating Bond Class

```
class CADFMatingBond
{
public:
	CADFMatingBond();
	virtual ~CADFMatingBond();

	char mb_id[20];

	CADFMatingPair adfMP;
	CADFMatingCondition adfMC;

};
```

Appendix A.8 Mating Condition Class

```
class CADFMatingCondition
{
public:
        CADFMatingCondition();
        virtual ~CADFMatingCondition();

        char mc_name[20];
        char mc_srName[20];
        char mc_srDOF[10][20];
        char mc_intraFeatureRelation[20];
        char mc_constraint[10][20];

};
```

Appendix A.9 Mating Pair Class

```
class CADFMatingPair
{
public:
	CADFMatingPair();
	virtual ~CADFMatingPair();

	char mp_id[20];
	char mp_mEntityID[2][20];
	char mp_fFeatureID[2][20];
	char mp_interFeatureRel[10][20];
	char mp_dConstraint[10][20];
};
```

Appendix A.10 Spatial Relationship Class

```
class CADFSpatialRelationship
{
public:
	//void updateSR(CADFAssembly assy);
	CADFSpatialRelationship();
	virtual ~CADFSpatialRelationship();

	char sr_id[20];
	char sr_name[20];
	char sr_entity[10][20];
	char sr_dof[10][50];

};
```

Appendix A.11 Entity Indexer Class

```
class CADFEntityIndexer
{
public:

	CADFEntityIndexer();
	virtual ~CADFEntityIndexer();

	char m_indexAttr[10][20];
	char m_bodyAttr[10][20];
	char* addEntityIndex(ENTITY* entity);

	void setEntityIndex(HSSelectionSet* select);
	void setBodyIndex(HSSelectionSet* select);
};
```

Appendix A.12 XML Data Class

```
class CADFXMLData
{
public:
        void generateXML();
        CADFXMLData();
        virtual ~CADFXMLData();

};
```

APPENDIX B

XML DATA

Appendix B.1 AsD Model's XML Data


```
<?xml version="1.0" ?>
<!-- AsD XML Description By Assembly Design Formalism ! -->
<ASD>
   <info>
         <name> adf10545740085 </name>
         <unit> SI-millimeter </unit>
         <description>  </description>
   </info>
   <AF>
         <name> adfaf10545740098 </name>
         <MF>
                  <MF-ID> adfmf10545740112 </MF-ID>
                  <SR> against </SR>
                  <mating-component> body10427394300 </mating-component>
                           <mating-entity> face10545740114 </mating-entity>
                  <mating-component> body10427394301 </mating-component>
                           <mating-entity> face10545740115 </mating-entity>
         </MF>
         <MF>
                  <MF-ID> adfmf10545740109 </MF-ID>
                  <SR> aligned </SR>
                  <mating-component> body10427394300 </mating-component>
                           <mating-entity> edge10545740110 </mating-entity>
                  <mating-component> body10427394301 </mating-component>
                           <mating-entity> edge10545740111 </mating-entity>
         </MF>
         <MF>
                  <MF-ID> adfmf10545740116 </MF-ID>
                  <SR> aligned </SR>
                  <mating-component> body10427394300 </mating-component>
                           <mating-entity> edge10545740117 </mating-entity>
                  <mating-component> body10427394301</mating-component>
                           <mating-entity> edge10545740119 </mating-entity>
         </MF>
         <JF>
                  <JF-ID> adfjf10545740121 </JF-ID>
                  <joining-method>  GMAW </joining-method>
                  <joining-component> body10427394300 </joining-component>
```

```
            <joining-entity> face10545740114 </joining-entity>
        <joining-component> body10427394301 </joining-component>
            <joining-entity> edge10545743100 </joining-entity>
        <groove-shape>  No groove </groove-shape>
        <joining-constraint>
            <welding-condition>
                <amperage> 30 </amperage>
                <voltage> 220 </voltage>
                <feedrate> 3 </feedrate>
                <weld-speed> 2 </weld-speed>
                <weaving> No Weaving </weaving>
            </welding-condition>
            <fixture-location>
                <id> face10545744130 </id>
                <id> face10545744131 </id>
                <id> face10545744132 </id>
            </fixture-location>
        </joining-constraint>
        <tolerance>
            <max-var-straightness> 2 </max-var-straightness>
        </tolerance>
    </JF>
    <MB>
        <MB-ID> adfmb10545745213 </MB-ID>
        <mating-pair>
            <mating-feature>
            <ID> face10545740114</ID>
                <form-feature>
                    <ID> body10427394300 </ID>
                    <inter-feature-association> J:body10427323479
                        </inter-feature-association>
                    <inter-feature-association> J:body10427395141
                        </inter-feature-association>
                    <dimensional-constraint>
                        <Width> 25.40 </Width>
                        <Height> 25.40 </Height>
                        <Length> 350.03 </Length>
                    </dimensional-constraint>
                </form-feature>
            </mating-feature>
            <mating-feature>
            <ID> face10545740115 </ID>
                <form-feature>
                    <ID> body10427394301 </ID>
                    <inter-feature-association>
                        </inter-feature-association>
                    <dimensional-constraint>
                        <Width> 25.40 </Width>
                        <Height> 28.58 </Height>
                        <Length> 254.00 </Length>
                    </dimensional-constraint>
                </form-feature>
            </mating-feature>
        </mating-pair>
        <mating-condition>
            <assembly-joining-relation>
```

<form-feature> body10427394300 </form-feature>
<form-feature> body10427394301 </form-feature>
</assembly-joining-relations>
<SR> against </SR>
<transformed-geometric-constraint> on_surface
</transformed-geometric-constraint>
<dof> {plane_z::rot_z} </dof>
<implied-constraint> </implied-constraint>
</mating-condition>
</MB>
<MB>
<MB-ID> adfmb10545745214 </MB-ID>
<mating-pair>
<mating-feature>
<ID> edge10545740110 </ID>
<form-feature>
<ID> body10427394300 </ID>
<inter-feature-association> J:body10427323479
</inter-feature-association>
<inter-feature-association> J:body10427395141
</inter-feature-association>
<dimensional-constraint>
<Width> 25.40 </Width>
<Height> 25.40 </Height>
<Length> 363.03 </Length>
</dimensional-constraint>
</form-feature>
</mating-feature>
<mating-feature>
<ID> edge10545740111</ID>
<form-feature>
<ID> body10427394301 </ID>
<inter-feature- association >
</inter-feature- association >
<dimensional-constraint>
<Width> 25.40 </Width>
<Height> 28.58 </Height>
<Length> 254.00 </Length>
</dimensional-constraint>
</form-feature>
</mating-feature>
</mating-pair>
<mating-condition>
<assembly-joining-relation>
<form-feature> body10427394300 </form-feature>
<form-feature> body10427394301 </form-feature>
</assembly-joining-relations>
<SR> aligned </SR>
<transformed-geometric-constraint> on_line & parallel
</transformed-geometric-constraint>
<dof> {lin_ edge10545740110::lin_ edge10545740111}
</dof>
<implied-constraint> </implied-constraint>
</mating-condition>
</MB>
<MB>

```
<MB-ID> adfmb10545745215 </MB-ID>
<mating-pair>
        <mating-feature>
        <ID> edge10545740117 </ID>
                <form-feature>
                        <ID> body10427394300 </ID>
                        <inter-feature-association> J:body10427323479
                                </inter-feature-association>
                        <inter-feature-association> J:body10427395141
                                </inter-feature-association>
                        <dimensional-constraint>
                                <Width> 25.40 </Width>
                                <Height> 25.40 </Height>
                                <Length> 363.03 </Length>
                        </dimensional-constraint>
                </form-feature>
        </mating-feature>
        <mating-feature>
        <ID> edge10545740119 </ID>
                <form-feature>
                        <ID> body10427394301 </ID>
                        <inter-feature- association >
                                </inter-feature- association >
                        <dimensional-constraint>
                                <Width> 25.40 </Width>
                                <Height> 28.58 </Height>
                                <Length> 254.00 </Length>
                        </dimensional-constraint>
                </form-feature>
        </mating-feature>
</mating-pair>
<mating-condition>
        <assembly-joining-relation>
                <form-feature> body10427394300 </form-feature>
                <form-feature> body10427394301 </form-feature>
        </assembly-joining-relations>
        <SR> aligned </SR>
        <transformed-geometric-constraint> on_line & parallel
                </transformed-geometric-constraint>
        <dof> {lin_ edge10545740117::lin_ edge10545740119}  </dof>
        <implied-constraint>  </implied-constraint>
</mating-condition>
</MB>
<Material>
        <Part-ID>  body10427394300 </Part-ID>
        <Name> Aluminum Alloy 6063 </Name>
        <Part-ID>  body10427394301 </Part-ID>
        <Name> Aluminum Alloy 6063 </Name>
</Material>
</AF>
</ASD>
```

Appendix B.2 XML Data for the ADDM


```
<?xml version="1.0" ?>
<!-- ADDM Setup File ! -->
<ADDM>
	<info>
		<name> adf1047568720 </name>
		<description>  </description>
	</info>
	<AsDAlternative>
	 <ALTERN>
		<name> A </name>
			<JF-ID> adfjf1047568720 </JF-ID>
			<joining-method> Structural Rivet </joining-method>
			<joining-component> body10475688320 </joining-component>
				<joining-entity> face10475687720 </joining-entity>
			<groove-shape>  </groove-shape>
			<joining-constraint>
				<riveting-condition>
					<washer>  </washer>
<num-rivet> 2 </num-rivet>
				</riveting-condition>
				<fixture-location>
					<id> face10475687980 </id>
				</fixture-location>
			</joining-constraint>
		<Material>
			<Part-ID> body10475688320 </Part-ID>
			<Material-Name> Aluminum Alloy </Material-Name>
			<Part-ID> body10475688321 </Part-ID>
			<Material-Name> Aluminum Alloy </Material-Name>
			<!-- rivet 1 !-->
			<Part-ID> body10475688322 </Part-ID>
			<Material-Name> Aluminum Alloy </Material-Name>
			<!-- rivet 2 !-->
			<Part-ID> body10475688323 </Part-ID>
			<Material-Name> Aluminum Alloy </Material-Name>
		</Material>
	 </ALTERN>
	 <ALTERN>
		<name> B </name>
			<JF-ID> adfjf1047568720 </JF-ID>
			<joining-method> Gas Metal Arc Welding </joining-method>
			<joining-component> body10475688320 </joining-component>
				<joining-entity> edge10461079112 </joining-entity>
			<joining-component> body10475688321 </joining-component>
				<joining-entity> face10461079985 </joining-entity>
			<groove-shape>  </groove-shape>
			<joining-constraint>
				<welding-condition>
					<amperage> 30 </amperage>
					<voltage> 220 </voltage>
					<feedrate> 2 </feedrate>
					<weld-speed> 3 </weld-speed>
```

<weaving> No Weaving </weaving>
</welding-condition>
<fixture-location>
<id> face10475687980 </id>
</fixture-location>
</joining-constraint>
<Material>
<Part-ID> body10475688320 </Part-ID>
<Material-Name> Aluminum Alloy </Material-Name>
<Part-ID> body10475688321 </Part-ID>
<Material-Name> Aluminum Alloy </Material-Name>
</Material>
</ALTERN>
<ALTERN>
<name> C </name>
<JF-ID> adfjf1047568720 </JF-ID>
<joining-method> Structural Rivet </joining-method>
<joining-component> body10475688320 </joining-component>
<joining-entity> face10475687720 </joining-entity>
<groove-shape> </groove-shape>
<joining-constraint>
<riveting-condition>
<washer> </washer>
<num-rivet> 1 </num-rivet>
</riveting-condition>
<fixture-location>
<id> face10475687980 </id>
</fixture-location>
</joining-constraint>
<Material>
<Part-ID> body10475688320 </Part-ID>
<Material-Name> Aluminum Alloy </Material-Name>
<Part-ID> body10475688321 </Part-ID>
<Material-Name> Aluminum Alloy </Material-Name>
<!-- rivet 1 !-->
<Part-ID> body10475688322 </Part-ID>
<Material-Name> Structural Steel </Material-Name>
</Material>
</ALTERN>
</AsDAlternative>

<AHPSetup>
<WCriterion>
<Criterion>
<Name> Design </Name>
<Weight> Cost: 0.25 </Weight>
<Weight> Quality: 0.33 </Weight>
</Criterion>
<Criterion>
<Name> Cost </Name>
<Weight> Design: 4.00 </Weight>
<Weight> Quality: 1.00 </Weight>
</Criterion>
<Criterion>
<Name> Quality </Name>
<Weight> Design: 3.00 </Weight>

```
                <Weight> Cost: 1.00 </Weight>
        </Criterion>
</WCriterion>
<LPAlternative>
        <Alternative>
                <Name> A </Name>
                <Criterion>
                        <Name> Design </Name>
                        <LPriority> B: 0.11 </LPriority>
                        <LPriority> C: 0.20 </LPriority>
                </Criterion>
                <Criterion>
                        <Name> Cost </Name>
                        <LPriority> B: 9.00 </LPriority>
                        <LPriority> C: 1.00 </LPriority>
                </Criterion>
                <Criterion>
                        <Name> Quality </Name>
                        <LPriority> B: 0.50 </LPriority>
                        <LPriority> C: 9.00 </LPriority>
                </Criterion>
        </Alternative>
</LPAlternative>
        <Alternative>
                <Name> B </Name>
                <Criterion>
                        <Name> Design </Name>
                        <LPriority> A: 9.00 </LPriority>
                        <LPriority> C: 5.00 </LPriority>
                </Criterion>
                <Criterion>
                        <Name> Cost </Name>
                        <LPriority> A: 0.11 </LPriority>
                        <LPriority> C: 0.11 </LPriority>
                </Criterion>
                <Criterion>
                        <Name> Quality </Name>
                        <LPriority> A: 2.00 </LPriority>
                        <LPriority> C: 9.00 </LPriority>
                </Criterion>
        </Alternative>
<LPAlternative>
</LPAlternative>
<LPAlternative>
        <Alternative>
                <Name> C </Name>
                <Criterion>
                        <Name> Design </Name>
                        <LPriority> A: 5.00 </LPriority>
                        <LPriority> B: 0.20 </LPriority>
                </Criterion>
                <Criterion>
                        <Name> Cost </Name>
                        <LPriority> A: 1.00 </LPriority>
                        <LPriority> B: 9.00 </LPriority>
                </Criterion>
```

```
                <Criterion>
                    <Name> Quality </Name>
                    <LPriority> A: 0.11 </LPriority>
                    <LPriority> B: 0.11 </LPriority>
                </Criterion>
            </Alternative>
        </LPAlternative>
    </AHPSetup>
</ADDM>
```

APPENDIX C

VAA INPUT

Appendix C.1 An Example of ANSYS Analysis Input

```
/batch
/config,nproc,2
*get,wallstrt,active,,time,wall
*get,version,active,,rev
/nopr
/track,-1
/prep7
shpp,off
fcum,add
sfcum,all,add
/com,*********** Nodes for Part 1 ***********
nblock,3
(1i8,3e20.9e3)
       1  -5.159428876E+001  -3.788754722E+000   1.120588235E+002
       2  -5.159428876E+001  -3.788754722E+000   9.711764706E+001
       3  -5.159428876E+001  -3.788754722E+000   8.217647059E+001
       4  -5.159428876E+001  -3.788754722E+000   6.723529412E+001
       5  -5.159428876E+001  -3.788754722E+000   5.229411765E+001
       6  -5.159428876E+001  -3.788754722E+000   3.735294118E+001
       7  -5.159428876E+001  -3.788754722E+000   2.241176471E+001
       8  -5.159428876E+001  -3.788754722E+000   7.470588235E+000
       9  -5.159428876E+001  -3.788754722E+000  -7.470588235E+000
      10  -5.159428876E+001  -3.788754722E+000  -2.241176471E+001
      11  -5.159428876E+001  -3.788754722E+000  -3.735294118E+001
      12  -5.159428876E+001  -3.788754722E+000  -5.229411765E+001
      13  -5.159428876E+001  -3.788754722E+000  -6.723529412E+001
      14  -5.159428876E+001  -3.788754722E+000  -8.217647059E+001
      15  -5.159428876E+001  -3.788754722E+000  -9.711764706E+001
      16  -5.159428876E+001  -3.788754722E+000  -1.120588235E+002
      17  -3.101092863E+001  -2.437211484E+001   1.120588235E+002
      18  -3.101092863E+001  -2.437211484E+001   9.711764706E+001
      19  -3.101092863E+001  -2.437211484E+001   8.217647059E+001
      20  -3.101092863E+001  -2.437211484E+001   6.723529412E+001
      21  -3.101092863E+001  -2.437211484E+001   5.229411765E+001
      22  -3.101092863E+001  -2.437211484E+001   3.735294118E+001
      23  -3.101092863E+001  -2.437211484E+001   2.241176471E+001
      24  -3.101092863E+001  -2.437211484E+001   7.470588235E+000
      25  -3.101092863E+001  -2.437211484E+001  -7.470588235E+000
```

26	-3.101092863E+001	-2.437211484E+001	-2.241176471E+001
27	-3.101092863E+001	-2.437211484E+001	-3.735294118E+001
28	-3.101092863E+001	-2.437211484E+001	-5.229411765E+001

!!! Continued --- intermediate steps are omitted due to space limitations

2272	-5.400260870E+001	-1.110387228E+001	7.470587790E+000
2273	-5.400260870E+001	-1.408043478E+001	1.494117558E+001
2274	-5.400260870E+001	-1.110387228E+001	2.241176337E+001
2275	-5.400260870E+001	-1.408043478E+001	2.988235116E+001
2276	-5.400260870E+001	-1.110387228E+001	3.735293895E+001
2277	-5.400260870E+001	-1.408043478E+001	4.482352674E+001
2278	-5.400260870E+001	-1.110387228E+001	5.229411453E+001
2279	-5.400260870E+001	-1.408043478E+001	5.976470421E+001
2280	-5.400260870E+001	-1.110387228E+001	6.723529390E+001
2281	-5.400260870E+001	-1.408043478E+001	7.470588169E+001
2282	-5.400260870E+001	-1.110387228E+001	8.217646948E+001
2283	-5.400260870E+001	-1.408043478E+001	8.964705821E+001
2284	-5.400260870E+001	-1.110387228E+001	9.711764695E+001
2285	-5.400260870E+001	-1.408043478E+001	1.045882352E+002
2286	-5.400260870E+001	-1.110387228E+001	1.120588235E+002
2287	-5.400260870E+001	-1.408043478E+001	1.195294117E+002
2288	-5.400260870E+001	-2.003355978E+001	-1.195294122E+002
2289	-5.400260870E+001	-1.705699728E+001	-1.120588244E+002
2290	-5.400260870E+001	-2.003355978E+001	-1.045882366E+002
2291	-5.400260870E+001	-1.705699728E+001	-9.711764884E+001
2292	-5.400260870E+001	-2.003355978E+001	-8.964706105E+001
2293	-5.400260870E+001	-1.705699728E+001	-8.217647326E+001
2294	-5.400260870E+001	-2.003355978E+001	-7.470588547E+001
2295	-5.400260870E+001	-1.705699728E+001	-6.723529768E+001
2296	-5.400260870E+001	-2.003355978E+001	-5.976470989E+001
2297	-5.400260870E+001	-1.705699728E+001	-5.229412210E+001
2298	-5.400260870E+001	-2.003355978E+001	-4.482353431E+001
2299	-5.400260870E+001	-1.705699728E+001	-3.735294652E+001
2300	-5.400260870E+001	-2.003355978E+001	-2.988235873E+001
2301	-5.400260870E+001	-1.705699728E+001	-2.241177094E+001
2302	-5.400260870E+001	-2.003355978E+001	-1.494118315E+001
2303	-5.400260870E+001	-1.705699728E+001	-7.470595360E+000
2304	-5.400260870E+001	-2.003355978E+001	-3.784894830E-006
2305	-5.400260870E+001	-1.705699728E+001	7.470587790E+000
2306	-5.400260870E+001	-2.003355978E+001	1.494117558E+001
2307	-5.400260870E+001	-1.705699728E+001	2.241176337E+001
2308	-5.400260870E+001	-2.003355978E+001	2.988235116E+001
2309	-5.400260870E+001	-1.705699728E+001	3.735293895E+001
2310	-5.400260870E+001	-2.003355978E+001	4.482352674E+001
2311	-5.400260870E+001	-1.705699728E+001	5.229411453E+001
2312	-5.400260870E+001	-2.003355978E+001	5.976470421E+001
2313	-5.400260870E+001	-1.705699728E+001	6.723529390E+001
2314	-5.400260870E+001	-2.003355978E+001	7.470588169E+001
2315	-5.400260870E+001	-1.705699728E+001	8.217646948E+001
2316	-5.400260870E+001	-2.003355978E+001	8.964705821E+001
2317	-5.400260870E+001	-1.705699728E+001	9.711764695E+001
2318	-5.400260870E+001	-2.003355978E+001	1.045882352E+002
2319	-5.400260870E+001	-1.705699728E+001	1.120588235E+002
2320	-5.400260870E+001	-2.003355978E+001	1.195294117E+002
2321	-5.400260870E+001	-2.340699728E+001	-1.120588244E+002

```
   2322  -5.400260870E+001  -2.340699728E+001  -9.711764884E+001
   2323  -5.400260870E+001  -2.340699728E+001  -8.217647326E+001
   2324  -5.400260870E+001  -2.340699728E+001  -6.723529768E+001
   2325  -5.400260870E+001  -2.340699728E+001  -5.229412210E+001
   2326  -5.400260870E+001  -2.340699728E+001  -3.735294652E+001
   2327  -5.400260870E+001  -2.340699728E+001  -2.241177094E+001
   2328  -5.400260870E+001  -2.340699728E+001  -7.470595360E+000
   2329  -5.400260870E+001  -2.340699728E+001   7.470587790E+000
   2330  -5.400260870E+001  -2.340699728E+001   2.241176337E+001
   2331  -5.400260870E+001  -2.340699728E+001   3.735293895E+001
   2332  -5.400260870E+001  -2.340699728E+001   5.229411453E+001
   2333  -5.400260870E+001  -2.340699728E+001   6.723529390E+001
   2334  -5.400260870E+001  -2.340699728E+001   8.217646948E+001
   2335  -5.400260870E+001  -2.340699728E+001   9.711764695E+001
   2336  -5.400260870E+001  -2.340699728E+001   1.120588235E+002
   2337  -5.400260870E+001  -2.340699728E+001   1.270000000E+002
   2338  -5.400260870E+001  -1.705699728E+001   1.270000000E+002
   2339  -5.400260870E+001  -1.110387228E+001   1.270000000E+002
   2340  -5.400260870E+001  -4.753872283E+000   1.270000000E+002
! end of nblock command
/com,*********** Elements for Part 1 ***********
et,1,95
eblock,10
(15i8)
       1       1       1       1       0      49     256     190     379      68     257     238     425     697    1517
       0       0       0       0       0     873     905    1168    1521    1180    1181     921    1541    1520    1909
       2       1       1       1       0      49     379     190     256      50     382     193     255     905     873
       0       0       0       0       0    1517     697     906     875    1510     698     922    1903    1516    1540
       3       1       1       1       0      50     382     193     255      51     385     196     254     906     875
       0       0       0       0       0    1510     698     907     877    1503     699     925    1896    1509    1539
       4       1       1       1       0      51     385     196     254      52     388     199     253     907     877
       0       0       0       0       0    1503     699     908     879    1496     700     928    1889    1502    1538
       5       1       1       1       0      52     388     199     253      53     391     202     252     908     879
       0       0       0       0       0    1496     700     909     881    1489     701     931    1882    1495    1537
       6       1       1       1       0      53     391     202     252      54     394     205     251     909     881
       0       0       0       0       0    1489     701     910     883    1482     702     934    1875    1488    1536
       7       1       1       1       0      54     394     205     251      55     397     208     250     910     883
       0       0       0       0       0    1482     702     911     885    1475     703     937    1868    1481    1535
       8       1       1       1       0      55     397     208     250      56     400     211     249     911     885
       0       0       0       0       0    1475     703     912     887    1468     704     940    1861    1474    1534
       9       1       1       1       0      56     400     211     249      57     403     214     248     912     887
       0       0       0       0       0    1468     704     913     889    1461     705     943    1854    1467    1533
      10       1       1       1       0      57     403     214     248      58     406     217     247     913     889
       0       0       0       0       0    1461     705     914     891    1454     706     946    1847    1460    1532
      11       1       1       1       0      58     406     217     247      59     409     220     246     914     891
       0       0       0       0       0    1454     706     915     893    1447     707     949    1840    1453    1531
      12       1       1       1       0      59     409     220     246      60     412     223     245     915     893
       0       0       0       0       0    1447     707     916     895    1440     708     952    1833    1446    1530
      13       1       1       1       0      60     412     223     245      61     415     226     244     916     895
       0       0       0       0       0    1440     708     917     897    1433     709     955    1826    1439    1529
      14       1       1       1       0      61     415     226     244      62     418     229     243     917     897
       0       0       0       0       0    1433     709     918     899    1426     710     958    1819    1432    1528
      15       1       1       1       0      62     418     229     243      63     421     232     242     918     899
       0       0       0       0       0    1426     710     919     901    1419     711     961    1812    1425    1527
      16       1       1       1       0      63     421     232     242      64     424     235     241     919     901
       0       0       0       0       0    1419     711     920     903    1412     712     964    1805    1418    1526
```

```
   17     1     1     1     0    64   424   235   241    69    87    98    99   920   903
    0     0     0     0     0  1412   712  1208  1217  1248  1206   967  1798  1411  1525

!!! Continued --- intermediate steps are omitted due to space limitations

  391     1     1     1     0    48   575   581   581    70    78    79    79  1115  2092
    0     0     0     0     0   581  1164  1201  1228    79  1203  1162  2091  2205  2205
  392     1     1     1     0    33   598   596   596    67   646   597   597  1118  2336
    0     0     0     0     0   596  1119  1192  2337   597  1191  1116  2320  2221  2221
  393     1     1     1     0    33   596   598   598    34   595   599   599  1119  2336
    0     0     0     0     0   598  1118  1122  2335   599  1121  1117  2220  2318  2318
  394     1     1     1     0    34   595   599   599    35   594   600   600  1122  2335
    0     0     0     0     0   599  1121  1125  2334   600  1124  1120  2219  2316  2316
  395     1     1     1     0    35   594   600   600    36   593   601   601  1125  2334
    0     0     0     0     0   600  1124  1128  2333   601  1127  1123  2218  2314  2314
  396     1     1     1     0    36   593   601   601    37   592   602   602  1128  2333
    0     0     0     0     0   601  1127  1131  2332   602  1130  1126  2217  2312  2312
  397     1     1     1     0    37   592   602   602    38   591   603   603  1131  2332
    0     0     0     0     0   602  1130  1134  2331   603  1133  1129  2216  2310  2310
  398     1     1     1     0    38   591   603   603    39   590   604   604  1134  2331
    0     0     0     0     0   603  1133  1137  2330   604  1136  1132  2215  2308  2308
  399     1     1     1     0    39   590   604   604    40   589   605   605  1137  2330
    0     0     0     0     0   604  1136  1140  2329   605  1139  1135  2214  2306  2306
  400     1     1     1     0    40   589   605   605    41   588   606   606  1140  2329
    0     0     0     0     0   605  1139  1143  2328   606  1142  1138  2213  2304  2304
  401     1     1     1     0    41   588   606   606    42   587   607   607  1143  2328
    0     0     0     0     0   606  1142  1146  2327   607  1145  1141  2212  2302  2302
  402     1     1     1     0    42   587   607   607    43   586   608   608  1146  2327
    0     0     0     0     0   607  1145  1149  2326   608  1148  1144  2211  2300  2300
  403     1     1     1     0    43   586   608   608    44   585   609   609  1149  2326
    0     0     0     0     0   608  1148  1152  2325   609  1151  1147  2210  2298  2298
  404     1     1     1     0    44   585   609   609    45   584   610   610  1152  2325
    0     0     0     0     0   609  1151  1155  2324   610  1154  1150  2209  2296  2296
  405     1     1     1     0    45   584   610   610    46   583   611   611  1155  2324
    0     0     0     0     0   610  1154  1158  2323   611  1157  1153  2208  2294  2294
  406     1     1     1     0    46   583   611   611    47   582   612   612  1158  2323
    0     0     0     0     0   611  1157  1161  2322   612  1160  1156  2207  2292  2292
  407     1     1     1     0    47   582   612   612    48   581   613   613  1161  2322
    0     0     0     0     0   612  1160  1164  2321   613  1163  1159  2206  2290  2290
  408     1     1     1     0    48   581   613   613    70    79    75    75  1164  2321
    0     0     0     0     0   613  1163  1203  1224    75  1204  1162  2205  2288  2288
-1
mp,ex,1,71000000.
mp,nuxy,1,0.33
mp,alpx,1,1.7e-005
/com,*********** Nodes for Part 2 ***********
nblock,3
(1i8,3e20.9e3)
    2341   1.850562148E+002   -1.380434783E+000   -1.143000000E+002
    2342   1.701150383E+002   -1.380434783E+000   -1.143000000E+002
    2343   1.551738608E+002   -1.380434783E+000   -1.143000000E+002
    2344   1.402326852E+002   -1.380434783E+000   -1.143000000E+002
    2345   1.252915058E+002   -1.380434783E+000   -1.143000000E+002
    2346   1.103503303E+002   -1.380434783E+000   -1.143000000E+002
    2347   9.540915467E+001   -1.380434783E+000   -1.143000000E+002
    2348   8.046797909E+001   -1.380434783E+000   -1.143000000E+002
```

2349	6.552679594E+001	-1.380434783E+000	-1.143000000E+002
2350	5.058562036E+001	-1.380434783E+000	-1.143000000E+002
2351	3.564444478E+001	-1.380434783E+000	-1.143000000E+002
2352	2.070326920E+001	-1.380434783E+000	-1.143000000E+002
2353	5.762093624E+000	-1.380434783E+000	-1.143000000E+002
2354	-9.179081956E+000	-1.380434783E+000	-1.143000000E+002
2355	-2.412025754E+001	-1.380434783E+000	-1.143000000E+002
2356	-3.906143312E+001	-1.380434783E+000	-1.143000000E+002
2357	-3.906143312E+001	1.794565217E+000	-1.143000000E+002
2358	-2.412025754E+001	1.794565217E+000	-1.143000000E+002
2359	-9.179081956E+000	1.794565217E+000	-1.143000000E+002
2360	5.762093624E+000	1.794565217E+000	-1.143000000E+002
2361	2.070326920E+001	1.794565217E+000	-1.143000000E+002
2362	3.564444478E+001	1.794565217E+000	-1.143000000E+002
2363	5.058562036E+001	1.794565217E+000	-1.143000000E+002
2364	6.552679594E+001	1.794565217E+000	-1.143000000E+002

!!! Continued --- intermediate steps are omitted due to space limitations

2636	-5.400260870E+001	2.070652174E-001	-1.143000000E+002
2637	-5.400260870E+001	-1.380434783E+000	-1.079500000E+002
2638	-5.400260870E+001	-1.380434783E+000	-1.206500000E+002
2639	-5.400260870E+001	2.070652174E-001	-1.016000000E+002
2640	1.999973913E+002	2.070652174E-001	-1.143000000E+002
2641	1.999973913E+002	-1.380434783E+000	-1.079500000E+002
2642	1.999973913E+002	-1.380434783E+000	-1.206500000E+002
2643	1.999973913E+002	2.070652174E-001	-1.016000000E+002
2644	-3.906143312E+001	2.070652174E-001	-1.016000000E+002
2645	-2.412025754E+001	2.070652174E-001	-1.016000000E+002
2646	-9.179081956E+000	2.070652174E-001	-1.016000000E+002
2647	5.762093624E+000	2.070652174E-001	-1.016000000E+002
2648	2.070326920E+001	2.070652174E-001	-1.016000000E+002
2649	3.564444478E+001	2.070652174E-001	-1.016000000E+002
2650	5.058562036E+001	2.070652174E-001	-1.016000000E+002
2651	6.552679594E+001	2.070652174E-001	-1.016000000E+002
2652	8.046797909E+001	2.070652174E-001	-1.016000000E+002
2653	9.540915467E+001	2.070652174E-001	-1.016000000E+002
2654	1.103503303E+002	2.070652174E-001	-1.016000000E+002
2655	1.252915058E+002	2.070652174E-001	-1.016000000E+002
2656	1.402326852E+002	2.070652174E-001	-1.016000000E+002
2657	1.551738608E+002	2.070652174E-001	-1.016000000E+002
2658	1.701150383E+002	2.070652174E-001	-1.016000000E+002
2659	1.850562148E+002	2.070652174E-001	-1.016000000E+002
2660	1.925268030E+002	-1.380434783E+000	-1.016000000E+002
2661	1.775856265E+002	-1.380434783E+000	-1.016000000E+002
2662	1.626444495E+002	-1.380434783E+000	-1.016000000E+002
2663	1.477032730E+002	-1.380434783E+000	-1.016000000E+002
2664	1.327620955E+002	-1.380434783E+000	-1.016000000E+002
2665	1.178209180E+002	-1.380434783E+000	-1.016000000E+002
2666	1.028797425E+002	-1.380434783E+000	-1.016000000E+002
2667	8.793856688E+001	-1.380434783E+000	-1.016000000E+002
2668	7.299738752E+001	-1.380434783E+000	-1.016000000E+002
2669	5.805620815E+001	-1.380434783E+000	-1.016000000E+002
2670	4.311503257E+001	-1.380434783E+000	-1.016000000E+002
2671	2.817385699E+001	-1.380434783E+000	-1.016000000E+002
2672	1.323268141E+001	-1.380434783E+000	-1.016000000E+002

```
   2673  -1.708494166E+000  -1.380434783E+000  -1.016000000E+002
   2674  -1.664966975E+001  -1.380434783E+000  -1.016000000E+002
   2675  -3.159084533E+001  -1.380434783E+000  -1.016000000E+002
   2676  -4.653202091E+001  -1.380434783E+000  -1.016000000E+002
! end of nblock command
/com,*********** Elements for Part 2 ***********
et,2,95
eblock,10
(15i8)
     409       2       2       2       0    2341    2431    2427    2411    2372    2390    2410    2409    2465    2642
       0       0       0       0       0    2617    2466    2515    2581    2600    2514    2449    2640    2634    2616
     410       2       2       2       0    2341    2433    2432    2431    2372    2388    2389    2390    2468    2660
       0       0       0       0       0    2641    2465    2516    2579    2580    2515    2449    2659    2643    2640
     411       2       2       2       0    2341    2411    2412    2342    2372    2409    2408    2371    2466    2618
       0       0       0       0       0    2469    2467    2514    2599    2517    2518    2449    2616    2615    2450
     412       2       2       2       0    2341    2342    2434    2433    2372    2371    2387    2388    2467    2471
       0       0       0       0       0    2661    2468    2518    2519    2578    2516    2449    2450    2658    2659
     413       2       2       2       0    2342    2412    2413    2343    2371    2408    2407    2370    2469    2619
       0       0       0       0       0    2472    2470    2517    2598    2520    2521    2450    2615    2614    2451
     414       2       2       2       0    2342    2343    2435    2434    2371    2370    2386    2387    2470    2474
       0       0       0       0       0    2662    2471    2521    2522    2577    2519    2450    2451    2657    2658
     415       2       2       2       0    2343    2413    2414    2344    2370    2407    2406    2369    2472    2620
       0       0       0       0       0    2475    2473    2520    2597    2523    2524    2451    2614    2613    2452
     416       2       2       2       0    2343    2344    2436    2435    2370    2369    2385    2386    2473    2477
       0       0       0       0       0    2663    2474    2524    2525    2576    2522    2451    2452    2656    2657
     417       2       2       2       0    2344    2414    2415    2345    2369    2406    2405    2368    2475    2621
       0       0       0       0       0    2478    2476    2523    2596    2526    2527    2452    2613    2612    2453
     418       2       2       2       0    2344    2345    2437    2436    2369    2368    2384    2385    2476    2480
       0       0       0       0       0    2664    2477    2527    2528    2575    2525    2452    2453    2655    2656
     419       2       2       2       0    2345    2415    2416    2346    2368    2405    2404    2367    2478    2622
       0       0       0       0       0    2481    2479    2526    2595    2529    2530    2453    2612    2611    2454
     420       2       2       2       0    2345    2346    2438    2437    2368    2367    2383    2384    2479    2483
       0       0       0       0       0    2665    2480    2530    2531    2574    2528    2453    2454    2654    2655
     421       2       2       2       0    2346    2416    2417    2347    2367    2404    2403    2366    2481    2623
       0       0       0       0       0    2484    2482    2529    2594    2532    2533    2454    2611    2610    2455
     422       2       2       2       0    2346    2347    2439    2438    2367    2366    2382    2383    2482    2486
       0       0       0       0       0    2666    2483    2533    2534    2573    2531    2454    2455    2653    2654
     423       2       2       2       0    2347    2417    2418    2348    2366    2403    2402    2365    2484    2624
       0       0       0       0       0    2487    2485    2532    2593    2535    2536    2455    2610    2609    2456
     424       2       2       2       0    2347    2348    2440    2439    2366    2365    2381    2382    2485    2489
       0       0       0       0       0    2667    2486    2536    2537    2572    2534    2455    2456    2652    2653
     425       2       2       2       0    2348    2418    2419    2349    2365    2402    2401    2364    2487    2625
       0       0       0       0       0    2490    2488    2535    2592    2538    2539    2456    2609    2608    2457
     426       2       2       2       0    2348    2349    2441    2440    2365    2364    2380    2381    2488    2492
       0       0       0       0       0    2668    2489    2539    2540    2571    2537    2456    2457    2651    2652
     427       2       2       2       0    2349    2419    2420    2350    2364    2401    2400    2363    2490    2626
       0       0       0       0       0    2493    2491    2538    2591    2541    2542    2457    2608    2607    2458
     428       2       2       2       0    2349    2350    2442    2441    2364    2363    2379    2380    2491    2495
       0       0       0       0       0    2669    2492    2542    2543    2570    2540    2457    2458    2650    2651
     429       2       2       2       0    2350    2420    2421    2351    2363    2400    2399    2362    2493    2627
       0       0       0       0       0    2496    2494    2541    2590    2544    2545    2458    2607    2606    2459
     430       2       2       2       0    2350    2351    2443    2442    2363    2362    2378    2379    2494    2498
       0       0       0       0       0    2670    2495    2545    2546    2569    2543    2458    2459    2649    2650
     431       2       2       2       0    2351    2421    2422    2352    2362    2399    2398    2361    2496    2628
       0       0       0       0       0    2499    2497    2544    2589    2547    2548    2459    2606    2605    2460
     432       2       2       2       0    2351    2352    2444    2443    2362    2361    2377    2378    2497    2501
```

```
       0       0       0       0       0    2671    2498    2548    2549    2568    2546    2459    2460    2648    2649
     433       2       2       2       0    2352    2422    2423    2353    2361    2398    2397    2360    2499    2629
       0       0       0       0       0    2502    2500    2547    2588    2550    2551    2460    2605    2604    2461
     434       2       2       2       0    2352    2353    2445    2444    2361    2360    2376    2377    2500    2504
       0       0       0       0       0    2672    2501    2551    2552    2567    2549    2460    2461    2647    2648
     435       2       2       2       0    2353    2423    2424    2354    2360    2397    2396    2359    2502    2630
       0       0       0       0       0    2505    2503    2550    2587    2553    2554    2461    2604    2603    2462
     436       2       2       2       0    2353    2354    2446    2445    2360    2359    2375    2376    2503    2507
       0       0       0       0       0    2673    2504    2554    2555    2566    2552    2461    2462    2646    2647
     437       2       2       2       0    2354    2424    2425    2355    2359    2396    2395    2358    2505    2631
       0       0       0       0       0    2508    2506    2553    2586    2556    2557    2462    2603    2602    2463
     438       2       2       2       0    2354    2355    2447    2446    2359    2358    2374    2375    2506    2510
       0       0       0       0       0    2674    2507    2557    2558    2565    2555    2462    2463    2645    2646
     439       2       2       2       0    2355    2425    2426    2356    2358    2395    2394    2357    2508    2632
       0       0       0       0       0    2511    2509    2556    2585    2559    2560    2463    2602    2601    2464
     440       2       2       2       0    2355    2356    2448    2447    2358    2357    2373    2374    2509    2513
       0       0       0       0       0    2675    2510    2560    2561    2564    2558    2463    2464    2644    2645
     441       2       2       2       0    2356    2426    2428    2429    2357    2394    2393    2392    2511    2633
       0       0       0       0       0    2638    2512    2559    2584    2583    2562    2464    2601    2635    2636
     442       2       2       2       0    2356    2429    2430    2448    2357    2392    2391    2373    2512    2637
       0       0       0       0       0    2676    2513    2562    2582    2563    2561    2464    2636    2639    2644
-1
mp,ex,2,71000000.
mp,nuxy,2,0.33
mp,alpx,2,1.7e-005
/com,*********** Create Contact Pair 1 ***********
*set,tid,3
*set,cid,4
et,tid,170
et,cid,174
r,tid
r,cid
eblock,10
(15i8)
     443       3       3       3       0     514     465     462     513    2054    2060    2061    2070
     444       3       3       3       0     513     462      81      80    2061    2064    1229    2069
     445       3       3       3       0     464     445      88      83    2055    1933    1232    2062
     446       3       3       3       0     467     446     445     464    2048    1934    2055    2056
     447       3       3       3       0     466     467     464     463    2050    2056    2057    2058
     448       3       3       3       0      83      82     463     464    1231    2063    2057    2062
     449       3       3       3       0     465     466     463     462    2052    2058    2059    2060
     450       3       3       3       0      82      81     462     463    1230    2064    2059    2063
-1
eblock,10
(15i8)
     451       4       4       4       0     514     465     462     513    2054    2060    2061    2070
     452       4       4       4       0     513     462      81      80    2061    2064    1229    2069
     453       4       4       4       0     464     445      88      83    2055    1933    1232    2062
     454       4       4       4       0     467     446     445     464    2048    1934    2055    2056
     455       4       4       4       0     466     467     464     463    2050    2056    2057    2058
     456       4       4       4       0      83      82     463     464    1231    2063    2057    2062
     457       4       4       4       0     465     466     463     462    2052    2058    2059    2060
     458       4       4       4       0      82      81     462     463    1230    2064    2059    2063
-1
eblock,10
(15i8)
```

```
     459       3       4       3       0    2355    2356    2426    2425    2509    2511    2632    2508
     460       3       4       3       0    2355    2447    2448    2356    2510    2675    2513    2509
     461       3       4       3       0    2356    2429    2428    2426    2512    2638    2633    2511
     462       3       4       3       0    2356    2448    2430    2429    2513    2676    2637    2512
-1
eblock,10
(15i8)
     463       4       3       4       0    2355    2356    2426    2425    2509    2511    2632    2508
     464       4       3       4       0    2355    2447    2448    2356    2510    2675    2513    2509
     465       4       3       4       0    2356    2429    2428    2426    2512    2638    2633    2511
     466       4       3       4       0    2356    2448    2430    2429    2513    2676    2637    2512
-1
keyo,cid,2,1
keyo,cid,8,1
keyo,cid,9,1
keyo,cid,12,5
rmod,tid,3,10.
rmod,tid,6,0.2
rmod,tid,12,1.e-002
rmod,tid,5,0.
rmod,cid,3,10.
rmod,cid,6,0.2
rmod,cid,12,1.e-002
rmod,cid,5,0.
nsel,all
esel,all
/com,*********** Displacements ***********
d,65,ux,0.
d,66,ux,0.
d,67,ux,0.
d,68,ux,0.
d,153,ux,0.
d,154,ux,0.
d,155,ux,0.
d,172,ux,0.
d,189,ux,0.
d,238,ux,0.
d,239,ux,0.
d,240,ux,0.
d,257,ux,0.
d,306,ux,0.
d,307,ux,0.
d,308,ux,0.
d,325,ux,0.
d,374,ux,0.
d,375,ux,0.
d,376,ux,0.
d,425,ux,0.
d,426,ux,0.
d,427,ux,0.
d,444,ux,0.

   !!! Continued --- intermediate steps are omitted due to space limitations

d,1794,uz,0.
d,1795,uz,0.
```

```
d,1796,uz,0.
d,1912,uz,0.
d,1913,uz,0.
d,1914,uz,0.
d,1915,uz,0.
d,2065,uz,0.
d,2066,uz,0.
d,2067,uz,0.
d,2068,uz,0.
d,2201,uz,0.
d,2202,uz,0.
d,2203,uz,0.
d,2204,uz,0.
d,2337,uz,0.
d,2338,uz,0.
d,2339,uz,0.
d,2340,uz,0.
/com,*********** Send Solved Temperatures ***********
bf,1,temp,298.741912842
bf,2,temp,300.507507324
bf,3,temp,303.380401611
bf,4,temp,307.372131348
bf,5,temp,312.498779297
bf,6,temp,318.781005859
bf,7,temp,326.24432373
bf,8,temp,334.919281006
bf,9,temp,344.842834473
bf,10,temp,356.060455322
bf,11,temp,368.635253906
bf,12,temp,382.667694092
bf,13,temp,398.377960205
bf,14,temp,416.277130127
bf,15,temp,438.071380615
bf,16,temp,459.557769775
bf,17,temp,298.741912842
bf,18,temp,300.507507324
bf,19,temp,303.380401611

   !!! Continued --- intermediate steps are omitted due to space limitations

bf,2291,temp,431.696746826
bf,2292,temp,422.347106934
bf,2293,temp,414.360565186
bf,2294,temp,405.653045654
bf,2295,temp,397.743286133
bf,2296,temp,389.842102051
bf,2297,temp,382.454956055
bf,2298,temp,375.305541992
bf,2299,temp,368.559051514
bf,2300,temp,362.116882324
bf,2301,temp,356.030731201
bf,2302,temp,350.263214111
bf,2303,temp,344.828552246
bf,2304,temp,339.710266113
bf,2305,temp,334.910369873
bf,2306,temp,330.419921875
```

```
bf,2307,temp,326.23727417
bf,2308,temp,322.35635376
bf,2309,temp,318.774719238
bf,2310,temp,315.487640381
bf,2311,temp,312.492797852
bf,2312,temp,309.786346436
bf,2313,temp,307.366333008
bf,2314,temp,305.22958374
bf,2315,temp,303.374694824
bf,2316,temp,301.799072266
bf,2317,temp,300.501861572
bf,2318,temp,299.481048584
bf,2319,temp,298.736297607
bf,2320,temp,298.266204834
bf,2321,temp,442.822601318
bf,2322,temp,429.558166504
bf,2323,temp,413.533630371
bf,2324,temp,397.455963135
bf,2325,temp,382.356231689
bf,2326,temp,368.524719238
bf,2327,temp,356.017974854
bf,2328,temp,344.8230896
bf,2329,temp,334.90737915
bf,2330,temp,326.235137939
bf,2331,temp,318.772918701
bf,2332,temp,312.491119385
bf,2333,temp,307.364715576
bf,2334,temp,303.37310791
bf,2335,temp,300.500305176
bf,2336,temp,298.734741211
bf,2337,temp,298.069366455
bf,2338,temp,298.070922852
bf,2339,temp,298.070922852
bf,2340,temp,298.069366455
/com,*********** Displacements ***********
d,2393,ux,0.
d,2394,ux,0.
d,2395,ux,0.
d,2396,ux,0.
d,2397,ux,0.
d,2398,ux,0.
d,2399,ux,0.
d,2400,ux,0.
d,2401,ux,0.
d,2402,ux,0.
d,2403,ux,0.
d,2404,ux,0.
d,2405,ux,0.
d,2406,ux,0.
d,2407,ux,0.
d,2408,ux,0.
d,2409,ux,0.
d,2410,ux,0.
d,2411,ux,0.
d,2412,ux,0.
d,2413,ux,0.
```

```
d,2414,ux,0.
d,2415,ux,0.
d,2416,ux,0.
d,2417,ux,0.
d,2418,ux,0.
d,2419,ux,0.
d,2420,ux,0.
d,2421,ux,0.
d,2422,ux,0.
d,2423,ux,0.
d,2424,ux,0.
```

!!! Continued --- intermediate steps are omitted due to space limitations

```
d,2610,uz,0.
d,2611,uz,0.
d,2612,uz,0.
d,2613,uz,0.
d,2614,uz,0.
d,2615,uz,0.
d,2616,uz,0.
d,2617,uz,0.
d,2618,uz,0.
d,2619,uz,0.
d,2620,uz,0.
d,2621,uz,0.
d,2622,uz,0.
d,2623,uz,0.
d,2624,uz,0.
d,2625,uz,0.
d,2626,uz,0.
d,2627,uz,0.
d,2628,uz,0.
d,2629,uz,0.
d,2630,uz,0.
d,2631,uz,0.
d,2632,uz,0.
d,2633,uz,0.
d,2634,uz,0.
d,2635,uz,0.
/com,*********** Send Solved Temperatures ***********
bf,2341,temp,302.565734863
bf,2342,temp,304.564880371
bf,2343,temp,307.828125
bf,2344,temp,312.370178223
bf,2345,temp,318.211608887
bf,2346,temp,325.378814697
bf,2347,temp,333.90423584
bf,2348,temp,343.826385498
bf,2349,temp,355.190124512
bf,2350,temp,368.04699707
bf,2351,temp,382.453918457
bf,2352,temp,398.480194092
bf,2353,temp,416.159179688
bf,2354,temp,435.515808105
bf,2355,temp,452.647186279
```

```
bf,2356,temp,464.658325195
bf,2357,temp,464.728637695
bf,2358,temp,454.159484863
bf,2359,temp,435.629333496
bf,2360,temp,416.194335938
bf,2361,temp,398.479980469

   !!! Continued --- intermediate steps are omitted due to space limitations

bf,2660,temp,301.986907959
bf,2661,temp,303.35748291
bf,2662,temp,305.986724854
bf,2663,temp,309.886444092
bf,2664,temp,315.074310303
bf,2665,temp,321.573791504
bf,2666,temp,329.414245605
bf,2667,temp,338.631134033
bf,2668,temp,349.266052246
bf,2669,temp,361.367248535
bf,2670,temp,374.986297607
bf,2671,temp,390.183654785
bf,2672,temp,406.942382812
bf,2673,temp,425.138122559
bf,2674,temp,442.465759277
bf,2675,temp,448.390228271
bf,2676,temp,447.642791748
tref,22.
tunif,22.
/com,*********** Performing WSORT ***********
wsort,all
/com,*********** Done With WSORT ***********
fini
*get,numnode,node,0,count
*get,numelem,elem,0,count
/go
/com,--- Number of total nodes = %numnode%
/com,--- Number of contact elements = 24
/com,--- Number of spring elements = 0
/com,--- Number of solid elements = 442
/com,--- Number of total elements = %numelem%
/com,--- Data in consistent NMM units. (See Unit Assistant for details.)
/title,Data in consistent DesignSpace NMM units
*get,wallbsol,active,,time,wall
/config,noeldb,2       ! don't write rst file
/solu
/com, Avg ratio=    1, totalParts=2, thickParts=0, thickPcent=  0
eqsl,sparse
solc,off
neqit,1
resc,,none
outres,all,none
outres,nsol,last
outres,rsol,last
outres,strs,last
outres,epel,last
outres,epth,last
```

```
solve,,,,,nocheck
fini
*get,wallasol,active,,time,wall
/post1
ernorm,on
esel,u,ename,,152,154,1
esel,u,ename,,14
esel,u,ename,,170,174,1
xmlo,dofs,epel,epth,s,serr,rfor,parm
/xml,file,xml,,,,,,,diag
fini
*get,walldone,active,,time,wall
preptime=(wallbsol-wallstrt)*3600
solvtime=(wallasol-wallbsol)*3600
posttime=(walldone-wallasol)*3600
totaltim=(walldone-wallstrt)*3600
/exit,nosa
```

APPENDIX D

SM Language

Appendix D.1 Executable Code Represented in SM Language for the Welding Cost Model

&WPCOST WELDING PROCESSING COST SECTOR

&WDATA WELD DATA

WELDi /pe/ There is a list of WELD.

WPCOST (WELDi) /a/ {WELD} : Real+ Every WELD has a nonnegative COST measured in USD.

TOT_WCOST (WELDi) /f/ 1 ; (@SUMi (WPCOSTi) There is a TOTAL WELDING COST associated with all WELD.

&EWDATA ELECTRODE_WIRE DATA

ELECTRODE_WIREj /pe/ There is a list of ELECTRODE/WIRE.

EWCOST (ELECTRODE_WIREj) /a/ {ELECTRODE_WIREj} : Real+ Every ELECTRODE AND AIRE has a nonnegative ELECTRODE AND WIRE COST measured in USD.

&SDATA SAW FLUX DATA

SAWFLUXk /pe/ There is a list of SAW FLUX.

SAWCOSTk (SAWFLUXk) /a/ {SAWFLUX} : Real+ Every SAW FLUX has a nonnegative COST measured in USD.

&GDATA GAS DATA

GASl /pe/ There is a list of GAS.

GASCOSTl (GASl) /a/ {GAS} : Real+ Every GAS has a nonnegative COST measured in USD.

&MDATA MISC DATA

MISCm /pe/ There is a list of MISC.

MISCCOSTm (MISCm) /a/ {MISC} : Real+ Every MISC has a nonnegative COST measured in USD.

&CDATA CONSUMABLES DATA

&TOTCCOST

TOTCCOST (EWCOSTj<t-4:t-1>, SAWCOSTk<t-4:t-1>, GASCOSTl<t-4:t-1>, MISCCOSTm<t-4:t-1>) /f/ 1 ; @SUMj (EWCOSTj) + @SUMk (SAWCOSTk) + @SUMl(GFASCOSTl) + @SUM(MISCCOSTm) There is a TOTAL COST CONSUMABLES associated with all CONSUMABLES.

&TDATA TOTAL WELDING PROCESSING DATA

&TOTWPCOST

TOTWPCOST /f/ 1 ; @SUM (TOT_WCOST, TOTCCOST) There is a TOTAL WELDING PROCESSING COST.

APPENDIX E

JOINING AND MATERIAL KNOWLEDGE

Appendix E.1 Comparison of Various Joining Methods

(Adopted from Kalpakjian 1995)

Method	Characteristics							
	Strength	Design Variability	Small Parts	Large Parts	Tolerances	Reliability	Ease of Maintenance	Visual Inspection
Arc welding	1	2	3	1	3	1	2	2
Resistance welding	1	2	1	1	3	3	3	3
Brazing	1	1	1	1	3	1	3	2
Bolts and nuts	1	2	3	1	2	1	1	1
Riveting	1	2	3	1	1	1	3	1
Fasteners	2	3	3	1	2	2	2	1
Seaming, crimping	2	2	1	3	3	1	3	1
Adhesive bonding	3	1	1	2	3	2	3	3

Note: 1, very good; 2, good; 3, poor

Appendix E.2 Weldable Materials

(Adopted from Kalpakjian 1995)

Material	Thickness	Welding process				
		SMAW	SAW	GMAW	FCAW	GTAW
Carbon steel	S	X	X	X		X
	I	X	X	X	X	X
	M	X	X	X	X	
	T	X	X	X	X	
Low-alloy steel	S	X	X	X		X
	I	X	X	X	X	X
	M	X	X	X	X	
	T	X	X	X	X	
Stainless steel	S	X	X	X		X
	I	X	X	X	X	X
	M	X	X	X	X	
	T	X	X	X	X	
Cast iron	I	X				
	M	X	X	X	X	
	T	X	X	X	X	
Nickel and alloys	S	X		X		X
	I	X	X	X		X
	M	X	X	X		
	T	X		X		
Aluminum and alloys	S	X		X		X
	I	X		X		X
	M	X		X		X
	T	X		X		
Titanium and alloys	S			X		X
	I			X		X
	M			X		X
	T			X		
Copper and alloys	S			X		X
	I			X		
	M			X		
	T			X		
Magnesium and alloys	S			X		X
	I			X		X
	M			X		
	T			X		
Refractory alloys	S			X		X
	I			X		
	M					
	T					

Note: SMAW – Shielded Metal-Arc Welding; SAW – Submerged Arc Welding; GMAW – Gas Metal-Arc Welding; FCAW – Flux-Cored Arc Welding; GTAW – Gas Tungsten-Arc Welding; S – Sheet: up to 3 mm; I – Intermediate: 3 to 6 mm; M – Medium: 6 to 19 mm; T – Thick: 19 mm and up.

Appendix E.3 Allowable Stress in Fastener and Joint Plate Materials

(Adopted from Messler 1993)

Material/Condition			Allowable Stress MPa (kpsi)		
Fastener	**Type**	**Condition**	**Tension**	**Shear**	**Bearing**
ASTM SA31 Rivets		SA 515 plate	_	62 (9.0)	124 (18.0)
ASTM A502-1 Rivets		A36 plate	_	93 (13.0)	276 (40.1)
ASTM A325 Bolts	Bearing-type	Threads in shear plane	_	145 (21.0)	a
		No threads in shear plane	_	207 (30.0)	a
	Friction-type	Clean mill scale	_	52 (17.5)	a
		Blasted clean	_	190 (27.5)	a
		Blasted + Zn paint	_	203 (29.5)	a
ASTM A490 Bolts	Bearing-type	Threads in shear plane	_	193 (28.0)	a
		No threads in shear plane	_	276 (40.0)	a
	Friction-type	Clean mill scale	_	152 (22.0)	a
		Blasted clean	_	238 (34.5)	a
		Blasted + Zn paint	_	255 (37.0)	a

Note: a, 1.5 S_u, here S_u is ultimate stress.

APPENDIX F

VAA SCENARIOS AND RESULTS

Appendix F.1 Welding for the Base Frame Sub-Assembly

1. Model

- The bounding box for all positioned parts in the model measures 304.8 by 330.2 by 355.6 mm along the global x, y and z axes, respectively.
- The model weighs a total of 0.69 kg.

Table F.1-1 Parts					
Name	**Material**	**Bounding Box (mm)**	**Mass (kg)**	**Nodes**	**Elements**
"Part 1"	*"Aluminum Alloy"*	254.0, 28.58, 25.4	0.18	1160	560
"Part 2"	*"Aluminum Alloy"*	76.2, 254.0, 25.4	0.22	1648	224
"Part 3"	*"Aluminum Alloy"*	25.4, 76.2, 355.6	0.29	1984	272

1.1. Contact

Table F.1-2 Contact Conditions		
Name	**Behavior**	**Associated Parts**
"Contact Region"	Bonded	*"Part 3"* and *"Part 1"*
"Contact Region 2"	Bonded	*"Part 3"* and *"Part 2"*

1.2. Mesh

- *Mesh"*, associated with *"Model"*, has an overall relevance of 0.
- *"Mesh"* contains 4792 nodes and 1056 elements.

No mesh controls specified.

2. Environment
"Environment" contains all loading conditions defined for *"Model"* in this scenario.
The following tables list local loads and supports applied to specific geometry.

2.1. Convection and Thermal Loading

Table F.1-3 Convection Loads				
Name	**Type**	**Ambient Temperature**	**Film Coefficient**	**Associated Parts**
"Convection"	Temperature-Dependent	27.0 °C	*"Stagnant Air - Simplified Case"*	*"Part 3"*, *"Part 2"* and *"Part 1"*

Table F.1-4 Thermal Loads				
Name	**Description**	**Value**	**Reaction**	**Associated Parts**
"Given Temperature"	Edge Temperature	800.0 °C	346.95 W	*"Part 1"*

2.2. Structural Supports

Table F.1-5 Structural Supports				
Name	**Type**	**Reaction Force**	**Reaction Vector**	**Associated Parts**
"Fixed Support"	Fixed Surface	6,996.48 N	[-1,550.5 N x, 4,290.37 N y, 5,304.66 N z]	*"Part 3"*
"Fixed Support 2"	Fixed Surface	5,756.95 N	[3,482.2 N x, 1,733.48 N y, -4,244.03 N z]	*"Part 1"*
"Fixed Support 3"	Fixed Surface	6,414.29 N	[-1,931.7 N x, -6,023.85 N y, -1,060.63 N z]	*"Part 2"*

3. Solution

"Solution" contains the calculated response for *"Model"* given loading conditions defined in *"Environment"*.
It was selected that the program would choose the solver used in this solution.
Thermal expansion calculations use a constant reference temperature of 22.0 °C for all parts in *"Model"*. Theoretically, at a uniform temperature of 22.0 °C no strain results from thermal expansion or contraction.

3.1. Structural Results

Table F.1-6 Values				
Name	**Scope**	**Minimum**	**Maximum**	**Alert Criteria**
"Equivalent Stress"	All Parts In *"Model"*	2.39 MPa	972.54 MPa	None
"Equivalent Strain"	All Parts In *"Model"*	4.48×10^{-5} mm/mm	1.82×10^{-2} mm/mm	None
"Total Deformation"	All Parts In *"Model"*	0.0 mm	3.95 mm	None

3.2. Thermal Results

Table F.1-7 Values				
Name	**Scope**	**Minimum**	**Maximum**	**Alert Criteria**
"Total Heat Flux"	All Parts In *"Model"*	1.37×10^{-3} W/mm²	1.49 W/mm²	None
"Temperature"	All Parts In *"Model"*	281.18 °C	800.0 °C	None

5. Definition of "Aluminum Alloy"

Table F.1-8 *"Aluminum Alloy"* Properties			
Name	**Type**	**Value**	**Temperature**
Modulus of Elasticity	Temperature-Independent	71,000.0 MPa	
Poisson's Ratio	Temperature-Independent	0.33	
Mass Density	Temperature-Independent	2.77×10^{-6} kg/mm³	
Coefficient of Thermal Expansion	Temperature-Independent	1.7×10^{-5} 1/°C	
Thermal Conductivity	Temperature-Dependent	0.11 W/mm·°C	-100.0 °C
Thermal Conductivity	Temperature-Dependent	0.14 W/mm·°C	0.0 °C
Thermal Conductivity	Temperature-Dependent	0.17 W/mm·°C	100.0 °C
Thermal Conductivity	Temperature-Dependent	0.18 W/mm·°C	200.0 °C

Table F.1-9 *"Aluminum Alloy"* Stress Limits		
Name	**Type**	**Value**
Tensile Yield Strength	Temperature-Independent	280.0 MPa
Tensile Ultimate Strength	Temperature-Independent	310.0 MPa
Compressive Yield Strength	Temperature-Independent	280.0 MPa
Compressive Ultimate Strength	Temperature-Independent	0.0 MPa

Fatigue properties come from MIL-HDBK-5H, page 3-277."
"Aluminum Alloy" contains nonlinear data for thermal conductivity. Thermal results for parts using this material usually require several iterations to converge.

Table F.1-10 Thermal Conductivity vs. Temperature

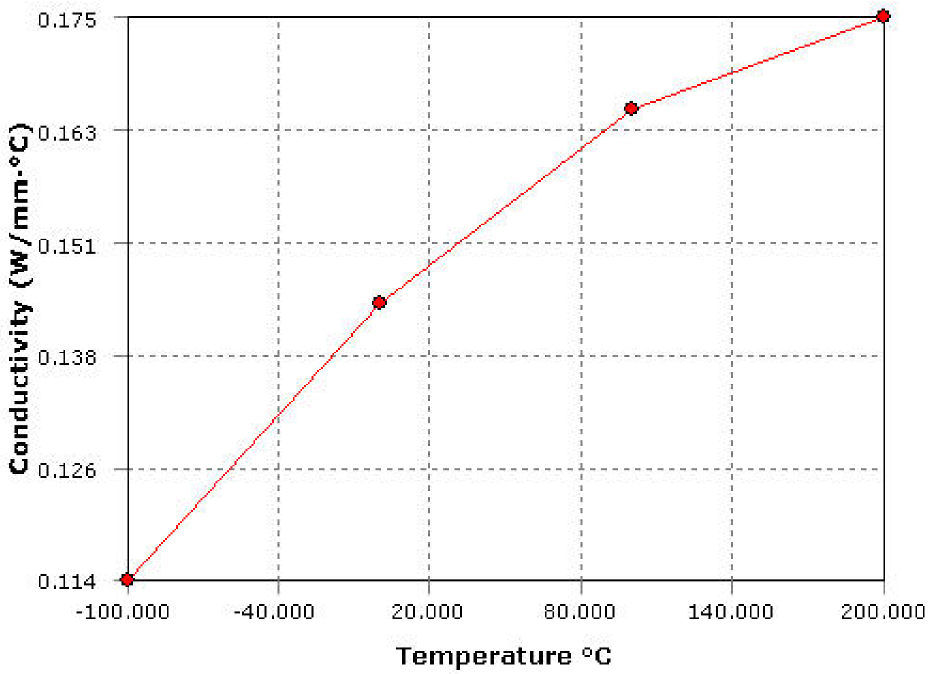
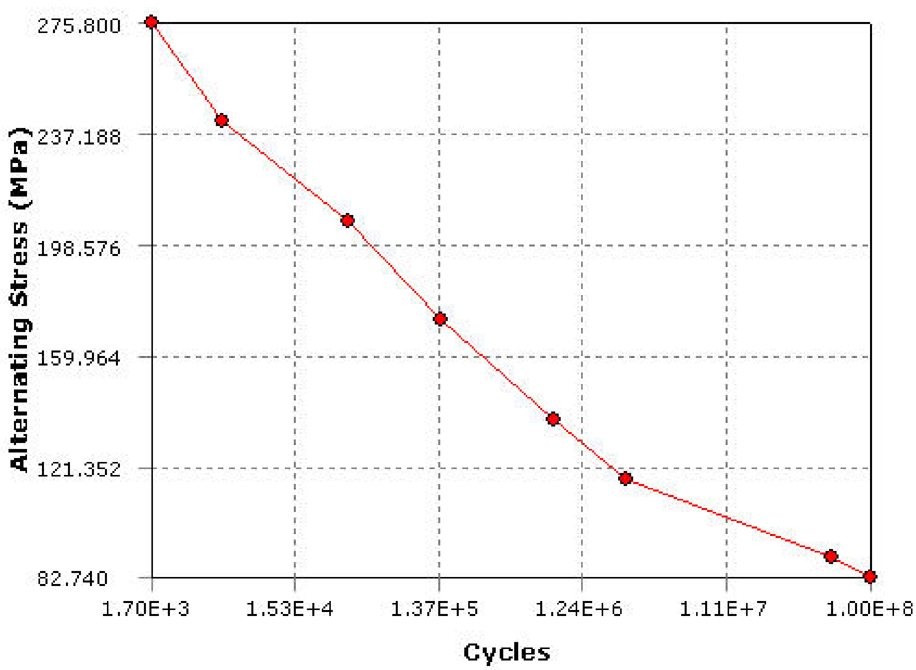


Table F.1-11 Alternating Stress vs. Cycles



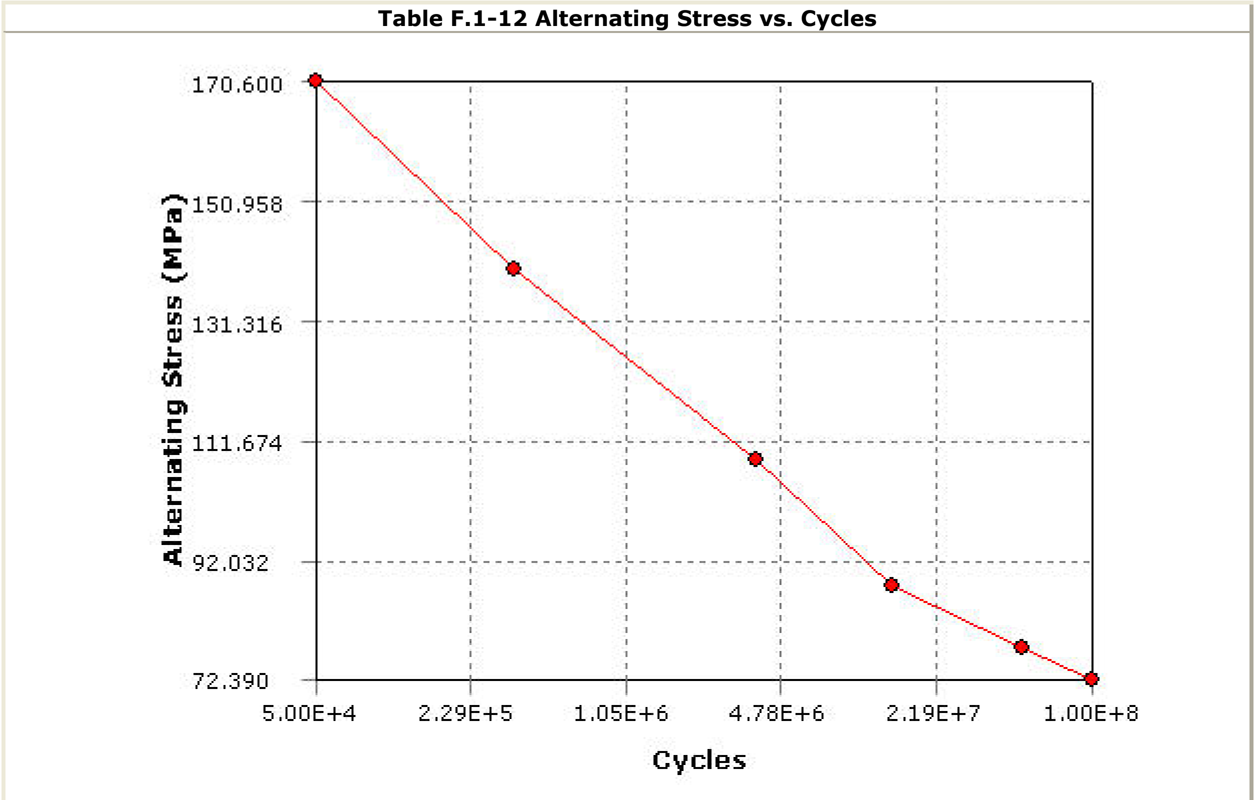


Table F.1-12 Alternating Stress vs. Cycles
170.600
150.958
131.316
111.674
92.032
72.390
Alternating Stress (MPa)
5.00E+4
2.29E+5
1.05E+6
4.78E+6
2.19E+7
1.00E+8
Cycles

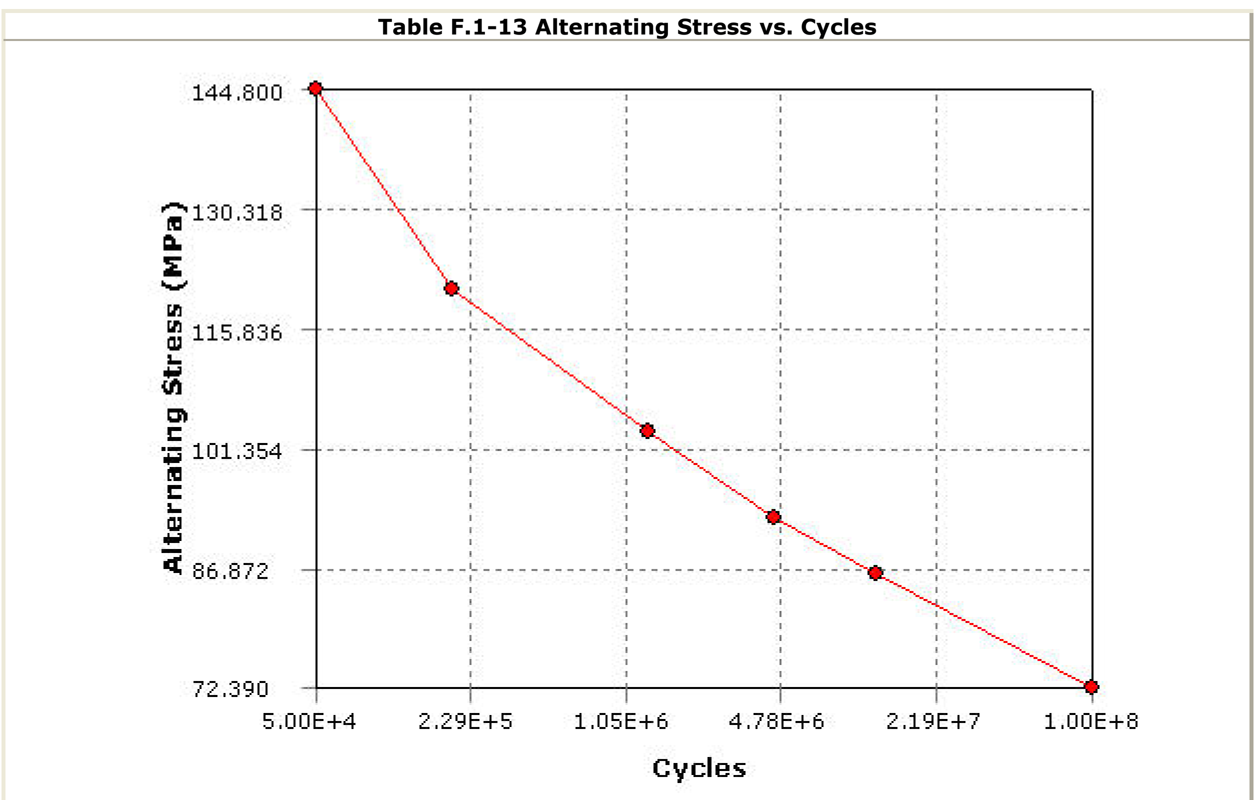
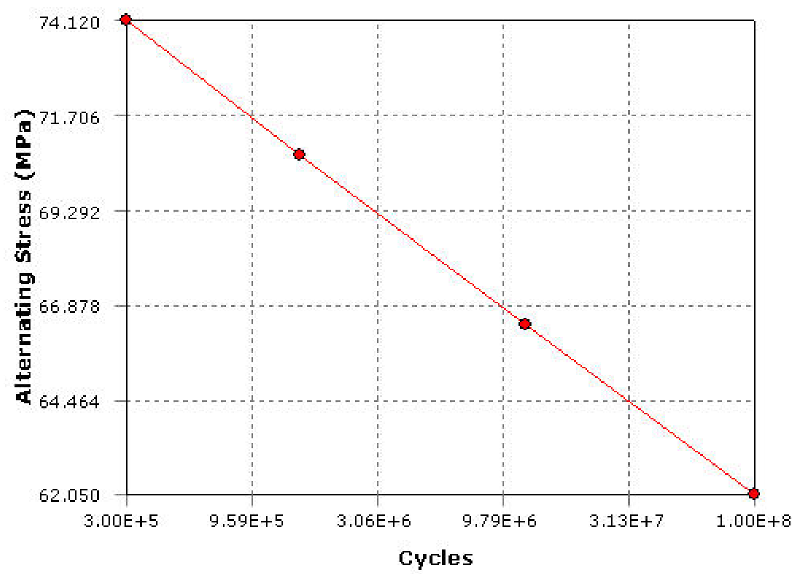


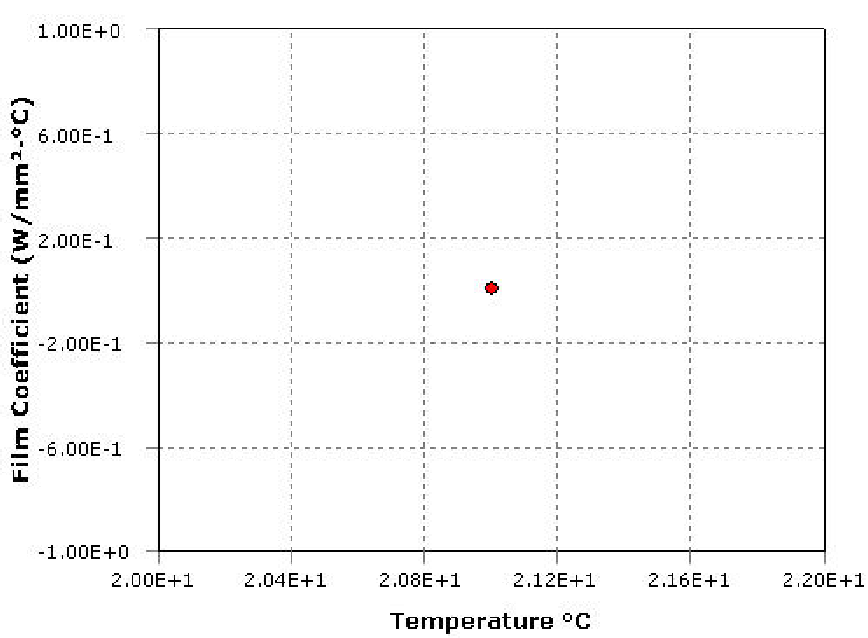
Table F.1-13 Alternating Stress vs. Cycles
144.800
130.318
115.836
101.354
86.872
72.390
Alternating Stress (MPa)
5.00E+4
2.29E+5
1.05E+6
4.78E+6
2.19E+7
1.00E+8
Cycles

Table F.1-14 Alternating Stress vs. Cycles



A4. Definition of "Stagnant Air - Simplified Case"
Temperature-independent film coefficient: 5.0×10^{-6} W/mm²·°C
Description: *"Stagnant Air Approximations, Any Geometric Configuation, Laminar or Turbulent. Using h = 5 W/m**2 - C."*
Convection data file: *"C:\Program Files\Common Files\Ansys Inc\Language\en-us\Engineering Data\Convections\Stagnant_Air_Simplified_Case.xml"*

Table F.1-15 Film Coefficient vs. Temperature



Appendix F.2 AsD Alternavie A

1. Model

- The bounding box for all positioned parts in the model measures 304.8 by 330.2 by 355.6 mm along the global x, y and z axes, respectively.
- The model weighs a total of 0.69 kg.

Table F.2-1 Parts

Name	Material	Bounding Box (mm)	Mass (kg)	Nodes	Elements
"Part 1"	*"Aluminum Alloy"*	254.0, 28.58, 25.4	0.18	1160	560
"Part 2"	*"Aluminum Alloy"*	76.2, 254.0, 25.4	0.22	1648	224
"Part 3"	*"Aluminum Alloy"*	25.4, 76.2, 355.6	0.29	1984	272

1.1 Contact

Table F.2-2 Contact Conditions

Name	Behavior	Associated Parts
"Contact Region"	Bonded	*"Part 3"* and *"Part 1"*
"Contact Region 2"	Bonded	*"Part 3"* and *"Part 2"*

1.2 Mesh

- *"Mesh"*, associated with *"Model"*, has an overall relevance of 0.
- *"Mesh"* contains 4792 nodes and 1056 elements.

2. Environment

"Environment" contains all loading conditions defined for *"Model"* in this scenario.
The following tables list local loads and supports applied to specific geometry.

2.1 Thermal Loading

Table F.2-3 Convection Loads

Name	Type	Ambient Temperature	Film Coefficient	Associated Parts
"Convection"	Temperature-Dependent	27.0 °C	*"Stagnant Air - Simplified Case"*	*"Part 3"*, *"Part 2"* and *"Part 1"*

Table F.2-4 Thermal Loads

Name	Description	Value	Reaction	Associated Parts
"Given Temperature"	Edge Temperature	400.0 °C	167.33 W	*"Part 1"*

2.2 Structural Supports

Table F.2-5 Structural Supports				
Name	**Type**	**Reaction Force**	**Reaction Vector**	**Associated Parts**
"Fixed Support"	Fixed Surface	3,420.18 N	[-756.12 N x, 2,099.02 N y, 2,592.3 N z]	*"Part 3"*
"Fixed Support 2"	Fixed Surface	2,810.77 N	[1,699.29 N x, 847.52 N y, -2,072.34 N z]	*"Part 1"*
"Fixed Support 3"	Fixed Surface	3,137.2 N	[-943.18 N x, -2,946.54 N y, -519.96 N z]	*"Part 2"*

3. Solution

"Solution" contains the calculated response for *"Model"* given loading conditions defined in *"Environment"*. It was selected that the program would choose the solver used in this solution.

Thermal expansion calculations use a constant reference temperature of 22.0 °C for all parts in *"Model"*. Theoretically, at a uniform temperature of 22.0 °C no strain results from thermal expansion or contraction.

3.1 Structural Results

Table F.2-6 Values				
Name	**Scope**	**Minimum**	**Maximum**	**Alert Criteria**
"Equivalent Stress"	All Parts In *"Model"*	1.17 MPa	474.36 MPa	None
"Equivalent Strain"	All Parts In *"Model"*	2.2×10^{-5} mm/mm	8.89×10^{-3} mm/mm	None
"Total Deformation"	All Parts In *"Model"*	0.0 mm	1.93 mm	None

Convergence tracking not enabled.

3.2 Thermal Results

Table F.2-7 Values				
Name	**Scope**	**Minimum**	**Maximum**	**Alert Criteria**
"Total Heat Flux"	All Parts In *"Model"*	6.63×10^{-4} W/mm²	0.72 W/mm²	None
"Temperature"	All Parts In *"Model"*	148.79 °C	400.0 °C	None

Convergence tracking not enabled.

Appendix F.3 AsD Alternavie B

1. Model

- The bounding box for all positioned parts in the model measures 304.8 by 330.2 by 355.6 mm along the global x, y and z axes, respectively.
- The model weighs a total of 0.69 kg.

Table F.3-1 Parts

Name	Material	Bounding Box (mm)	Mass (kg)	Nodes	Elements
"Part 1"	*"Aluminum Alloy"*	10.16, 11.68, 10.16	1.98×10^{-3}	623	308
"Part 2"	*"Aluminum Alloy"*	10.16, 11.68, 10.16	1.98×10^{-3}	623	308
"Part 3"	*"Aluminum Alloy"*	25.4, 76.2, 355.6	0.29	1984	272
"Part 4"	*"Aluminum Alloy"*	254.0, 28.58, 25.4	0.18	1160	560
"Part 5"	*"Aluminum Alloy"*	76.2, 254.0, 25.4	0.22	1648	224

1.1 Contact

Table F.3-2 Contact Conditions

Name	Behavior	Associated Parts
"Contact Region"	Bonded	*"Part 3"* and *"Part 1"*
"Contact Region 2"	Bonded	*"Part 4"* and *"Part 1"*
"Contact Region 3"	Bonded	*"Part 3"* and *"Part 2"*
"Contact Region 4"	Bonded	*"Part 4"* and *"Part 2"*
"Contact Region 6"	Bonded	*"Part 5"* and *"Part 3"*

1.2. Mesh

- *"Mesh"*, associated with *"Model"*, has an overall relevance of 0.
- *"Mesh"* contains 6038 nodes and 1672 elements.

No mesh controls specified.

2. Environment

"Environment" contains all loading conditions defined for *"Model"* in this scenario.
The following tables list local loads and supports applied to specific geometry.

2.1 Structural Loading

Table F.3-3 Structural Loads				
Name	**Type**	**Magnitude**	**Vector**	**Associated Parts**
"Pressure"	Surface Pressure	100.0 MPa	N/A	*"Part 2"*
"Pressure 2"	Surface Pressure	100.0 MPa	N/A	*"Part 1"*
"Pressure 3"	Surface Pressure	100.0 MPa	N/A	*"Part 1"*
"Pressure 4"	Surface Pressure	100.0 MPa	N/A	*"Part 2"*

2.2 Structural Supports

Table F.3-4 Structural Supports				
Name	**Type**	**Reaction Force**	**Reaction Vector**	**Associated Parts**
"Fixed Support"	Fixed Surface	20.08 N	[-4.12 N x, -19.38 N y, 3.3 N z]	*"Part 5"*
"Fixed Support 2"	Fixed Surface	16.71 N	[2.8 N x, 16.47 N y, 0.31 N z]	*"Part 4"*
"Fixed Support 3"	Fixed Surface	4.81 N	[1.32 N x, 2.91 N y, -3.6 N z]	*"Part 3"*

3. Solution

"Solution" contains the calculated response for *"Model"* given loading conditions defined in *"Environment"*. It was selected that the program would choose the solver used in this solution.

3.3.1. Structural Results

Table F.3-5 Values				
Name	**Scope**	**Minimum**	**Maximum**	**Alert Criteria**
"Equivalent Stress"	All Parts In *"Model"*	2.73×10^{-3} MPa	203.02 MPa	None
"Equivalent Strain"	All Parts In *"Model"*	5.12×10^{-8} mm/mm	3.8×10^{-3} mm/mm	None
"Total Deformation"	All Parts In *"Model"*	0.0 mm	2.98×10^{-2} mm	None

Convergence tracking not enabled.

Appendix F.4 AsD Alternavie C

1. Model

- The bounding box for all positioned parts in the model measures 304.8 by 330.2 by 355.6 mm along the global x, y and z axes, respectively.
- The model weighs a total of 1.95 kg.

Table F.4-1 Parts

Name	**Material**	**Bounding Box (mm)**	**Mass (kg)**	**Nodes**	**Elements**
"Part 1"	*"Structural Steel"*	254.0, 28.58, 25.4	0.51	1160	560
"Part 2"	*"Structural Steel"*	76.2, 254.0, 25.4	0.61	1648	224
"Part 3"	*"Structural Steel"*	25.4, 76.2, 355.6	0.82	1984	272

1.1 Contact

Table F.4-2 Contact Conditions

Name	**Behavior**	**Associated Parts**
"Contact Region"	Bonded	*"Part 3"* and *"Part 1"*
"Contact Region 2"	Bonded	*"Part 3"* and *"Part 2"*

1.2 Mesh

- *"Mesh"*, associated with *"Model"*, has an overall relevance of 0.
- *"Mesh"* contains 4792 nodes and 1056 elements.

No mesh controls specified.

2. "Environment"
"Environment" contains all loading conditions defined for *"Model"* in this scenario.
The following tables list local loads and supports applied to specific geometry.

2.1 Thermal Loading

Table F.4-3 Convection Loads

Name	**Type**	**Ambient Temperature**	**Film Coefficient**	**Associated Parts**
"Convection"	Temperature-Dependent	27.0 °C	*"Stagnant Air - Simplified Case"*	*"Part 3"*, *"Part 2"* and *"Part 1"*

Table F.4-4 Thermal Loads

Name	**Description**	**Value**	**Reaction**	**Associated Parts**
"Given Temperature"	Edge Temperature	800.0 °C	217.09 W	*"Part 1"*

2.2 Structural Supports

Table F.4-5 Structural Supports				
Name	**Type**	**Reaction Force**	**Reaction Vector**	**Associated Parts**
"Fixed Support"	Fixed Surface	7,938.65 N	[-2,102.43 N x, 4,502.41 N y, 6,191.14 N z]	*"Part 3"*
"Fixed Support 2"	Fixed Surface	7,192.95 N	[4,464.4 N x, 1,877.73 N y, -5,318.07 N z]	*"Part 1"*
"Fixed Support 3"	Fixed Surface	6,859.1 N	[-2,361.97 N x, -6,380.14 N y, -873.07 N z]	*"Part 2"*

3. Solution

"Solution" contains the calculated response for *"Model"* given loading conditions defined in *"Environment"*. It was selected that the program would choose the solver used in this solution.

Thermal expansion calculations use a constant reference temperature of 22.0 °C for all parts in *"Model"*. Theoretically, at a uniform temperature of 22.0 °C no strain results from thermal expansion or contraction.

3.1 Structural Results

Table F.4-6 Values				
Name	**Scope**	**Minimum**	**Maximum**	**Alert Criteria**
"Equivalent Stress"	All Parts In *"Model"*	3.75 MPa	1,064.68 MPa	None
"Equivalent Strain"	All Parts In *"Model"*	2.44×10^{-5} mm/mm	6.92×10^{-3} mm/mm	None
"Total Deformation"	All Parts In *"Model"*	0.0 mm	1.84 mm	None

Convergence tracking not enabled.

3.2 Thermal Results

Table F.4-7 Values				
Name	**Scope**	**Minimum**	**Maximum**	**Alert Criteria**
"Total Heat Flux"	All Parts In *"Model"*	4.2×10^{-4} W/mm²	0.85 W/mm²	None
"Temperature"	All Parts In *"Model"*	105.19 °C	800.0 °C	None

Convergence tracking not enabled.

4. Definition of "Structural Steel"

Table F.4-8 *"Structural Steel"* Properties		
Name	**Type**	**Value**
Modulus of Elasticity	Temperature-Independent	200,000.0 MPa
Poisson's Ratio	Temperature-Independent	0.3
Mass Density	Temperature-Independent	7.85×10^{-6} kg/mm³
Coefficient of Thermal Expansion	Temperature-Independent	1.2×10^{-5} 1/°C
Thermal Conductivity	Temperature-Independent	0.06 W/mm·°C

Table F.4-9 *"Structural Steel"* Stress Limits		
Name	**Type**	**Value**
Tensile Yield Strength	Temperature-Independent	250.0 MPa
Tensile Ultimate Strength	Temperature-Independent	460.0 MPa
Compressive Yield Strength	Temperature-Independent	250.0 MPa
Compressive Ultimate Strength	Temperature-Independent	0.0 MPa

Description: *"Fatigue Data at zero mean stress comes from 1998 ASME BPV Code, Section 8, Div 2, Table 5-110.1"*

Table F.4-10 Thermal Conductivity vs. Temperature

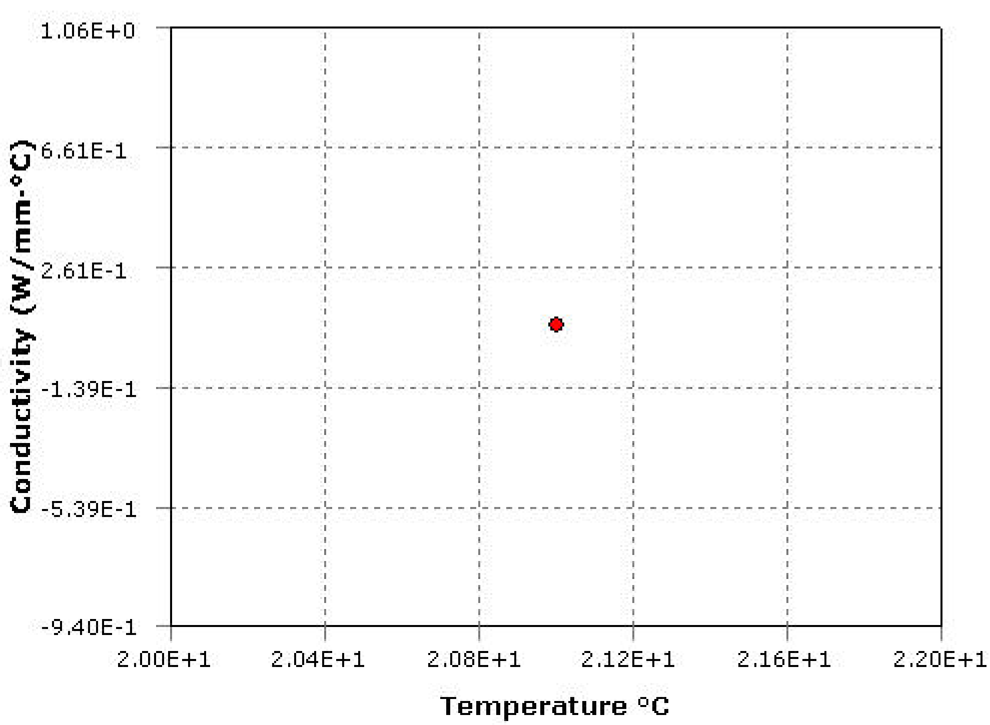
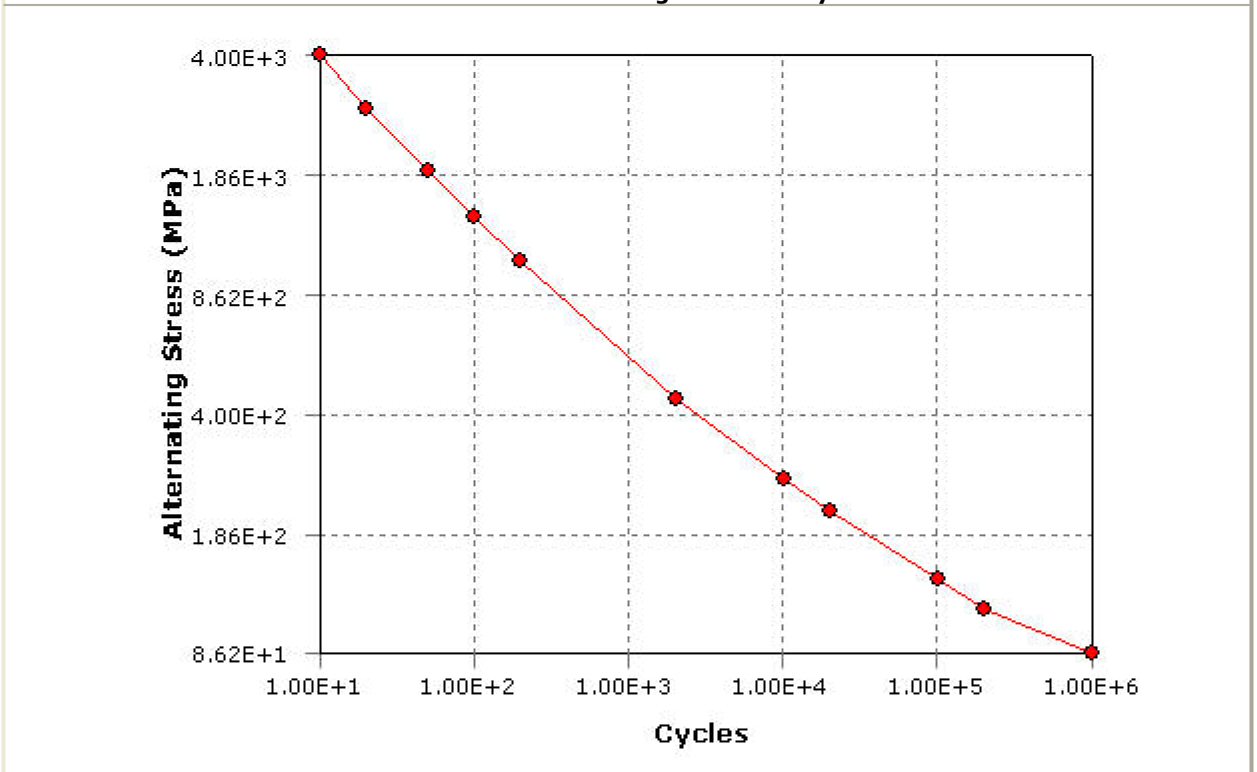


Table F.4-11 Alternating Stress vs. Cycles



Definition of "Stagnant Air - Simplified Case"
Temperature-independent film coefficient: 5.0×10^{-6} W/mm²·°C
Description: *"Stagnant Air Approximations, Any Geometric Configuation, Laminar or Turbulent. Using h = 5 W/m**2 - C."*

BIBLIOGRAPHY

BIBLIOGRAPHY

ABRAHAMSON, S., WALLACE, D., SENIN, N. and SFERRO, P., 2000, Integrated Design in a Service Marketplace, *Computer-Aided Design*, Vol. 32, No. 2, 97-107.

ALLEN, A. H. and Ashley, M. F., 1990, Manufacturing process selection and costing, Proc. Int. Mech. Eng., Part B – Journal of Engineering Manufacture, Vol. 204.

AMBLER, A. P. and POPPLESTONE, R. J., 1975, "Inferring the positions of bodies from specified spatial relationships," *Artificial Intelligence*, Vol. 6, No. 2.

ANSYS, 2002-a, *ANSYS 6.1 Theory reference*, ANSYS, Inc.

ANSYS, 2002-b, *ANSYS 6.1 Documentation*, ANSYS, Inc.

ASHLEY, S., 1991, "Contour: the shape of cars to come?" *Mechanical Engineering*, Vol. 113, No. 5, May, pp. 36-44.

BAE, K. Y. and NA, S. J., 1995, "A study of the effect of pre-straining on angular distortion in one-pass fillet welding incorporating large deformation theory," *Proc. Instn. Mech. Engrs.*, Vol. 209, IMechE, pp. 401-409.

BANDYOPADHYAY, P., GU, F., HUANG, N., MONTGOMERY, P., and XIAO, G., 2001, *e-Manufacturing Activities and Needs at General Motors*, Presentation at the NSF Workshop for Tether-free Technologies for e-Manufactuing, Milwaukee, WI, USA, October 1.

BOOTHROYD, G., DEWTURST, P., and KNIGHT, W., 1994, *Product Design for Manufacture and Assembly*, Marcel Deller, Inc., New York, USA.

BOUJUT, J. F., TICHKIEWITCH, S., and BLANCO, E., 1997, "Integration of Downstream actors in the design process using a dedicated expert CAD tool for forged parts", *Concurrent Engineering Research and Applications*, Vol. 5, No. 4.

BRANDON, D. and KAPLAN, W. D., 1997, *Joining Processes: An Introduction*, West Sussex, England: John Wiley and Sons, p. 131.

BRANS, J. P., MARESCHAL, B., and VINCKE, P., 1984, “PROMETHEE: A new family of outranking methods in multicriteria analysis,” *1984 Int. Operational Research*, IFORMS, Elsevier, North Holland, Amsterdam.

BROWN, D.R., CUTKOSKY, M.R., and TENEBAUM, J.M., 1989, Next-Cut: A Second Generation Framework for Concurrent Engineering, in *Computer-Aided Cooperative Product Development, Proceedings of MIT-JSME Workshop, Cambridge, November 20-21*, Sriram, D., Logcher, R. and Fukuda, S. (eds.), Berlin: Springer-Verlag, pp. 8-25.

BUCHHOLZ, K., 1999, "Alcoa shows aluminum association its concept vehicle," *Automotive Engineering International*, January, pp. 53-55.

BURNS, J. R., WINSTEAD, W. H., and Haworth, D. A., 1989, “Semantic nets as paradigms for both causal and judgmental knowledge representation,” *IEEE Trans. on Systems, Mans, and Cybernetics*, Vol. 19, No. 1, p. 58 – 67.

CATIA, 2002, Tolerance Analysis for Flexible Assembly, CATIA, Version 5 Release 9, Training material, May 2002.

CHANG, B. H., *et al.,* 1999, "A study on the role of adhesives in weld-bonded joints," *Welding Journal*, August, pp. 275-279.

CHANG, S. K., 1989, *Principles of Pictorial Information Systems Design*, Englewood Cliffs, N.J.: Prentice Hall Int’l Ed.

CHARNES, A. and COOPER, W. W., 1961, *Managing Models and Industrial Applications of Linear Programming*, John Wiley, New York, USA.

CHARNES, A., COOPER, W. W., and RHODES, E., 1978, “Measuring the efficiency of decision making units,” *European Journal of Operational Research*, Vol. 2, No. 6, pp. 429-444.

CHENG, K. PAN, P. Y., and HARRISON, D. K., 2001, "Web-based design and manufacturing support systems: implementation perspectives," *Int. Jr. of CIM*, Vol. 14, No. 1, pp. 14-27.

CHUA, C. K., TEH, S. H, and GAY, R. K. L., 1999, "Rapid prototyping versus virtual prototyping in product design and manufacturing," *Int. Jr. of Advanced Manufacturing Technology*, Vol. 15, pp. 597-603.

CHUI, W.H. and WRIGHT, P.K., 1999, “A WWW computer integrated manufacturing environment for rapid prototyping and education,” *International Journal of Computer Integrated Manufacturing*, Vol. 12, No. 1, pp. 54-60.

CUTKOSKY, M. R., TENENBAUM, J. M., and GLICKSMAN, J., 1996, "Madefast: Colloaborative Engineering over the Internet," Communication of ACM, Vol. 39, No. 9, September, pp. 78 - 87.

DEGARMO, E. P., 1984, *Materials and Processes in Manufacturing*, 6th ed., Collier Macmillan Publishers.

DENEUX, D., 1999, "Introduction to assembly features: an illustrated synthesis methodology," *Jr. of Intelligent Manufacturing*, Vol. 10, pp. 29-39.

DETTMER, H. W., 1999, “The conflict resolution diagram: creating win-win solutions,” *Quality Progress*, March, 1999, pp. 41-47.

DOLK, D. and KOTTEMANN J., 1993, "Model Integration and a Theory of Models," *Decision Support Systems*, Vol. 9, No. 1, pp. 51-63.

DOLK, D.R., 1988, "Model Management and Structured Modeling: The Role of an Information Resource Dictionary System," *Communications of ACM*, Vol. 31, No. 6, pp. 704-718.

FANG, Y. and LIOU, F. W., 1997, "Virtual prototyping of mechanical assemblies with deformable components," *Jr. of Manufacturing Systems*, Vol. 16, No. 3, pp. 211-219.

FAWAZ, S. A., 1998, "Application of the virtual crack closure technique to calculate stress intensity factors for through cracks with an elliptical crack front," *Engineering Fracture Mechanics,* Vol. 59, No. 3, pp. 327-342.

FIPER, 2001, National Institute of Standard Technology Annual Review on FIPER, General Electric Aircraft Engines, Springdale, OH, December 12-13.

FLORIDA-JAMES, B., ROSSITER, N., and CHAO, K. M., 2000, "An agent system for collaborative version control in engineering," *Integrated Manufacturing Systems*, Vol. 11, No. 4, pp. 258-266.

FU, Z., DE PENNINGTON, A., and SAIA, S., 1993, "A graph grammar approach to feature representation and transformation," *Int. Jr. of Computer Integrated Manufacturing*, Vol. 6, No. 1 & 2, pp. 137-151.

FUME, A. and KNIGHT, W. A., 1989, "Computer based early cost estimating for sintered powder metal parts," *Proc. of 4th Int. Conf. on Product Design for Manufacture and Assembly*, Rhode Island, USA, June.

GADH, R., 2002, [Online] "Collaborative Product Modeling on the Internet" [Online] http://smartcad.me.wisc.edu/groups/internet/

GAGLIARDI, M. and SPERA C., 1995, "Toward a Formal Theory of Model Integration," *Annals of Operations Research*, Vol. 58, pp. 405-440.

GEE, K., 2001, "Vehicle analysis using an agent-based analysis tools framework," *Proceedings of the ASME 2001 Design Technical Conferences: DETC2001/CIE-21290*, Pittsburgh, PA, September 9-12.

GEOFFIRION, A., DYER, J., and FEINBERG, A., 1972, "An interactive approach for multi-cirterion optimization, with an application to the operation of an academic department," *Management Science*, Vol. 19, pp. 357-368.

GOBELI, D. H. *et al.*, 1998, "Managing conflict in software development teams: a multilevel analysis," *Journal of Product Innovation Management*, Vol. 15, pp. 423-435.

GODH, R., 2002, [Online], "Collaborative Product Modeling on the Internet," http://smartcad.me.wisc.edu/groups/internet/

GORIATCHEV, V. *et al.,* 2001, "Net informational and computational system for CFD researchers," *Proceedings of the ASME 2001 Design Technical Conferences: DETC2001/CIE-21283*, Pittsburgh, PA, September 9-12.

GREENHILL, S. and VENKATESH, S., 1998, "NOETICA: A tool for semantic data modeling," *Information Processing & Management*, Vol. 34, No. 6, pp. 739-760.

GRIFFIOEN, J., MEHROTRA, R. and YAVATKAR R., 1993, "An Object-Oriented Model for Image Information Representation," *Proceedings of the second international conference on Information and knowledge management* , pp. 393–402.

HAN, J.H. and REQUICHA, A.A.G., 1998, "Modeler-independent feature recognition in a distributed environment," *Computer-Aided Design*, Vol. 30, No. 6, 453-463.

HINLOOPEN, E., NIJKAMP, P., and RIETVELD, P., 1983, "The regime method: A new multi-criteria technique," Hansen, P. (Ed.), *Essays and Surveys on Multiple Criteria Decision Making*, Springer, pp. 146-155.

ISIGHT, 2002, 2002 International iSIGHT Users' Conference and FIPER Workshop, Washington, D.C., July 15 to July 18.

WELCH, J., 1996, Jack Welch and GE, *Business Week*, Oct. issue 1996.

JEONG, S. K. and CHO, H. S., 1997, "An analytical solution for transient temperature distribution in fillet arc welding including the effect of molten metal," *Journal of Engineering Manufacture*, Vol. 211, No. B7, pp. 63-72.

JONES, C.V., 1992, "Attributed Graphs, Graph-Grammars, and Structured Modeling," *Annals of Operations Research*, Vol. 38, pp. 281-324.

KALPAKJIAN, S., 1995, *Manufacturing Engineering and Technology*, 3rd ed., Addison-Wesley Co..

KAO, Y.C. and LIN, G.C.I., 1996, “CAD/CAM Collaboration and Remote Machining,” *Computer Integrated Manufacturing Systems*, Vol. 9, No. 13, pp. 149-160.

KAO, Y.C. and LIN, G.C.I., 1998, “Development of a collaborative CAD/CAM system,” *Robotics and Computer-Integrated Manufacturing*, Vol. 14, No. 1, pp. 55-68.

KEENEY, R. L. and RAIFFA, H., 1976, *Decisions with Multiple Objectives: Preference and Value Tradeoffs*, Wiley, New York.

KIM, C. Y., KIM N. K., KIM Y. H., KANG, S. H., and O'GRADY, P., 1998, Iowa Internet laboratory Technical report TR98-02, "Distributed Concurrent Engineering: Internet-Based Interactive 3-D Dynamic Browsing and Markup of STEP Data", March 4.

KIM, D. W., 1994, *A feature-based process planning system for robotic arc welding*, unpublished Ph. D. Dissertation, Department of Precision Engineering, Hokkaido University, Japan, pp. 25-33.

KIM, K. Y., KIM, D. W., and NNAJI, B. O., 2002, "Robot arc welding task sequencing using genetic algorithms," *IIE Transaction*, Vol. 34, No. 10, pp. 865-880.

KIM, D. W., LIU, T. L., NNAJI, B. O., and KIM, K. Y., 2003-a, "Sheet metal weld assemblies modeling with spatial relationships," submitted to *Jr. of Manufacturing System*.

KIM, K. Y., WANG, Y., MUOGBOH, O. S., and NNAJI, B. O., 2003-b, “Assembly design formalism for service-based collaborative assembly design environment,” submitted to *Computer Aided Design*.

KIM, K. Y., MUOGBOH, O. S., WANG, Y., NNAJI, B. O., and LOVELL, M., 2003-c, “Virtual Assembly Analysis in Service-oriented Collaborative *e*-Design and Realization Environment,” submitted to *International Journal of Production Research*.

KLEIN, M., 1991, “Supporting conflict resolution in cooperative design system,” *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 21, No. 6, pp.1379-1390.

KORHONEN, P. and LAAKSO, J., 1986, “A visual interactive method for solving the multiple criteria problem,” *European Journal of Operational Research*, Vol. 24, pp. 277-287.

KORHONEN, P. and WALLENIUS, J., 1988, “A Pareto race,” *Naval Research Logistics*, Vol. 35, pp. 615-623.

KORHONEN, P., WALLENIUS, J., and ZIONTS, S., 1984, “Solving the discrete multiple criteria problem using convex cones,” *Management Science*, Vol. 30, pp. 1336-1345.

KORHONEN, P., 1986, “A hierarchical interactive method for ranking alternatives with multiple qualitative criteria,” *European Journal of Operational Research*, Vol. 24, pp. 265-276.

KORHONEN, P., 1988, “A visual reference direction approach to solving discrete multiple criteria problems,” *European Journal of Operational Research*, Vol. 34, pp. 152-159.

KRISHNAMURTHY, L. and LAW, K. H., 1997, "A data management model for collaborative design in a CAD environment," *Engineering with Computers*, Vol. 13, pp. 65-86.

LARSON, J. and CHENG, H. H., 2000, Object-Oriented Cam Design Through the Internet, *Journal of Intelligent Manufacturing*, Vol. 11, No. 6, pp. 515-534.

LEBACQ, C., BRECHET, Y., SHERCLIFF, H. R., JEGGY, T., and SALVO, L., 2002, “Selection of joining methods in mechanical design,” *Materials and Design*, Vol. 23, pp. 405-416.

LEE, J.Y., HAN, S.B., KIM, H. and PARK, S.B., 1999, “Network-Centric Feature-Based Modeling,” *IEEE Proceedings of The Seventh Pacific Conference on Computer Graphics and Applications, October 5-7, Seoul, Korea*, pp.280-289

LEWIS, R. W., MORGAN, K., THOMAS, H. R., and SEETHARAMU, K. N., 1996, *The finite Element Method in Heat Transfer Analysis*, John Wiley & Sons Ltd, West Sussex, England.

Lim, K.H., Benbasat, I. and Ward L.M., 2000, “The Role of Multimedia in Changing First Impression Bias”, *Information Systems Research*, Vol. 11, No. 2, pp. 115-136.

LIU, H. C. and NNAJI, B. O., 1991, "Design with spatial relationships," *Jr. of Manufacturing Systems*, Vol. 10, No. 6.

LIU, T. L. and NNAJI, B. O., 2003-a, , "A framework of design advisory system for mechanical assemblies," submitted to *Int. Jr. of Production Research*.

LIU, T. L. and NNAJI, B. O., 2003-b, "Realization and management of product design constraints in CAD modeling," submitted to *Computer Aided Design*.

MASUBUCHI, K., 1980, *Analysis of Welded Structures: Residual Stresses, Distortion, and their Consequences*, New York: Pergamon Press Inc., pp. 60-236.

MENZEMER, C. C., FEI, L., and SRIVATSAN, T. S., 1999, "Design criteria for bolted connection elements in aluminum alloy 6061," *Journal of Mechanical Design*, Vol. 121, No. 3, pp. 348-358.

MERVYN, F., KUMAR, S. A., BOK, S. H., and NEE, A. Y. C., 2003, "Development of an Internet-enabled interactive fixture design system," *Computer-Aided Design*, Vol. 35, pp. 945-957.

MESSLER, R. W., 1993, *Joining of Advanced Materials*, Butterworth-Heinemann, 1993.

MILEWSKI, J. O. and BARBE, M. B., 1999, "Modeling and analysis of layer melting with in a narrow groove weld joint," *Welding Journal*, April, pp. 109-115.

Mollon, O., Tilley, S., and Warman, E. A., 1998, *Design for Manufacutring and Assembly*, Captman and Hall, London, UK.

MOON, H. S., and NA, S. J., 1997, "Optimum design based on mathematical model and neural network to predict weld parameters for fillet joints," *Jr. of Manufacturing Systems*, Vol. 16, No. 1, pp. 13-23.

MUELLER, A., 1999, [Online], Technical White Paper: "Shared Engineering with OneSpace from CoCreate", January 25.

MUHANNA, W., 1993, "An Object-Oriented Framework for Model Management and DSS Development," *Decision Support Systems*, Vol. 9, No. 2, pp. 217-229.

NILSSON, Nils J., 1980, *Principles of Artificial Intelligence*, Tioga Publishing Co.

NNAJI, B. O., LIU, H. C., AND REMBOLD, U., 1993, "A product modeler for discrete components," *Int. Jr. of Production Research*, Vol. 31, No. 9, pp. 2017-2044.

NNAJI, B. O., GUPTA, D., and KIM, K. Y., 2003-a, "Welding distortion minimization for an aluminum alloy extruded beam structure using a 2D model," submitted to *Jr. of Manufacturing Sci. and Eng.*.

NNAJI, B. O., WANG Y., KIM, K. Y., and MUOGBOH, O, S., 2003-b, “PEGASUS: A Service-Oriented Product Design and Realization Engineering System Over the Internet,” submitted to *IIE Transactions*.

NSF WORKSHOP, 2000, National Science Foundation Workshop on *e*-Product Design and Realization for Mechanically Engineered Products, University of Pittsburgh, Pittsburgh, PA, October 19-20.

ONESPACE, 2002, [Online], CoCreate Corporate, http://www.cocreate.com.

PAHNG, F., BAE, S. and WALLACE, D., 1998-a, “A Web-based Collaborative Design Modeling Environment,” *IEEE Proceedings of the Seventh workshop on Enabling Technologies: Infrastructure for collaborative Enterprises*, June 17-19, Stanford, California, pp.161-167.

PAHNG, F., SENIN, N. and WALLACE, D., 1998-b, “Distribution modeling and evaluation of product design problems,” *Computer-Aided Design*, Vol. 30, No. 6, 411-423.

PASTIJN, H. and LEYSEN, J., 1989, “Constructing an outranking relation with ORESTE,” *Mathematical and Computer Modeling*, Vol. 12, pp. 1255-1268.

PEAK, R. S., FULTON, R. E., NISHIGAKI, I., and OKAMOTO, N., 1998, "Integrating Engineering Design and Analysis Using a Multi-representation Approach," *Engineering with Computers*, Vol. 14, pp. 93-114.

PEGASUS, 2003, NSF IUCRC for *e*-Design: Strategic Planning Meeting, Pittsburgh, PA, USA, Dec 9 to 10, www.e-designcenter.info.

PEÑE-MORA, F. *et al*, 1995, “Conflict mitigation system for collaborative engineering,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 9, pp. 101-124.

PRATT, M. J., 1994, "Virtual prototypes and product models in mechanical engineering," *Proc. IFIP WG 5.10 on Virtual Environments and their Applications and Virtual Prototyping*, pp. 113-128.

PTC, 2000, [Online] "PTC Windchill Pro/INTERLINK Datasheet", http://www.ptc.com/products/windchill/engineering/ds_prointralink.htm, (2/15/00).

QIANG, L., ZHANG, Y. F. and NEE, A.Y.C., 2001, "A Distributive and Collaborative Concurrent Product Design System through the WWW/Internet," *International Journal of Advanced Manufacturing Technology*, Vol. 17, No. 5, pp. 315-322.

RADA, R., HAFEDH, M., BICKNELL, E, and BLETTNER, M., 1989, "Development and application of a metric on semantic nets," *IEEE Trans. on Systems, Mans, and Cybernetics*, Vol. 19, No. 1, p. 17 – 30.

RADAJ, D., 1992, *Heat Effects of Welding*, Springer-Verlag, New York, USA.

RAHMAN, A., *et al*., 2000, "Boundary correction factors for elliptical surface cracks emanating from countersunk rivet holes," *AIAA Journal*, Vol. 38, No. 11, pp. 2171-2175.

REKIEK, B., DE LIT, P., and DELCHAMBRE, A., 2002, "Hybrid assembly line design and user's preferences," *International Journal of Production Research*, Vol. 40, No. 5, pp. 1095-1111.

RÉMONDINI, L., LÉON, J. C., and TROMPETTE, P., 1998, "High-level Operations Dedicated to the Integration of Mechanical Analysis within a Design Process," *Engineering with Computers*, Vol. 14, pp. 81-92.

ROJAS, E. M. and SONGER, A. D., 1999, "Web-centric Systems: a new paradigm for collaborative engineering," *Journal of Management in Engineering*, Vol. 15, No. 1, pp. 39-45.

ROUBENS, M., 1982, "Preference relations on actions and criteria in multi-criteria decision making," *European Journal of Operational Research*, Vol. 10, pp. 51-55.

ROY, B., 1973, "How outranking relation helps multiple criteria decision making," Cochrane, J. and Zeleny M. (Eds.), *Multiple criteria decision making*, University of South Carolina Pross, pp. 179-201.

RUSSELL, S. and NORVIG, P., 1995, *Artificial Intelligence: A Modern Approach*, Prentice Hall, Englewood Cliffs, New Jersey.

RYAN, L. and MONAGHAM, J., 2000, "Failure mechanism of riveted joint in fibre metal laminates," *Journal of Materials Processing Technology*, Vol. 103, No. 1, pp. 36-43.

SAATY, T. L., 1974, "Measuring the fuzziness of sets," *Journal of Cybernetics*, Vol. 4, pp. 53-61.

SAATY, T. L., 1977, “A scaling method for priorities in hierarchical structures,” *Journal of Mathematical Psychology*, Vol. 15, No. 3, pp. 234-281.

SAATY, T. L., 1996, *Decision Making with Dependence and Feedback: The Analytic Network Process*, RWS Publications, Pittsburgh, PA, 1996.

SAHU, K. and GROSSE, I. R., 1994, "Current iterative design and the integration of finite element analysis results," *Engineering with Computers*, Vol. 10, pp. 245-257.

SHAH, J. J. and ROGERS, M. T., 1988, "Functional Requirements and Conceptual Design of the Feature-Based Modeling System," *Computer-Aided Engineering Journal*, Vol. 5, February 1988, pp. 9-18.

SHANBHAG, S., GROSSE, I. R., WILEDEN, J.C, and KAPLAN, A., 2001, “A Meta-Object Approach to Finite Element Modeling”, *Proceedings of the ASME 2001 Design Technical Conferences: DETC2001/DAC-21062*, Pittsburgh, PA, September 9-12.

SHEA, C., REYNOLD, C., and DEWHURST, P., 1989, “Computer aided material and process selection,” *Proc. Of 4th Int. Conf. on Product Design for Manufacture and Assembly*, Vol. I, Rohde Island, USA, June.

SHEEHY, M. and GROSSE, I. R., 1997, "An object-oriented blackboard based approach for automated finite element modeling and analysis of multichip modules," *Engineering with Computers*, Vol. 13, pp. 197-210.

SHYAMSUNDAR, N., ASHAI, Z., and GADH, R., 1998, "Geometry-based metric formulation and methodology to support virtual design for disassembly," *Eng. Design and Automation*, Vol. 4, No. 1, pp. 13-26.

SIEGEL, J., 2000, *CORBA 3: fundamentals and programming*, 2nd ed., John Wiley & Sons, Inc.

SINGH, A., and JOHNSON, H. M., 1998, “Conflict management diagnosis at project management organizations,” *Journal of Management in Engineering*, Vol. 14, No. 5, pp. 48-63.

SMITH, C.S. and WRIGHT, P.K., 1996, “CyberCut: A World Wide Web based design-to-fabrication tool,” *Journal of Manufacturing Systems*, Vol. 15, No. 6, pp. 432-441.

SRINIVASAN, H., FIGUEROA, R., and GADH, R., 1999, "Selective disassembly for virtual prototyping as applied to de-manufacturing," *Robotics and Computer Integrated Manufacturing*, Vol. 15, pp. 231-245.

SRIRAMAN, V., 1999, "Assembly modeling," *Eng. Design Graphics Journal*," Vol. 63, No. 1, pp. 9-19.

STEUER, R. and CHOO, E. U., 1983, "An interactive weighted Tchebycheff procedure for multiple objective programming," *Mathematical Programming*, Vol. 26, pp. 326-344.

SU, D. and AMIN, N., 2001, "A CGI-based approach for remotely executing large program for integration of design and manufacture over the Internet," *Int. Jr. of CIM*, Vol. 14, No. 1, pp. 55-65.

SUBRU, R., SANDERSON, A. C., HOCAUĞLU, C., and GRAVES, R. J., 1999, "Evolutionary decision support for distributed virtual design in modular product manufacturing," *Production Planning and Control*, Vol. 10, No. 7, pp. 627 – 642.

SWAELENS, B. and KRUTH J. P., 1993, "Medical applications of rapid prototyping techniques", *Proc. of the 4th Inter. Conf. on Rapid Prototyping*, University of Dayton, June, pp. 107-120.

SWIFT, K. G. and BOOKER, J. D., 1997, *Process selection from design to manufacture*, Arnold, London, 1997.

TARNG, Y. S., WU, J. L., YEH, S. S., and JUANG, S. C., 1999, "Intelligent modeling and optimization of the gas tungsten arc welding process," *Jr. of Intelligent Manufacturing*, Vol. 10, pp. 73-79.

TSAI, C. L., PARK, S. C., and W. T. CHENG, 1999, "Welding Distortion of a Thin-Plate Panel Structure," *Welding Journal*, May pp. 156-165.

TSAI, C., KIM, D., JAEGER, J., SHIN, Y., FENG, Z., and PATRITAN, J., 2001, "Design analysis for welding of heavy W shapes," *Welding Journal*, Feb., pp. 35-41.

U.S. DEPARTMENT OF COMMERCE, 2002, U.S. Department of Commerce Industry sector data, http://www.ita.doc.gov/td/industry/otea/usito98 /tables_naics.htm.

VAN HOLLAND, W. and BRONSVOORT, W. F., 2000, "Assembly features in modeling and planning," *Robotics and Computer Integrated Manufacturing*, Vol. 16, pp. 277-294.

VICUÑA, F., 1990, *Semantic Formalization in Mathematical Modeling Languages*, Ph.D. Dissertation, Computer Science Department, UCLA.

WAGNER, R., CASTANOTTO, G., and GOLDBERG, K., 1997, "FixtureNet: interactive computer-aided design via the World Wide Web," *Int. Jr. of Human-Computer Studies*, Vol. 46, pp. 773-788.

WEAVER, M. A., 1999, "Determination of weld loads and throat requirements using finite element analysis with shell element models - a comparison with classical analysis," *Welding Journal*, April, pp. 116-126.

WELDING WORKBOOK, 2001, *Welding Journal*, Vol. 80, No. 8, August, pp. 67-68.

WHITNEY, D. E., MANTRIPRAGADA, R., ADAMS, J. E., and RHEE, S. J., 1999, "Toward a theory for design of kinematically constrained mechanical assemblies," *Int. Jr. of Robotics Research*, Vol. 18, No. 12, pp. 1235-1248.

WHITNEY, D. E., 2001, “A design procedure applicable to different classes of assemblies”, *Proceedings of the ASME 2001 Design Technical Conferences: DETC2001/CIE-21304*, Pittsburgh, PA, September 9-12.

WINDCHILL, 2002, [Online], Parametric Technology Corporate, http://www.ptc.com.

WINSTON, W. L., 1993, *Operations Research – Applications and Algorithms* (3rd Ed.), Duxbury Press, Belmont, CA, pp. 798-806..

WOODWARD, J. A. and CORBETT, J., 1989, “An expert system to assist the design for manufacture of die cast components,” *Proc. of 1989 Int. Conf. on Eng. Design*, (ICED89), Vol. II, Harrogate, UK, August.

W3S DTD, 2000, Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation 6, October 2000, http://www.w3.org/TR/REC-xml.

W3S SCHEMA, 2001, XML Schema work draft, http://www.w3.org/TR/xmlschema-0/, http://www.w3.org/TR/xmlschema-1/, and http://www.w3.org/TR/xmlschema-2/.

XIONG, Y. and BEDAIR O. K., 1999, "Analytical and finite element modeling of riveted lap joints in aircraft structure," *AIAA Journal*, Vol. 37, No. 1,, pp. 93-99.

ZADEH, L. A., 1965, “Fuzzy Sets,” *Information and Control*, Vol. 8, No. 3, pp. 338-353.

ZELENY, M., 1982, *Multiple Criteria Decision Making*, McGraw Hill.

ZENGER, D. C. and BOOTHROYD, G., 1989, "Selection of manufacturing processes and materials for component parts," *Proc. of 4th Int. Conf. on Product Design for Manufacture and Assembly*, Rhode Island, USA, June.

ZHA, X. F., 2002, "A knowledge intensive multi-agent framework for cooperative/collaborative design modeling and decision support of assemblies," *Knowledge-Based Systems*, Vol. 15, pp. 493-506.

ZIMMERMANN, H. J., 1991, *Fuzzy Set Theory and its Applications*, 2nd ed., Kluwer, Boston.