AN APPROACH TO COLLABORATIVE ASSEMBLY DESIGN MODIFICATION AND ASSEMBLY PLANNING

LU CONG

(*B.Eng., M.Eng.*)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2006

ACKNOWLEDGEMENTS

My PhD research and thesis writing was carried out and completed with the kind advice and guidance of my supervisors, Associate Professor Jerry, Fuh Ying Hsi, and Associate Professor Wong Yoke San. They always encourage me to go ahead when I met difficulties during the study and research, and always provide me good suggestions on my research. Their strict requirements, wise insight, timely feedback, and careful revision ensured my research project can be completed. Hereby, I would like to show the most sincere gratitude to them.

I would also like to show the sincere gratitude to Dr. Li Weidong, Dr. Lu Yiqiang and Mr. Zhou Hong from SIMTech, and Dr. Qiu Zhimin from LCEL for their kind cooperation and help during my PhD research. And I would also like to thank National University of Singapore and SIMTech for the financial support on my PhD research.

In addition, I would also like to thank Associate Professor Zhang Yun Feng and Associate Professor Loh Han Tong, for their comments and suggestions on my research during my PhD qualification exam.

In LCEL, I would like to thank the friends Dr Feng Wei, Dr Huang Xingang, Fan Liqing, Mervyn, Chen Xiao Long, Wu Yifeng, Tang Yaxin, Li Min and Zhu Huabing, for the help and the friendly atmosphere they made.

Finally, I would like to express my special gratitude to my family members, especially my parents, my wife, and my daughter, for their spiritual encouragement and support accompanying with me throughout my PhD research career.

Table of Contents

CHAPTER 5 AN ENHANCED ASSEMBLY PLANNING APPROACH USING A MULTI-OBJECTIVE GENETIC ALGORITHM…………………86

CHAPTER 7 COLLABORATIVE ASSEMBLY PLANNING…………………..141

Summary

Product assembly design and assembly planning are two important steps in product development. Effective and rapid assembly design and assembly planning can shorten the product development life cycle, reduce the development cost, and thereby help manufacturers to enhance profit. The research presented in this thesis investigates a collaborative assembly design modification and assembly planning approach to improve the efficiency of product assembly design and assembly planning in a collaborative design environment.

In order to realize effective collaborative assembly design, the design modification issues are first addressed, and a methodology to support the effective design modification in collaborative assembly design is developed. A feature-based hierarchical co-assembly representation model is proposed and a design modification propagation control mechanism is developed, upon which a three tier client-server system framework that is suitable for realizing the design modification in collaborative assembly design is proposed and developed.

To realize effective assembly planning, an enhanced assembly planning approach using a multi-objective Genetic Algorithm (GA) is developed. In this approach, the tolerance influence on product assemblability in different assembly sequences is considered and used as a constraint in assembly planning. A concept called Sensitive Tolerance in Assembly is proposed and its influence on the assembly is investigated. The approach using transformation matrix is proposed to determine the geometric deviations of mated features caused by the tolerance and assembly clearance, and their propagation and accumulation in the different assembly sequences. Using this approach, the relative assemblability of different assembly sequences can be concluded. In order to find more feasible non-dominated solutions, a Genetic Algorithm with multiple search directions is proposed, and different fitness functions are built using the fuzzy weights distribution algorithm proposed in this research. Using this algorithm, more non-dominated solutions can be found while the experience of the decision maker is considered.

To evaluate the product assembly design and modification, it is discussed how to identify the potential design problems through the evaluation of the assembly planning results. According to the design problems, a set of redesign guidelines is proposed. These guidelines focus on the two following areas: to improve the product assemblability, and to reduce the assembly cost of the product. These redesign guidelines can effectively help the designer improve the product design considering the detailed assembly process in the design stage. Therefore, the design modification or redesign should be more practical and feasible.

To speed up the assembly planning process, especially for the complex product, a collaborative assembly planning approach is proposed based on the aforementioned GA-based assembly planning approach. It enables several planners to carry out the assembly planning collaboratively. A Browser/Server system framework is developed, and an algorithm to check the feasibility of the subassembly task assignment is proposed. During assembly planning, through the subassembly task assignment,

feasibility check of the subassembly task assignment, parameter selection, the assembly can be decomposed into several subassemblies, and for each subassembly, the non-dominated solutions can be derived considering the detailed assembly condition and facilities, and the experience of the planners.

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Background

Product assembly design and assembly planning are two important steps during product development. Effective and rapid assembly design and assembly planning can shorten the product development life cycle, reduce the development cost, and thereby help manufacturers to enhance profit.

With the development of the Internet and computer technology, the traditional assembly design and assembly planning have evolved to collaborative assembly design and assembly planning in an Internet-enabled working environment, to speed up the product development process. Therefore, research to facilitate and realize collaborative assembly design and assembly planning in an Internet-enabled environment has attracted much attention. In the following sections, we will discuss research issues in collaborative assembly design and assembly planning, respectively.

1.2 Research issues in collaborative assembly design

Assembly design is an important step in product design, as it enables designers to provide a complete concept of a product that usually consists of many different components. Generally, in traditional computer-aided assembly design, each part is designed in a standalone computer system and then assembled into a sub-assembly or a more complex assembly by an individual or a group of designers in the same location. With the advancement of the Internet and communication technologies, more and more products are designed and manufactured in different locations to meet the fast-changing market requirements. Rezayat [Rezayat, 2000] reported that about 50-80% of the components in a product from Original Equipment Manufacturers (OEMS) are outsourced to external suppliers geographically dispersed. Hence, products are usually divided into several sub-assemblies or even more detailed parts, and are assigned to multiple designers located in different sites. These designers can design and assemble the parts collaboratively and synchronously through the Internet to speed up the design process. Usually the following four consecutive steps should occur in the product design: firstly, each designer in a different location designs the parts assigned to him according to the design requirements; secondly, the designers assemble these parts into a sub-assembly or more complex assembly product collaboratively through the Internet; thirdly, when one designer modifies his part, the modification should be propagated to the associated parts designed by other designers located in other sites to maintain the validity and consistency of the whole assembly; finally, when the modification of all of the affected parts are completed, a new assembly product will be re-assembled collaboratively.

From the above-mentioned four steps in co-assembly design, the first step is basically the same as the computer-aided design in a standalone computer system, while the second and the fourth steps are mainly the geometric assembly modeling functions but realized in a collaborative design environment. However, the third step is much different with the traditional computer-aided assembly design. In a collaborative design environment, when each designer finishes designing his parts according to the initial design requirements, those parts should be assembled together correctly. However, if a designer modifies his design after the assembly process is finished, he may not know how the modification can affect the other parts developed by other designers because the whole assembly relationship with other associated parts may not be completely known to him, and neither are the geometric shape and dimension of the affected parts designed by others. So, it is unavoidable that some conflicts arise during the co-assembly design process.

Therefore, a methodology to support effective design modification in collaborative assembly design is an important research issue.

1.3 Research issues in collaborative assembly planning

Assembly planning is another important step during product development. The objective of assembly planning is to find a feasible assembly sequence with the minimum assembly cost and assembly time. Because assembly costs account for 10-30% of total industrial product labor costs [Nevis and Whitney, 1980] and as much as 50% of the product manufacturing costs [Rembold et al, 1985], effective assembly planning can significantly reduce the product development cost, and thereby improve the profit margin.

Besides the above, effective assembly planning at the design stage can make the assembly design more practical when considering the detailed assembly process of the product. The assembly planning results, that represent the feasibility and difficulty of the product assembly process and the assembly cost, can provide appropriate decision support to the designers, and help them to identify the design problems and make the appropriate design modification or redesign in the early design stage. Therefore, the product development lead time can be greatly shortened.

Due to the importance of assembly planning, it has attracted much research attention in recent years. In order to improve the efficiency of assembly planning, the traditional assembly planning approach using graph-based approach has evolved to approaches using artificial intelligence, such as genetic algorithm, and the working mode has evolved from the single-user assembly planning to the multi-user collaborative assembly planning to speed up the assembly planning process. In the assembly planning area, the following research issues are very important and need to be addressed:

- How to evaluate the product assemblability in different assembly sequences?
- How to derive more effective solutions for decision maker considering different assembly conditions?
- How to evaluate the assembly design from the assembly planning results?
- How to realize the collaborative assembly planning effectively?

The above research can further facilitate the efficiency of assembly planning.

1.4 Organization of the thesis

The thesis is organized as follows:

Chapter 2 is a systematic literature review of the previous works on assembly design and assembly planning, and the objectives of the research are clarified based on the review.

Chapter 3 discusses the design modification issues in collaborative assembly design. An assembly representation model is proposed and a new definition of the assembly feature is given to resolve the collaborative assembly design issues. In order to realize the design modification, a design modification propagation control mechanism is proposed, and a system framework that is suitable for realizing the design modification is also proposed and developed.

Chapter 4 investigates an approach to evaluate the product assembability in different assembly sequences considering the influence of tolerance and assembly clearance. This approach will be used to assist the downstream assembly planning system to find optimal assembly sequences with good assemblability, and can also help the designer to find the design problems.

Chapter 5 proposes an enhanced assembly planning approach using a multi-objective genetic algorithm. The influence of tolerance and clearance on product assemblability in different assembly sequences is considered and used as a constraint in assembly planning. For more comprehensive search for feasible non-dominated solutions, this chapter proposes a multi-objective genetic algorithm which establishes different fitness functions through a fuzzy weight distribution algorithm. It also considers the experience of the decision maker.

Chapter 6 discusses the potential design problems which can be identified through the evaluation of the assembly planning results, and further proposes redesign guidelines to help the designer to make appropriate design modification or redesign considering the detailed assembly process in the design stage.

Chapter 7 presents a collaborative assembly planning approach based on the GA-based assembly planning approach proposed in chapter 5. The system framework and working mechanism are proposed and developed, and the detailed collaborative assembly planning procedure is illustrated.

Chapter 8 concludes the thesis by summarizing the main contributions of the research, and suggesting proposals for future research.

Chapter 2 Literature Review

This chapter reviews previous works on product assembly design and assembly planning, and the research objectives are clarified based on the literature review. Section 2.1 reviews previous works on assembly design and Section 2.2 reviews those on evaluation of the tolerance influence on product assemblability, while Section 2.3 reviews works on assembly planning. Based on the review, Section 2.4 further elaborates and clarifies the research objectives of the thesis.

2.1 Previous works on assembly design

In assembly design, one key aspect is the development of a proper assembly representation approach to specify the relationship between different parts. The representation approaches for assembly design can be categorized into two main areas: representation approach for traditional assembly design, and representation approach for collaborative assembly design.

2.1.1 Assembly representation approach in traditional assembly design

In traditional assembly design, some researchers used different methods to represent the assembly, Shah and Rogers [1993] proposed an assembly representation approach that can encapsulate the relationships between the elements of each level of the assembly- sub-assembly, parts, form features and feature-producing volumes. In this paper, assembly features are defined as an association between two form features on different parts, and it encodes mutual constraints on mating features based on their shape, dimensions, position and orientation. Ye et al. [2000] proposed a feature-based and object-oriented representation to represent hierarchical assembly of injection moulds. Besides the feature paradigm to the assembly design, it also encapsulates operational functions and geometric constraints, and thus enables the routine process of assembly design such as interference check within a mould assembly. Holland and Bronsvoort [2000] defined the assembly feature as an information carrier for assembly-specific information. It carries all assembly-specific information within modeling and planning. Then the assembly features can be used in assembly planning, such as stability analysis, motion planning, assembly sequence planning and so on. Yin et al. [2003] proposed a hierarchical connector-based structure to represent assembly, using a connector to provide constraints on the corresponding joined components to ensure that they perform required functions. Based on this structure, a set of assembly precedence graphs can be generated for assembly sequence planning. The other definitions of assembly features include: De Fazio [1990] defined assembly feature as elementary relation between components extended with some assembly information; Lee and Andrews [1985] defined it as elementary relations between components; Sodhi and Turner [1991] defined it as a collection of elementary relations and matching form features.

The above researchers proposed the approaches to represent the assembly relationship between different components in the assembly, but did not consider the collaboration between different designers in different locations, so those assembly representations cannot be adapted to the assembly design in the collaborative design environment.

2.1.2 Assembly representation approach in collaborative assembly design

In order to address the above problem, some researchers proposed new approaches. Chen et al. [2004] proposed a co-assembly representation including Master Assembly Model (MAM) and Slave Assembly Model (SAM). MAM is a complete representation stored in the server, and SAM is a simplified version of MAM used for visualization in the client. The MAM includes the composite component information, atomic component information, and link entity information. This representation can realize the co-assembly modeling, but it cannot realize design modification in a collaborative design environment. Kim et al. [2004] proposed design formalism in a co-assembly design environment to capture the non-geometric aspects of a designer's intent on an assembly, with focus on the joining process used in the assembly. The purpose of joining relations is to infer mathematical and physical implications, and the use of an assembly design model is to support some assembly design activities, such as joining analysis, process planning and so on. However the design modifications in a co-assembly design environment was not considered either.

2.1.3 Approaches for design modification in collaborative assembly design

Recently, efforts have been made to enhance the existing CAD systems to deal with the collaborative assembly design and design modification. Some commercial

CAD systems, for example, the Pro/ENGINEER Wildfire [PTC, 2004] provides the Peer-to-Peer Design Conferencing package to make it easier for engineers to collaborate simultaneously with one another by enabling multiple users to share control of a live design session; and Alibre Design [Alibre, 2004] and OneSpace [CoCreate, 2004] allow multiple designers to set up a session to discuss with each other through messages, video, audio, etc.

Besides the above commercial systems, there are some recent works related to assembly design and design changes. Noort et al. [2002] presented a multiple-view feature modeling approach to integrate part design and assembly design. This approach integrates a part’s detailed design view and the assembly design view by linking the part model with the associated components in an assembly model, and thus enables the modification propagation between the two views. Furthermore through connection features this modification of the component can be propagated to the component connected with it in the assembly design view. Based on this approach, Bidarra et al. [2002] proposed a collaborative framework for integrated part and assembly modeling; in this framework, the team members can discuss the assembly design issues through a collaborative validity maintenance scheme including phone, chat channel, shared camera, etc. Shyamsundar and Gadh [2001] defined an assembly feature as a property of an assembly unit with respect to other components. In addition, assembly features can be classified into relational assembly features and assembly form features. Relational features indicate a specific relation between two geometric features. The assembly form features are formed by certain shape features belonging to two components that can be joined together. In addition, they proposed interface assembly features as a subset of the assembly features. These interface assembly features are considered as hard constraints and cannot be modified unilaterally by the designer. It can only be changed through negotiation with other designers.

In the above approaches, the collaboration in design modification of an assembly is conducted through on-line chatting, involving negotiation with other designers working simultaneously in different geographical locations. Those approaches to collaborative design modification generally do not allow a designer to make a design modification asynchronously, i.e., without the on-line attention of some other designers. However, in geographically dispersed environment, it is not easy to get all designers to come together at the same time to work simultaneously, especially when they work in different time zones. So, sometime those approaches cannot realize design modification when some designers are absent.

Some other research works related to the design modification in a collaborative design environment are as follows. Mervyn et al. [2004] proposed a common manufacturing application middleware to solve compatibility and synchronization problems between different distributed applications, such as design and manufacturing planning process; but means to realize the design modification in a co-assembly design environment was not discussed in detail. Toshiki and Cutkosky [1998] proposed an agent-based architecture and a set of algorithms to coordinate the actions of different design agents using the theory of Pareto optimality. The agents are reactive and they can track and respond to changes in the state of the design when one designer changes his design, and thus brings about potential conflicts. In each design agent, there is a design process manager that is responsible for recording the design process, and manages rule-based knowledge to coordinate and control the actions of agents. However, the communicating protocol to exchange information between the design agents is simple and limited, so that this architecture is not suitable for the more complex co-assembly design.

From the above review, the previous works have not proposed a sufficiently complete and effective synchronous and asynchronous supportive approach to realize design modification in collaborative assembly design.

2.2 Previous works on evaluation of the tolerance influence on product assemblability

In assembly design and assembly planning, tolerance design is a key issue, which not only ensures that effective function, but also assemblability of the product.

In an ideal assembly design without consideration of tolerance, the relative position and orientation of each part in the assembly can be inferred by the spatial assembly configuration. However, in practice, the actual position and orientation of the part in assembly would deviate from the ideal condition due to the following two factors: Firstly, in the actual manufacturing process which cannot produce the part to the nominal geometric shape and dimension, a tolerance exists and must be given in the design stage. Generally the design tolerance can be categorized into dimensional, positional and form tolerance. They can result in the positional and orientation deviations of different features in one part. Secondly, during the assembly process, in some mating conditions such as peg-hole mating, there exists clearance due to the geometric tolerance of mating features, and the clearance could result in the positional and orientation deviations between the mated features of different parts. So, the above two factors can jointly result in the relative positional and orientation deviations of the features in assembly.

In a given assembly sequence, when parts are assembled into the sub-assembly one-by-one, the positional and orientation deviations are accumulated and propagated from the first part to the last, and the deviations accumulated can result in interference occurring in a later stage of the assembly process. The part geometric shape and dimensions caused by the manufacturing process are stochastic and can be limited within the design tolerance range. The clearance in assembly is decided by the stochastic geometric shape and dimensions of mating features in the assembly process and likewise is also stochastic. Consequently the positional and orientation deviations of part features in assembly are stochastic.

The evaluation of the product assemblability in different assembly sequences is very important. It can help the assembly planner find optimal assembly sequences with good assemblability, and can help the designer identify design problems, hence make proper design modification or redesign during the design stage.

Currently, many research works have already been done on tolerance design and tolerance analysis in assembly, with focus on different areas. Some works ([Lin et al, 1997], [Srikanth et al, 2001], [Yang and Naikan, 2003], [Ngoi and Ong, 1999a], [Ashiagbor et al, 1998]) focused on the allocation of tolerance to different parts, aiming to minimize the manufacturing cost and maintain the assembly function. Some ([Treacy et al, 1991], [Ngoi and Ong, 1999b]) used one-dimensional stack-up analysis to optimize the tolerance allocation based on the assembly requirement. The works considering the tolerance influence on assemblability can be summarized as follows:

Whitney and Gilbert [1993] used a three-dimensional kinematic parameter boundary to define the tolerance zone, and then Monte-Carlo simulation to approximate the tolerance zone into an ellipsoid boundary representation. Similar to the work of Whitney and Gilbert, Lee and Yi [1995a, 1995b, 1997] used kinematic parameters to approximate the tolerance and clearance zone into an ellipsoid, and proposed a statistical method based on Monte-Carlo simulation to calculate the tolerance and clearance propagation. However, the repetitive simulation for assemblability measure is time consuming, and not so suitable for integration into the assembly planning system.

Chase et al. [1996] characterized the geometric feature tolerance in a vector-loop-based assembly tolerance model, and proposed a direct linearization method to analyze the assembly tolerance, which includes geometric feature variations. This method can be used for the estimation of assembly failure in a 2D or 3D assembly. However, the clearance in the assembly was not considered, and the tolerance influence on different assembly sequence was not studied.

Sodhi and Turner [1994] used a constraint optimization technique to determine relative part positions in assemblies containing manufacturing variations. The contact relations are defined as non-interference constraints. The other assembly relations including contacts, attachments, alignments, etc. are sequentially optimized by the generation of objective constraints. The focus is on the sequence of assembly relations in the assembly process. The influence of tolerance on the assembly sequence was not considered.

Park and Lee [1998] proposed a method to calculate the minimum distance between variational features, which is used to decide the contact state between mating parts, and then to determine the assemblability by subdividing the ranges of relative position between variational parts recursively until no subdivided regions exist that can cause interference in the assembly. This method can judge the assemblability between parts. However subdividing the regions recursively needs much computing time when the parts in the assembly increase. They did not consider the clearance influence on assemblability in different assembly sequences.

Desrochers and Riviere [1997] represented the tolerance zone with a matrix approach by defining the tolerance zone with a set of inequalities. However the relationship between different assembly sequences with the different geometric tolerance and clearance accumulation was not discussed. Similarly, Wang et al. [2002] proposed a method to convert the tolerance of mating features into a set of inequalities in a deviation space to represent feature deviation from the nominal shape, but the assemblability in different assembly sequences caused by the tolerance was also not considered.

The above research works have not proposed an effective approach to evaluate the product assemblability in different assembly sequences. There is a need to investigate the evaluation of the product assemblability in different assembly sequences, which considers the influence of both tolerance and clearance.

2.3 Previous works on assembly planning

As an important step during product development, effective assembly planning can achieve an assembly sequence, which is not only feasible, but also optimal by considering the assembly cost or assembly time. Meanwhile, the assembly planning results that indicate the difficulties during assembly process can provide the designer the decision support on improving the product design in the design stage.

The research works in assembly planning can be divided into two main categories: graph-based approach and artificial intelligence (AI-) based approach.

2.3.1 Graph-based approach

The graph-based approach can be further divided into three areas: directed graphs of assembly, AND/OR graphs of subassembly, and connector-based assembly sequence graph. The approach using directed graphs of assembly was first proposed by Bourjault [1984], and developed by other researchers ([De Fazio and Whitney, 1987], [Homen de Mello and Sanderson, 1988], [Baldwin et al, 1991], [Delchambre, 1990], [Laperriere and ELMaraghy, 1992]). This approach can represent all assembly sequences intuitively based on the assembly relationships and precedence constraints. The nodes in the graph represent the set of parts or a subset of parts in each subassembly already assembled at each state of the assembly process. In the approach using AND/OR graphs of subassembly ([Homen de Mello and Sanderson, 1990, 1991], [Henrioud and Bourjault, 1991]), the problem of generating the assembly sequence is transformed into the problem of generating disassembly sequence, through generating all cut-sets of the assembly's graph of connections and checking the feasible decompositions of the cut-sets, to return the AND/OR graph representation of assembly sequence. The concept of connectors was proposed by Gui and Mantyla [1994] as those connectors that provide constraints on the joined components to ensure that they perform the required functions. Based on this concept, Tseng and Kweili [1999] provided a novel means to generate an assembly sequence. In this approach, an assembly product can be decomposed into a set of connector-based assembly elements through a definition and representation scheme, and an assembly sequence generation algorithm, by which a connector-based assembly sequence graph is generated. Yin et al. [2003] also used the fundamental of connector concept to decompose an assembly into a set of connector-based structure (CBS) hierarchy, and generate the assembly plans by merging plans for primitive nodes in the CBS hierarchy.

The graph-based approach theoretically can find global optimal solutions based on the assembly relationship and the precedence constraints, with the objective to reduce the assembly cost. However, it is time consuming, especially when the product includes many parts and components, and the possible assembly sequences can rise exponentially, which would cost much computation time and resource; so it is not suitable for situations where rapid calculation and response speed are required.

2.3.2 AI-based approach

In order to improve the efficiency of assembly planning, some AI-based techniques have been used in the assembly sequence planning recently. Generally the AI-based approaches can be further divided into two main areas: knowledge-based approach and GA-based approach.

Knowledge-based assembly planning approach generally uses relevant rules from the knowledge base and the inference mechanism to get the assembly plan. Rabemanantssoa and Pierre [1996] used an object-oriented database system to translate the design information from IGES and NEUTRAL files, and used relevant rules from the knowledge base and the inference mechanism to get the assembly plan. In this work, automated feature recognition and position plus orientation information are integrated in the knowledge base for part mating, and the possibility of contact and relative mobility for each pair of components is defined from rule-based specifications. Zha et al. [1999] proposed a knowledge-based approach for integrated design and assembly planning. The knowledge base includes the database, static knowledge base and dynamic knowledge base. The knowledge of assembly design and planning was presented by “if-then” production rules, and the assembly sequence planning can be realized through a knowledge base management system and an inference engine. The other works using the knowledge-based system to generate assembly sequence were reported in [Rabemanantssoa and Pierre, 1993a, 1993b].

In a knowledge-based approach, the feasible assembly sequence can be found using knowledge base and inference mechanism; however it is difficult to find the optimal assembly sequence without a suitable searching algorithm, especially when the assembly has many parts and components, and there exists many alternative assembly sequences.

Compared to the knowledge-based approach, the GA-based approach is more attractive in assembly planning. Using this approach, not only the optimal or near-optimal solutions can be found, but high computing efficiency can also be achieved.

Up to now, some works have been done in this area. Dini et al. [1999] proposed a method using GA to generate and evaluate the assembly sequence, and adopted a fitness function considering simultaneously the geometric constraints and some assembly process including the minimization of gripper changes and object orientations, and the possibility of grouping similar assembly operations. Hong and Cho [1999] proposed a GA-based approach to generate the assembly sequence for robotic assembly, and the fitness function is constructed based on the assembly costs that are reflected by the degree of motion instability and assembly direction changes assigned with the different weights. Lazzerini and Marcelloni [2000] used GA to generate and assess the assembly plans. The feasible assembly sequence is based on three criteria: number of orientation changes of the product, number of the gripper replacements, and grouping of similar assembly operations. In this approach, the suitable weights are selected for three criteria to construct the fitness function through experiments, and the good probability that GA converges to a feasible assembly sequence can be achieved. Chen and Liu [2001] proposed an adaptive genetic algorithm (AGA) to find global-optimal or near-global-optimal assembly sequences. In this algorithm, the genetic-operator probabilities are varied according to certain rules, and calculated by the simulation function. Then the calculated genetic-operator probability settings are used to dynamically optimize the AGA search for an optimal assembly sequence. Lit et al. [2001] proposed an original ordering genetic algorithm to plan the assembly sequence. In this approach, precedence constraints are used to ensure that the assembly sequence is valid. In addition, a multi-objective cost function was also proposed in this approach, based on five technical criteria: the number of reorientations, the stability of subsets, the parallelism between operations, and the latest or earliest components to be put in the plan. A multi-criterion decision-aided method is used, whereby the decision maker assigns and adjusts the weights of the criteria until good solutions can be found. Guan et al. [2002] proposed the concept of gene-group to consider the assembly process planning, not merely the assembly sequence planning. A gene-group includes the components to be assembled, tool used to handle the component, assembly direction, and type of assembly operation, to express the information of assembly process. The change times of the assembly tools, assembly directions, and assembly types are used in the fitness function to evaluate the assembly costs. Smith et al. [2002] proposed an enhanced genetic algorithm based on the traditional genetic algorithm. This approach does not choose the next-generation assembly sequence based on the fitness, instead it periodically repopulates with high fitness assembly plans to find optimal or near-optimal assembly plans more reliably and quickly than the traditional approaches.

Some success has been achieved in the above-mentioned GA-based assembly planning works; however, in these works, only the precedence feasibility was considered to ensure the assembly sequence was feasible. The tolerance and clearance influence on product assemblability in different assembly sequences has not been considered. Tolerance and clearance can cause geometric deviations during the assembly process. A different assembly sequence can result in a different propagation and accumulation of geometric deviations, and the assembly sequence is not feasible when the accumulated deviation exceeds the limits. In addition, to deal with a multi-objective optimization problem, these works generally used constant weights to build the fitness function by some form of evolutionary trial. The search direction was fixed, and sometimes they could not find the optimal or near-optimal solution, and other non-dominated solutions. According to the above limitations, more research effort needs to be done in this area to enhance the function of assembly planning. The tolerance and clearance influence on product assemblability should be considered, and more non-dominated solutions be found.

2.3.3 Collaborative assembly planning

As discussed in the section 2.3.2, the existing assembly planning approach mostly focus on single-user assembly planning, by which all the assembly planning tasks are carried out by one planner in one location. Although there are many research works on collaborative assembly design and modeling, there are few works on collaborative assembly planning. For a complex assembly product with many parts, e.g., automobile, due to the function and assembly condition of different part being usually different, in practice, those parts are usually divided into different subassemblies and assembled in different workshops with different facilities. For each subassembly, the planner should know well the assembly requirement and assembly condition for the parts in the subassembly which he is responsible for, and he should also know well the assembly condition and facilities of the workshops where the assembly task will be done. Therefore, it is difficult for only one planner to carry out all the assembly planning tasks because it is not easy for him to grasp the large amount of information.

In recent years, with the Internet and web, several works on collaborative assembly planning have been reported.

Wang et al. [2004] proposed a web-based collaborative assembly planning system that enables several experts to do disassembly planning collaboratively. During this disassembly process, each planner sends the request about the distance and direction of the assigned parts to disassemble them from the product one at a time, through an interference check using a disassembly matrix to ensure that the disassembly process is feasible. Using this method, the parts can be disassembled from the product by several planners synchronously and several disassembly sequences are generated. Finally, the disassembly sequences are reversed and merged into one assembly sequence. This collaborative assembly planning approach depends much on the judgement and experience of the planners during the part disassembly process, and the final assembly sequence can only ensure a feasible sequence, not necessarily an optimal sequence, without considering the objective optimization, such as number of tool changes, etc.

Li et al. [2003] proposed an approach to decompose the assembly planning task to different planners in collaborative assembly planning environment to reduce the complexity of the assembly planning, enabling each planner to carry out the assembly planning task on the assigned parts, and through conflict detection, negotiation and control right alternation to finish the assembly planning of the whole product. This approach can effectively resolve the conflict during assembly through conflict detection, negotiation and control right alternation. However, during the assembly process, all the efforts are made to achieve a feasible sequence, but not on optimizing the assembly sequence to save cost.

Hirai and Nagata [1994] proposed an approach that used the directed graph to represent the connective relation among the parts. This connective relation and the geometric restriction on connection of parts are represented as knowledge, and they are given to each part, which behaves as agents. The assembly plan can be generated as result of autonomous behavior of these agents. By handling of each manipulator as an agent, subtasks can be allocated to different manipulators through the negotiation of these agents. Using this approach for assembly planning, because each agent behaves autonomously, this system can likely trigger unnecessary actions and cause deadlocks, especially for more complicated assembly planning problems. In addition, using the connective relation among the parts and the geometric restriction on connection of parts as the knowledge base, only the feasible assembly sequence can be derived. However, optimal sequence can not be concluded without an optimization method.

Dong et al. [2005] proposed an assembly representation model using the connection semantics based assembly tree, which includes two types of nodes: parts and connectors. By geometric reasoning and knowledge-based reasoning according to some heuristic rules, an assembly plan can be derived from a knowledge base that includes stored assembly plans. In collaborative planning, the planning tasks are assigned to different planners to carry out the assembly planning using the above reasoning approach, respectively. Actually this assembly planning approach belongs to the knowledge-based assembly planning approach, and it can provide the planner with a feasible assembly planning solution. However an optimal solution is not easy to achieve.

From the above, the existing works related to collaborative assembly planning adopt different approaches to realize the assembly planning, enabling different planners to carry out different assembly planning tasks collaboratively, to achieve a feasible assembly plan through negotiation, rule-based reasoning, etc. However, these works can only conclude a precedence feasible assembly plan because they only consider the assembly relationship and geometric restriction among the parts in the assembly. Without an optimization mechanism, an optimal assembly plan considering the detailed assembly condition and manufacturing facilities are not considered by the planners during assembly planning.

2.4 Research objectives

The objectives of this research are to investigate and develop effective approaches to realize the collaborative assembly design modification and assembly planning, focusing on the following four areas:

1) Development of a methodology to support effective design modification in collaborative assembly design. In order to realize effective design modification in collaborative assembly design, the following need to be developed:

- A collaborative assembly representation approach for design modification in collaborative assembly design.
- A design modification propagation control mechanism to effectively propagate the design modification of one part to the associated parts designed by others.
- A suitable system framework for the design modification in collaborative assembly design.

2) Development of an effective GA-based assembly planning approach. In order to enhance the efficiency and function of the GA-based assembly planning, the following issues need to be addressed:

- To develop an effective approach to conclude the influence of tolerance and clearance on product assemblability in different assembly sequences. This product assemblability is to be used as a constraint during assembly planning to evaluate the feasibility of different assembly sequences.
- To propose a method to build different fitness functions. A fuzzy weight distribution algorithm is proposed for assigning fuzzy weights to different objectives so that the search directions toward the Pareto frontier can be expanded and more non-dominated solutions can be found.

- To develop the corresponding evolution mechanism according to the proposed fitness functions.

3) To provide redesign guidelines to help the designer to consider appropriate design modifications or redesign in the design stage. This involves identification of potential design problems which can be found through the evaluation of the assembly planning results. The redesign guidelines should be derived from the following aspects:

- Redesign suggestion from the assemblability evaluation
- Redesign suggestion from the number of assembly orientation changes
- Redesign suggestion from the number of assembly tool changes
- Redesign suggestion from the number of assembly operation type changes

4) Development of a collaborative assembly planning approach based on the proposed GA-based assembly planning approach, by which several geographically dispersed planners can carry out the assembly planning collaboratively, to speed up the assembly planning process, and the assembly plan can be not only feasible, but also optimal or near- optimal. The main works focus on the following areas:

- To develop a suitable system framework for collaborative assembly planning
- To build the subassembly task assignment mechanism and the feasibility check algorithm
- To realize the proposed GA-based assembly planning approach in collaborative assembly planning environment

Chapter 3 Design Modification in a Collaborative Assembly Design Environment

This chapter discusses the design modification issues in a collaborative assembly (co-assembly) design environment, which enables multiple geographically dispersed designers to design and assemble parts collaboratively through the Internet. An assembly representation model, a feature-based hierarchical co-assembly representation, is proposed and a new definition of the assembly feature is given to resolve the co-assembly design issues. In order to realize the design modification, a design modification propagation control mechanism is proposed. A system framework that is suitable for realizing the design modification is also proposed and developed.

Section 3.1 proposes a feature-based hierarchical representation model for co-assembly design; Section 3.2 discusses the functions of the proposed co-assembly representation model; Section 3.3 proposes and investigates a design modification propagation control mechanism; Section 3.4 proposes the system framework; a case study is presented in Section 3.5 to illustrate the detailed design modification propagation control mechanism; Section 3.6 is the summary of this chapter.

3.1 An assembly representation model for collaborative design

In a co-assembly design process, how to represent the assembly is very important to realize the design modification and communication between different designers geographically dispersed. The representation approach is required to represent not only the assembly relationship between features of different parts, but also the network-based working relationship between different designers. Based on the above requirements, a feature-based hierarchical co-assembly representation model is proposed, as discussed in section 3.1.1.

3.1.1 Feature-based hierarchical co-assembly representation

A feature-based hierarchical co-assembly representation model is proposed, as shown in Figure 3.1.

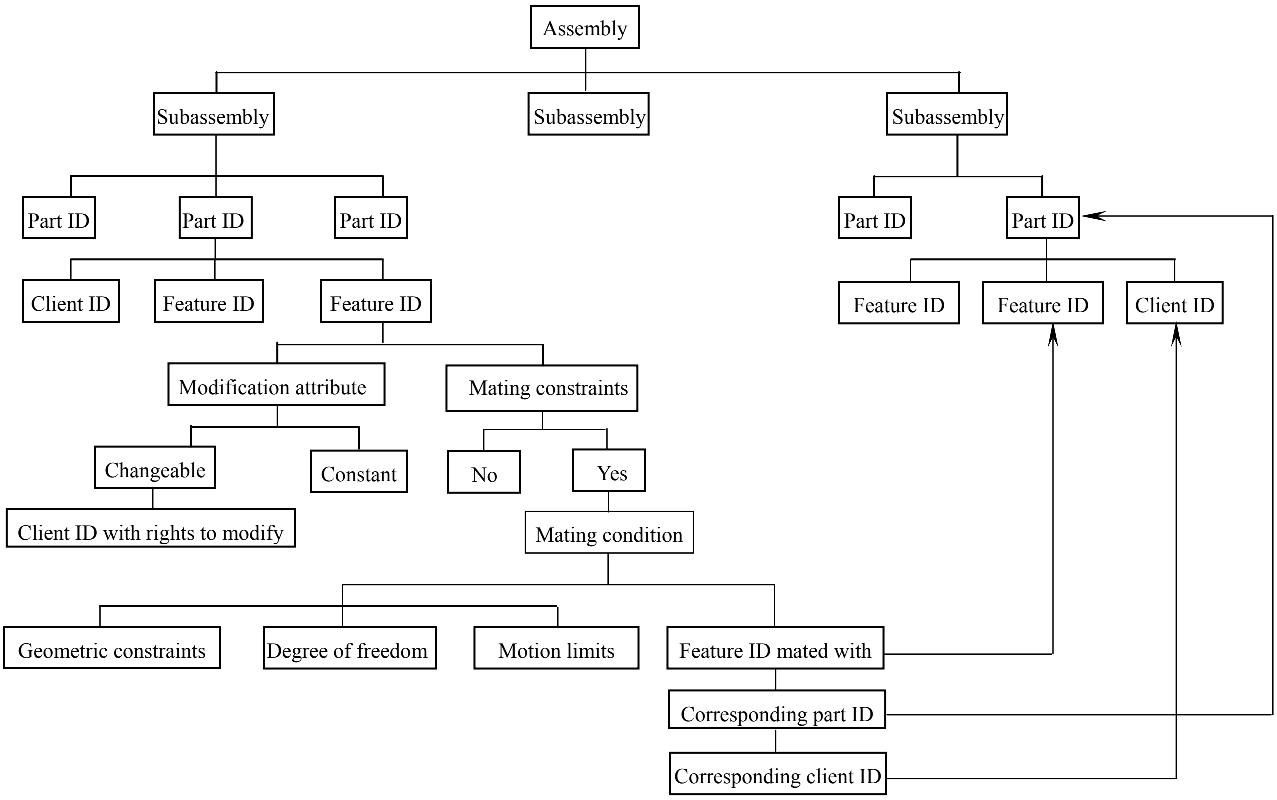


Figure 3.1 Feature-based hierarchical co-assembly representation

This hierarchical data structure organizes an assembly as a compound of sub-assemblies; and the sub-assemblies are composed of several parts. The parts can be divided into a number of form features that are composed of boundary entities using Boolean algorithms. In addition, a part has one element “client ID” which indicates the designer for this part. In the following classification, each form feature has two basic elements: “modification attribute” and “mating constraints”. The modification attribute includes two states: “changeable” and “constant”. The changeable state means the geometrical shape of this form feature or its position in the assembly can be changed after the assembly design finished, and the constant state means the both above must be kept the same. Typically, the constant attribute is often used in some critical and standard parts in the assembly. The “Client ID with rights to modify” indicates the designers with rights to modify the feature. The other basic element, i.e., “mating constraints”, has two attributes: “no” and “yes”. If “no”, this form feature does not have any assembly relationship with other form features; otherwise, “yes” means the feature has the assembly relationships. If a feature has assembly relationships, the “mating condition” of this feature further includes sub-elements: “feature ID mated with”, “geometric constraints”, “degrees of freedom” and “motion limits”. The “feature ID mated with” points to the form feature mated with it, and through this form feature, the corresponding part ID and client ID can be searched and retrieved.

This hierarchical data structure not only represents the longitudinal “part-of” relationships, but also the latitudinal “mating” relationships between different form features belonging to different parts which are designed by the different designers geographically dispersed. In addition, by assigning the modification attribute and modification rights of each form feature, the design modification propagation routes can be built up when the design of a form feature is modified in the co-assembly design environment.

3.1.2 A definition of assembly feature in collaborative design

From the above assembly representation model, a new definition of the assembly feature is extracted. The defined assembly feature includes not only the relationship between two form features, but also the relationship between two different designers in different geographical locations. The structure of this assembly feature is shown in Figure 3.2. Only the form feature that has the mating relationship with others can be used to combine into the assembly feature.

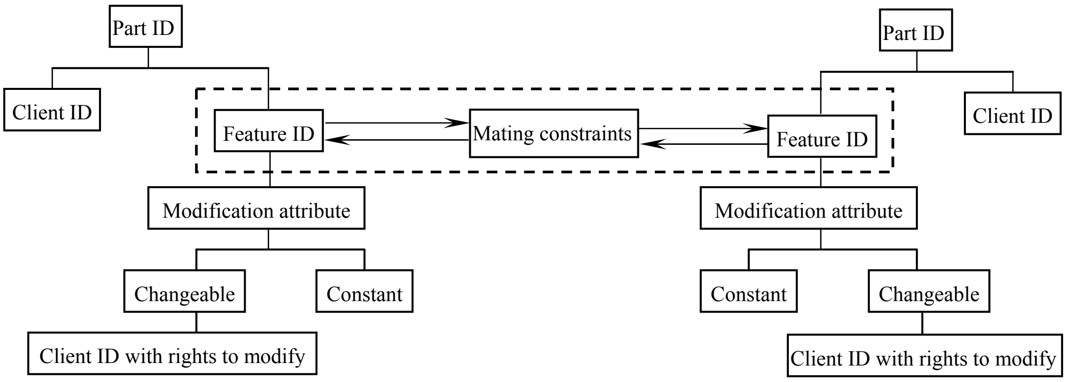


Figure 3.2 Structure of assembly features

In our new definition of the assembly feature, the assembly feature is divided into two properties — internal assembly property and external assembly property.

Definition 1: Internal assembly property

The internal assembly property lies within the dashed rectangle (in Figure 3.2), and it represents the assembly relationships between two form features through mating constraints, which often include geometric constraints, degree of freedom and the motion limits.

Definition 2: External assembly property

The external assembly property lies outside the dashed rectangle, which includes the modification attribute, corresponding part ID and client ID of the form features, as shown in Figure 3.2.

Basically, the internal assembly property is the same as the traditional assembly feature, and its main function is to define the assembly relationships between form features. However, the external assembly property defines the other two important factors in the co-assembly design environment:

(1) Through corresponding part ID and client ID, the relationships between two form features can be used to build up the relationships between two geographically dispersed designers.

(2) The modification attribute of form features helps to decide the design modification propagation routes in the co-assembly design environment.

Therefore, this new assembly feature represents not only the specific assembly information, but also the working relationship between designers in a co-assembly design environment. The function of this assembly representation model is described in Section 3.2.

3.2 Functions of the co-assembly representation model

In this section, an example will be given to illustrate the function of the proposed co-assembly representation model for design modification in the co-assembly design process.

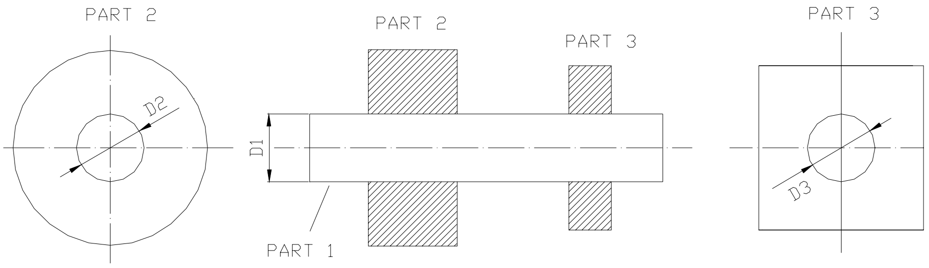


Figure 3.3 Assembly consisting of three parts

As shown in Figure 3.3, a simple assembly consists of three parts – Part 1, Part 2 and Part 3, and each part is to be designed by different designer geographical dispersed, i.e., Client 1, Client 2 and Client 3, respectively. The cylinder feature of Part 1 has the mating relationship with the hole feature of Part 2 and Part 3, respectively. Hence, there are two assembly features in this assembly: one between Part 1 and Part 2, and the other between Part 1 and Part 3. The designer of Part 1 sets the modification attribute of the cylinder feature of Part 1 as “changeable”. The designer of Part 2 also sets the modification attribute of the hole feature of Part 2 as “changeable”. The designer of Part 3 sets the modification attribute of the hole feature of Part 3 as “constant”. These designers do not assign the modification rights to each other. The assembly feature between Part 1 and Part 2 can be represented in Figure 3.4, and the assembly feature between Part 1 and Part 3 can be represented in Figure 3.5.

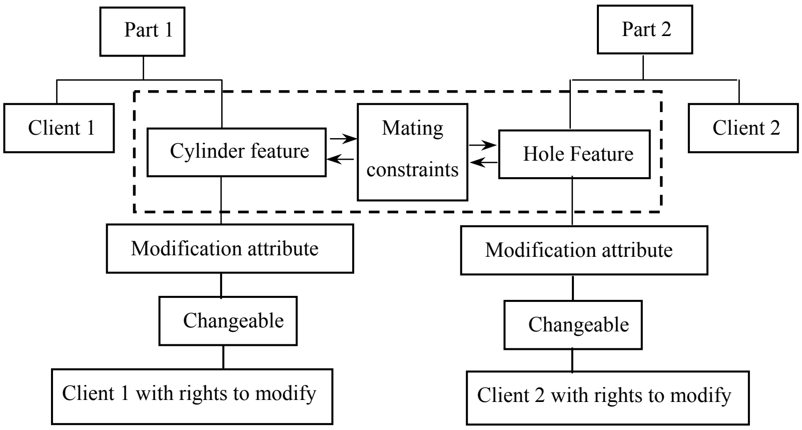


Figure 3.4 Assembly feature between Part1 and Part2

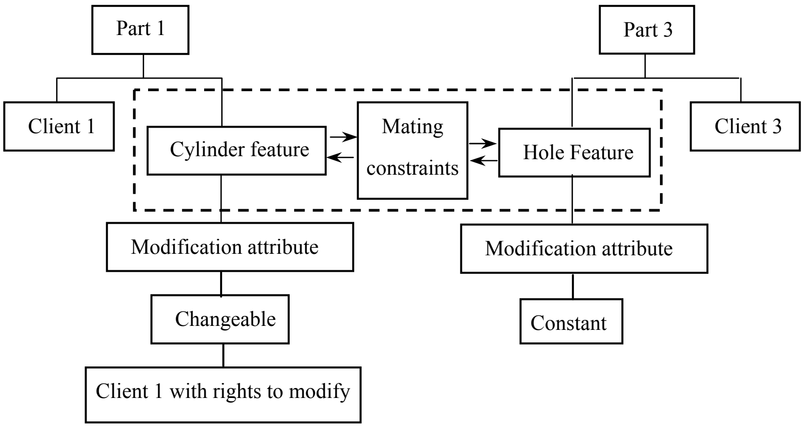


Figure 3.5 Assembly feature between Part1 and Part3 (Condition1)

In the co-assembly design process, after the assembly modeling of these three parts is completed, if the designer in Client 2 modifies the design of the hole feature of Part 2 by increasing the diameter of the hole from *D*2 to *D*2', through the assembly information and the working relationship defined in the assembly feature between Part 1 and Part 2 (in Figure 3.4), this change should be first propagated to the cylinder feature of Part 1, and the diameter of the cylinder *D*1 should be increased to *D*1'. Then, through the assembly information and the working relationship defined in the assembly feature between Part 1 and Part 3 (in Figure 3.5), the modification of the cylinder feature should be propagated to the hole feature of Part 3, and its diameter should also be increased. However, since the modification attribute of this hole feature is assigned “constant”, the diameter *D*3 must maintain constant. Finally, the modification in Part 2 and the two assembly features defined jointly decide the design modification as follows:

(1) The designer in Client 1 should modify Part 1 into a stepped cylinder with the diameter *D*1’ and *D*1.

(2) The designer in Client 3 should keep Part 3 unchanged.

The design modification results are illustrated in Figure 3.6.

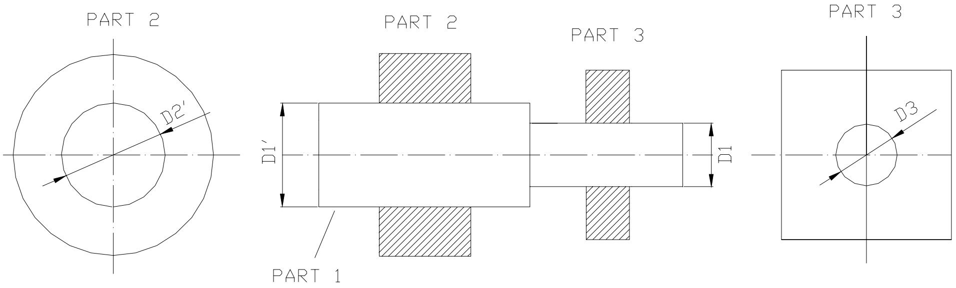


Figure 3.6 The design modification results (1)

Another assembly condition defined by designers is that the assembly feature between Part 1 and Part 2 keeps same as that shown in Figure 3.4, but the assembly feature between Part 1 and Part 3 is changed, in which the modification attribute of the hole feature in Part 3 is set “changeable” by the designer in Client 3, as shown in Figure 3.7.

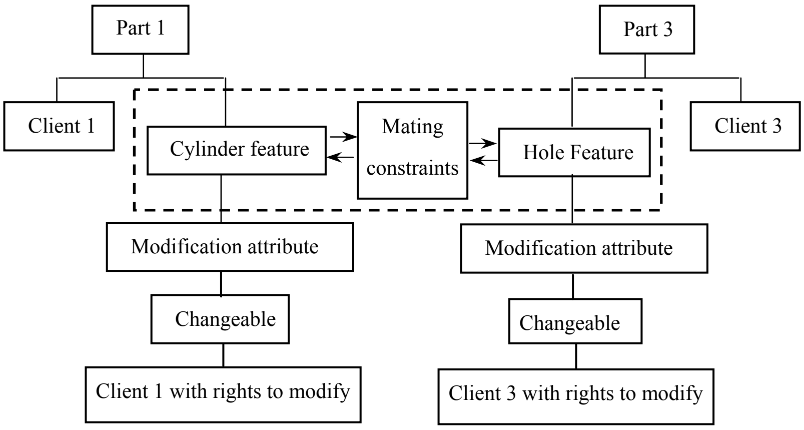


Figure 3.7 Assembly feature between Part1 and Part3 (Condition 2)

Then the modification in Part 2 and the two defined assembly features should jointly decide two design modification schemes. The first one is the same as that shown in Figure 3.6:

(1) The designer in Client1 should modify Part1 into a stepped cylinder with the diameter *D*1' and *D*1.

(2) The designer in Client 3 should keep Part 3 unchanged.

The second scheme is shown in Figure 3.8:

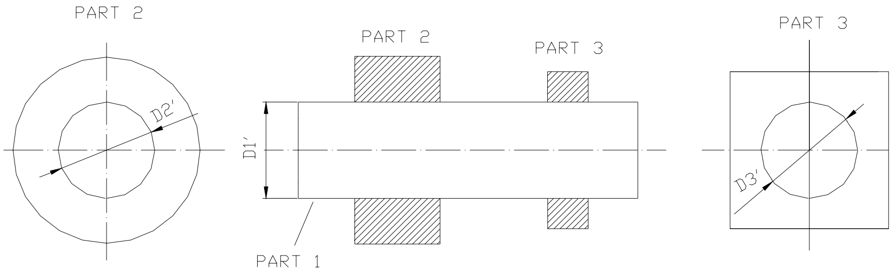


Figure 3.8 The design modification results (2)

(1) The designer in Client 1 should modify Part 1 into a cylinder with the increased diameter *D*1'.

(2) The designer in Client 3 should increase the hole diameter of Part 3 to *D*3'.

From this example, it can be seen that through different definitions of assembly features including internal and external assembly properties, the design modification of the form feature of one part can trigger different change propagation routes in the co-assembly design, and we can get the different design modification results of the other parts designed by other designers geographically dispersed.

3.3 Design modification propagation control mechanism

In this section, a design modification propagation control mechanism is proposed to realize the design modification in the co-assembly design process.

3.3.1 XML representation

The Extensible Markup Language (XML) is a simple and very flexible text format derived from SGML [ISO 8879, 1986]. Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the web and elsewhere [http://www.w3.org/xml].

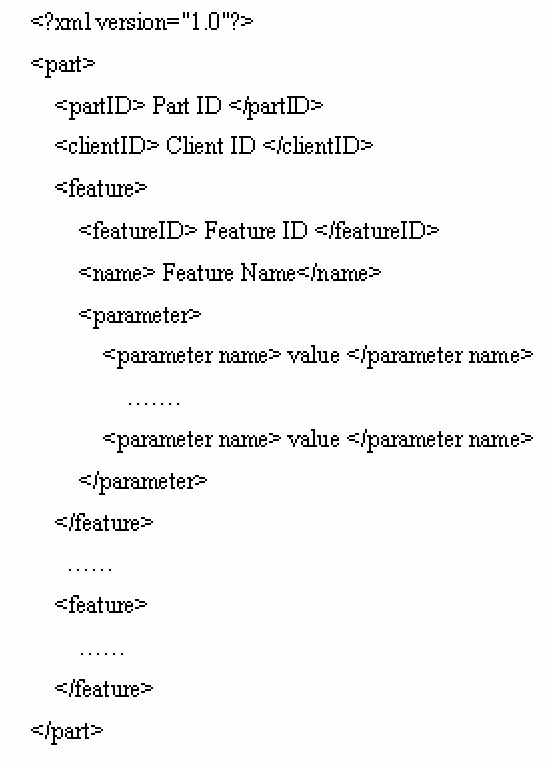
Because of its very flexible data format, it is suitable to embed and transfer various kinds of information via the XML across the Internet. Meanwhile, because the XML format can be defined by meaningful tags to applications, documents represented in XML with an appropriate tagging scheme can be semantically processed by extracting and combining relevant information from a number of XML documents [Pahng et al, 1999].

3.3.2 Using XML file to exchange information

In order to realize the design modification propagation control, two kinds of XML file formats have been adopted to embed different information. Based on the XML files, the design parameters and assembly information of features can be exchanged during the co-assembly design process, as presented below.

XML format 1 (List 1): For defining the design parameters of each feature

List 1 is an XML format for defining the design parameters of each feature. When a designer input the parameters of each feature in a feature-based design working model in his client, the XML file defining the parameters can be written through the XML writer in the client.

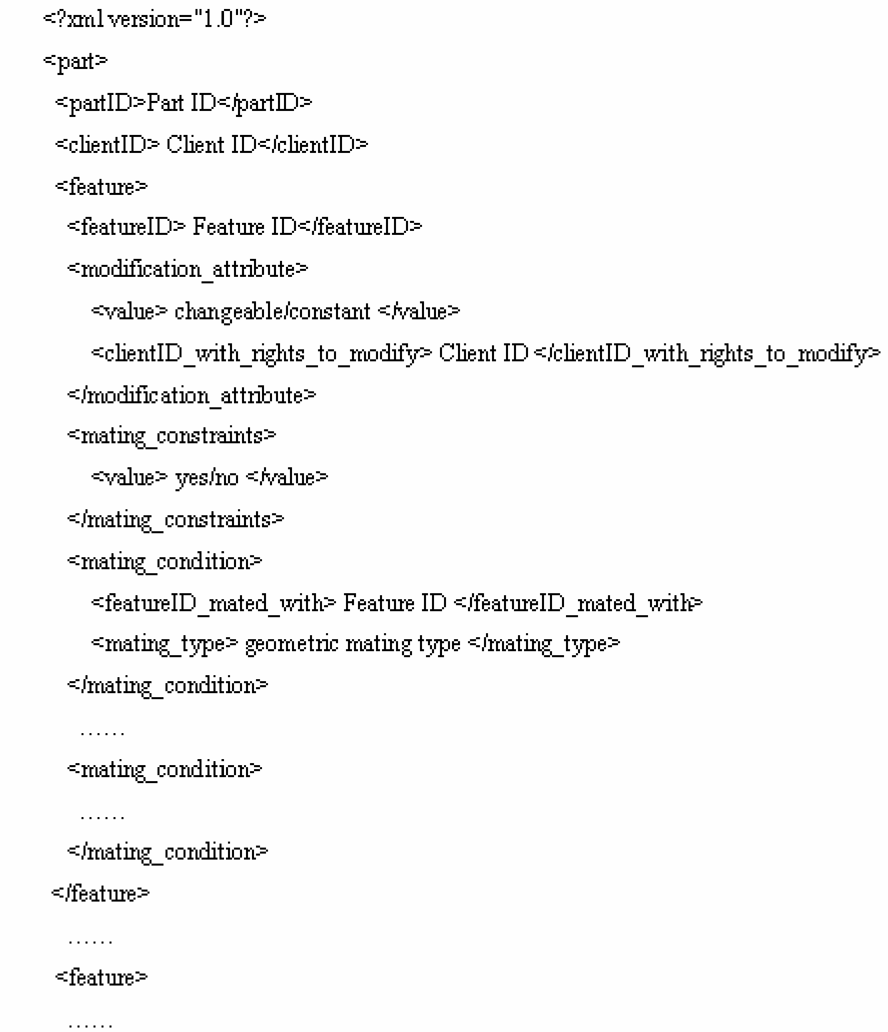


```
<?xml version="1.0"?>
<part>
   <partID> Part ID </partID>
   <clientID> Client ID </clientID>
   <feature>
      <featureID> Feature ID </featureID>
      <name> Feature Name</name>
      <parameter>
         <parameter name> value </parameter name>
            .......
         <parameter name> value </parameter name>
      </parameter>
   </feature>
    ......
   <feature>
      ......
   </feature>
</part>
```

List 1. XML format defining design parameters of each feature in a Part

XML format 2 (List 2): For defining the assembly information of each feature

In order to exchange the assembly information using an XML file, we define another XML format (in List 2) to embed the assembly information of each feature.



```
<?xml version="1.0"?>
<part>
  <partID>Part ID</partID>
  <clientID> Client ID</clientID>
  <feature>
    <featureID> Feature ID</featureID>
    <modification_attribute>
        <value> changeable/constant </value>
        <clientID_with_rights_to_modify> Client ID </clientID_with_rights_to_modify>
    </modification_attribute>
    <mating_constraints>
        <value> yes/no </value>
    </mating_constraints>
    <mating_condition>
        <featureID_mated_with> Feature ID </featureID_mated_with>
        <mating_type> geometric mating type </mating_type>
    </mating_condition>
      ……
    <mating_condition>
      ……
    </mating_condition>
  </feature>
    ……
  <feature>
    ……
  </feature>
</part>
```

List 2. XML format defining the assembly information of each feature in a Part

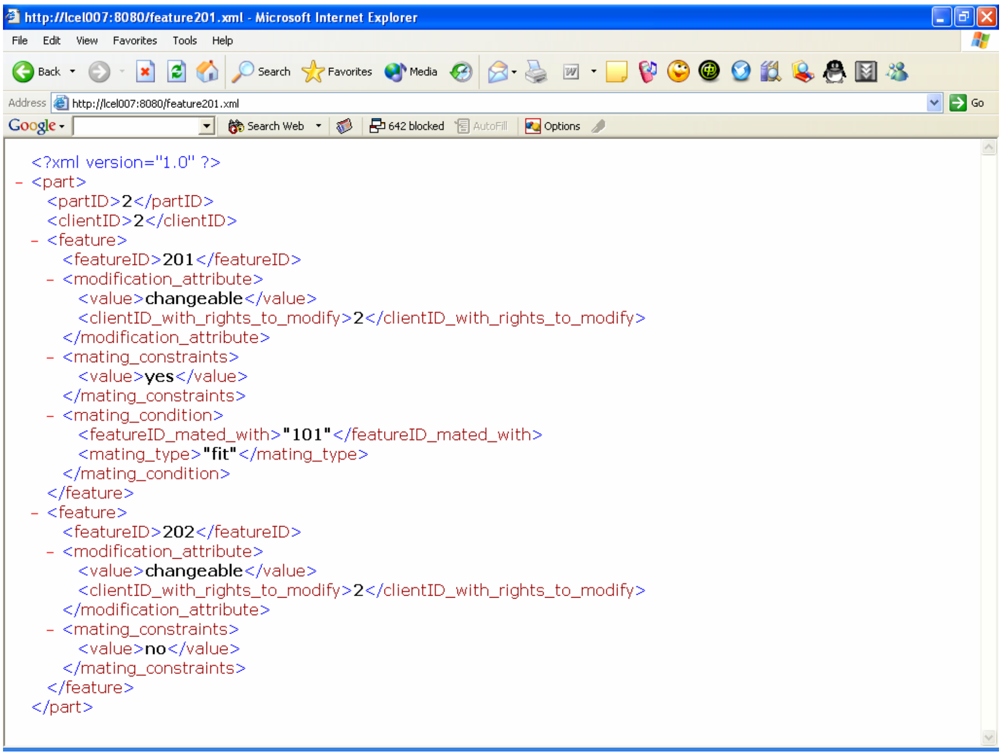
This format is defined according to the feature-based hierarchical co-assembly representation model proposed in Section 3.1. The XML files defining each feature and the assembly information can be written through the XML writer in the client when assembly modeling is completed.

This XML format is defined based on one part. Inside the node <part>, there are sub-nodes that include <partID>, <clientID> and <feature>. Inside each node <feature>, there are four kinds of sub-nodes, which include <featureID>, <modification_attribute>, <mating_constraints> and <mating_condition>. The node <modification_attribute> has the sub-nodes <value> and <clientID_with_rights_to_modify>. The node <mating_constraints> has the sub-node <value>. The node <mating_condition> has the sub-node <featureID_mated_with> and <mating_type>.

The values of the node <featureID>, <partID>, <value> and <clientID_with_rights_to_modify> in <modification_attribute> are decided by the designer in the client during the design process of the feature. However, the values of the node <value> in <mating_constraints>, <featureID_mated_with> and <mating_type> in <mating_condition> are decided in the client during the assembly modeling process.

3.3.3 XML files parsing process

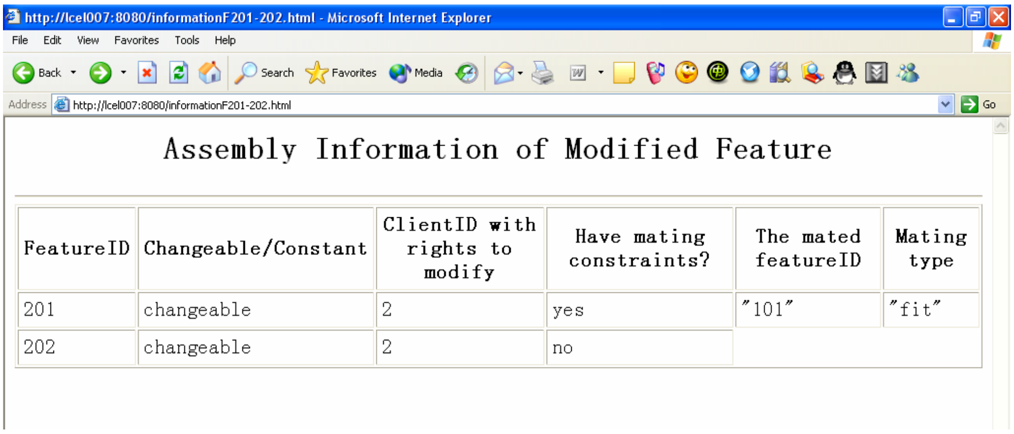
In this section, the assembly example shown in Figure 3.3 is used to illustrate the XML files parsing process when design modification occurred in co-assembly design process. Figure 3.9 shows the XML file defining the assembly information of each feature in Part 2.



```
  <?xml version="1.0" ?>
- <part>
    <partID>2</partID>
    <clientID>2</clientID>
  - <feature>
      <featureID>201</featureID>
    - <modification_attribute>
        <value>changeable</value>
        <clientID_with_rights_to_modify>2</clientID_with_rights_to_modify>
      </modification_attribute>
    - <mating_constraints>
        <value>yes</value>
      </mating_constraints>
    - <mating_condition>
        <featureID_mated_with>"101"</featureID_mated_with>
        <mating_type>"fit"</mating_type>
      </mating_condition>
    </feature>
  - <feature>
      <featureID>202</featureID>
    - <modification_attribute>
        <value>changeable</value>
        <clientID_with_rights_to_modify>2</clientID_with_rights_to_modify>
      </modification_attribute>
    - <mating_constraints>
        <value>no</value>
      </mating_constraints>
    </feature>
  </part>
```

Figure 3.9 XML file defining the assembly information of each feature in Part 2

When the feature defined in the above XML file- <featureID> “201” (hole feature) is modified, through the XML parser implemented we can extract the value of node <value> in the parent node <mating_constraints>, since it is “yes”, it is an assembly feature with assembly relationship with others. Then we further extract the value of the node <feature_ID_mated_with>, which is the feature that has the assembly relationship with the modified feature and will probably be affected by the design modification. Otherwise, if the value of the node <value> in the parent node <mating_constraints> is “no”, e.g. <featureID> “202”, then it is a feature without assembly relationship, and the modification of it cannot affect other features. This process can be executed by the XML parser when the designer sends his XSL file which defined the parsing requirement to the parser. The parsing result- a HTML file including the modified feature information and the mated feature information can be generated automatically when the parsing process is finished. Figure 3.10 shows the parsing result of XML file in Figure 3.9 when <featureID> “201” and “202” are modified, and the result is displayed as a readable web page.



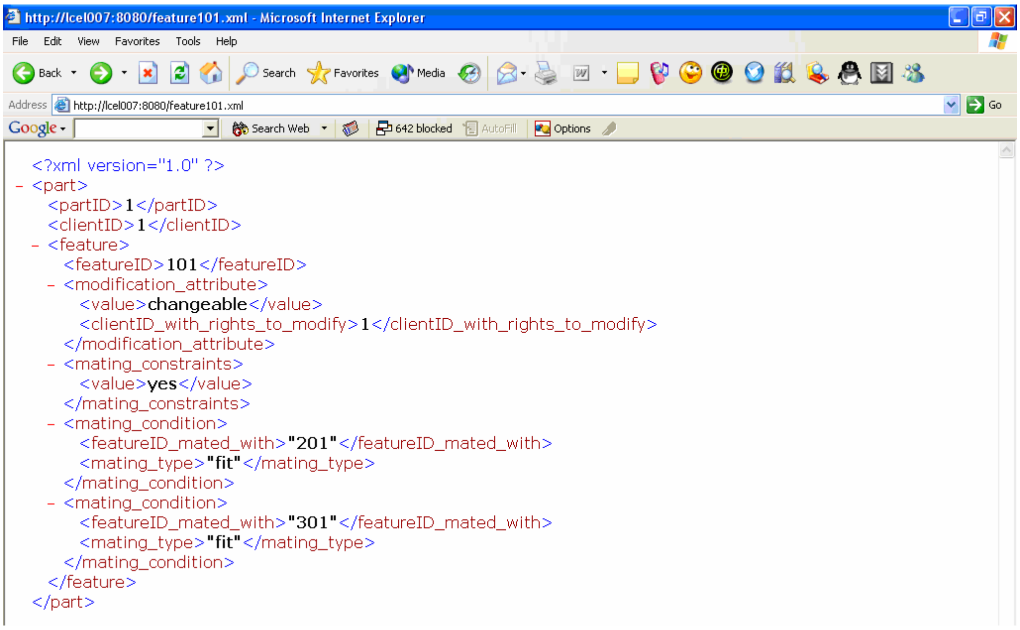
Assembly Information of Modified Feature

FeatureID	Changeable/Constant	ClientID with rights to modify	Have mating constraints?	The mated featureID	Mating type
201	changeable	2	yes	"101"	"fit"
202	changeable	2	no		

Figure 3.10 Parsing result of XML file (in Figure 3.9) when <featureID> “201” and “202” are modified

From the XML file (shown in Figure 3.9), once we extract the value of <featureID_mated_with>, we can get the “The mated featureID” of the modified <featureID> “201”, which is shown in the web page (Figure 3.10). Therefore, we can further search the corresponding XML files defining “The mated featureID”, and parse these files in the same way. In this example, the feature with the <featureID> “101” is affected by the modification of <featureID> “201”. The XML file defining the <featureID> “101” is then further parsed and the other corresponding XML files are also appropriately searched.

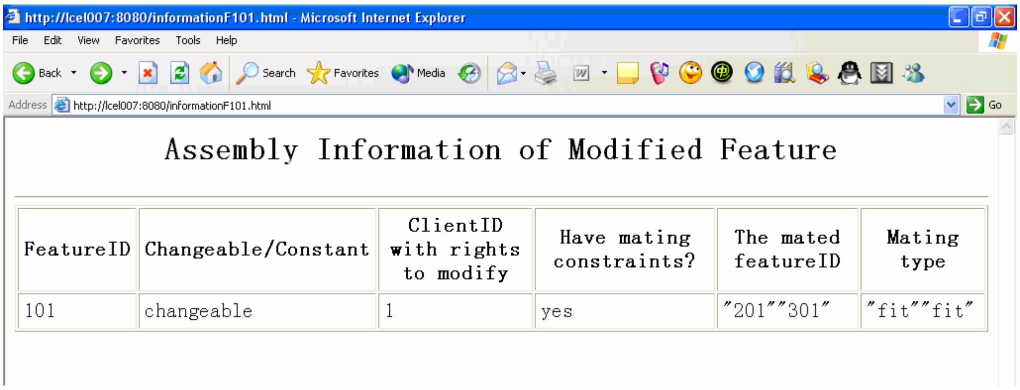
The XML file defining the assembly information of <featureID> “101” is shown in Figure 3.11. This file defines the assembly information of each feature (only one feature) in Part 1 (Figure 3.3). Because the <value> in <modification_attribute> is "changeable", the <featureID> "101" would be affected by the modification, this XML file should be parsed in the same way. (On the contrary, if <value> in <modification_attribute> is "constant", this feature cannot be modified. This means, we do not need to parse this file further, and the design modification cannot propagate through it to others.) The client who designs the <featureID> "101" can check the HTML file that is embeded with the updated design parameters of <featureID> "201". The designer can then use the updated information of <featureID> "201" to modify the affected <featureID> "101". Using the same parsing process, the modification of <featureID> "101" can be propagated to <featureID> "301" (hole feature) in Part 3 (Figure 3.3) based on the assembly information defined in the XML file (Figure 3.11).



```
  <?xml version="1.0" ?>
- <part>
    <partID>1</partID>
    <clientID>1</clientID>
  - <feature>
      <featureID>101</featureID>
    - <modification_attribute>
        <value>changeable</value>
        <clientID_with_rights_to_modify>1</clientID_with_rights_to_modify>
      </modification_attribute>
    - <mating_constraints>
        <value>yes</value>
      </mating_constraints>
    - <mating_condition>
        <featureID_mated_with>"201"</featureID_mated_with>
        <mating_type>"fit"</mating_type>
      </mating_condition>
    - <mating_condition>
        <featureID_mated_with>"301"</featureID_mated_with>
        <mating_type>"fit"</mating_type>
      </mating_condition>
    </feature>
  </part>
```

Figure 3.11 XML file defining the assembly information of each feature in Part 1

The parsing result of the XML file defining the <featureID> "101" (Figure 3.11) when <featureID> "101" is modified is displayed in Figure 3.12.



Assembly Information of Modified Feature

FeatureID	Changeable/Constant	ClientID with rights to modify	Have mating constraints?	The mated featureID	Mating type
101	changeable	1	yes	"201""301"	"fit""fit"

Figure 3.12 Parsing result of the XML file in Figure 3.11

The flowchart of the whole parsing process is given in Figure 3.13. Through this parsing process, a design modification propagation control mechanism is built in the co-assembly design.

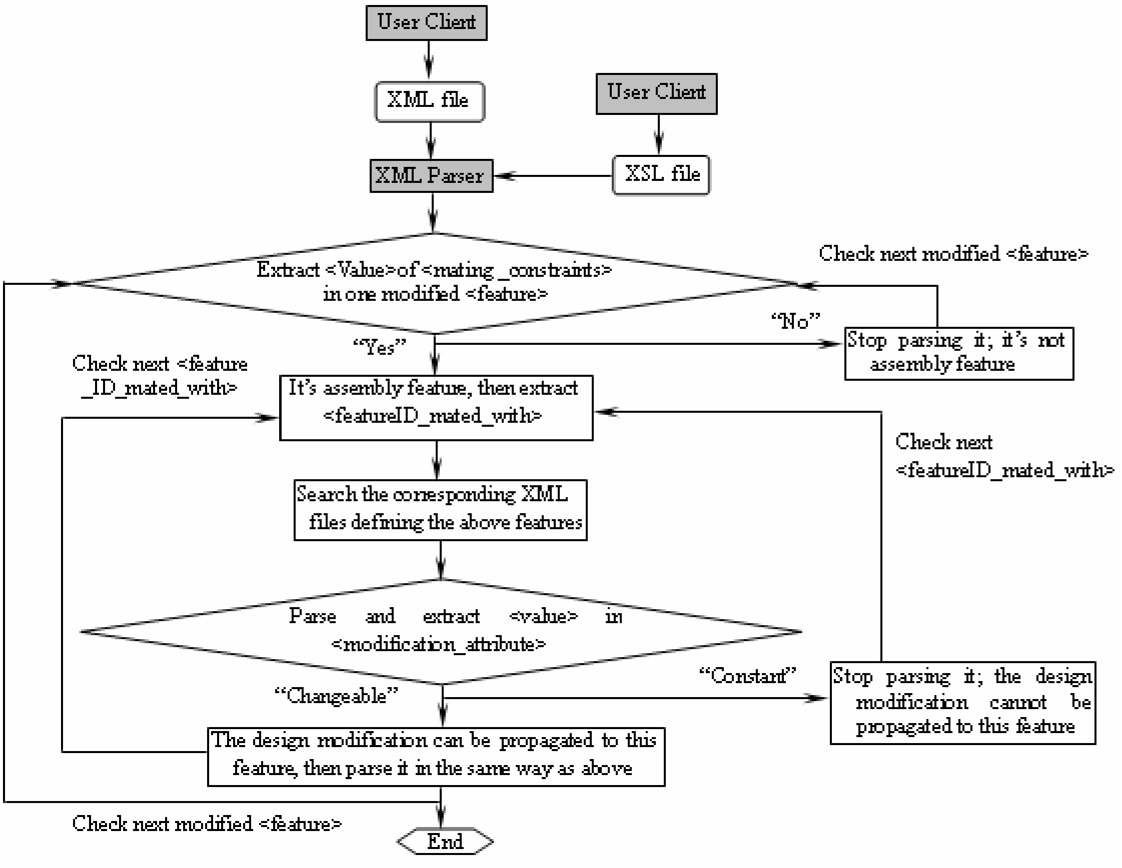


Figure 3.13 Flowchart of whole XML files parsing process

3.4 System implementation

As described in this chapter, a prototype system has been developed based on JDK 1.4 and Open CASCADE 4.0, and its system framework is shown in Figure 3.14. It is a three-tier client-server structure that includes modeling server, Apache web server and design clients.

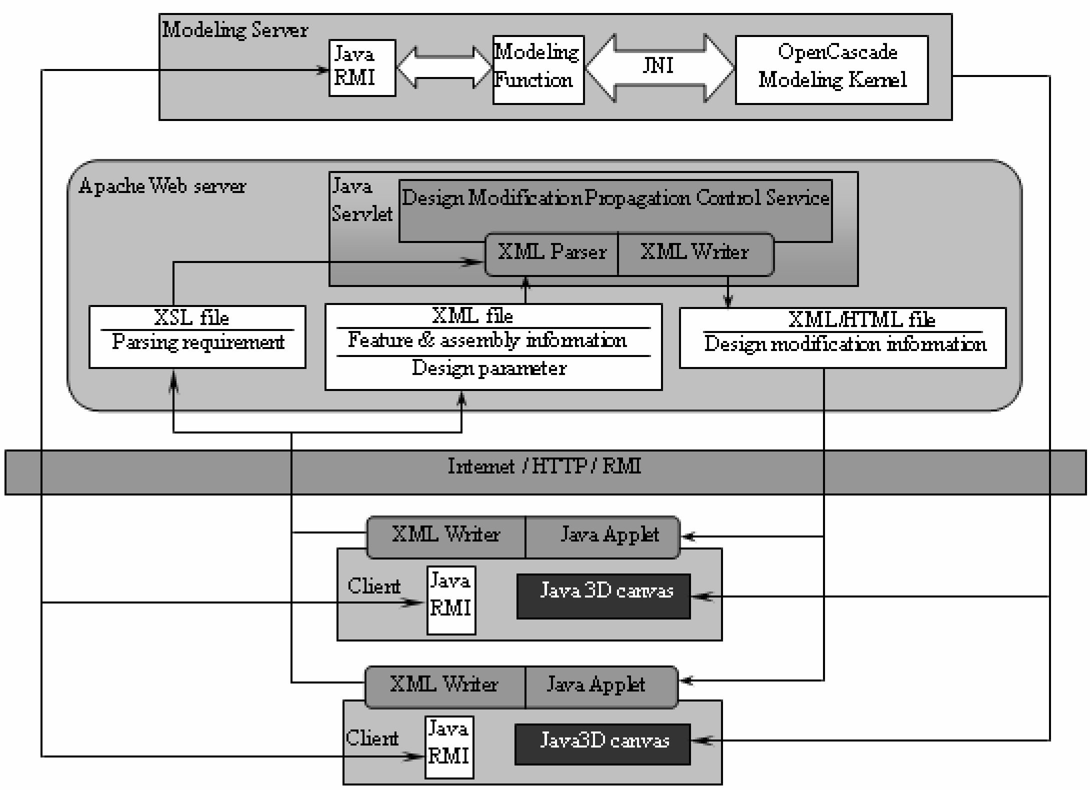


Figure 3.14 The proposed system framework

The modeling server executes the modeling function to realize the part and assembly modeling. It communicates with the design clients through Java RMI (Remote Method Invocation). When the design client sends the designing order to the modeling server through Java RMI, the modeling server finishes the modeling by calling OpenCascade modeling kernel [http://www.opencascade.com/products/] through JNI (Java Native Interface). The completed geometric model can be polygonized using OpenCascade kernel and maintains the pointers to the corresponding entities in the geometric model in the server, and this polygonized model will be transferred to the client for visualization and manipulation in the Java 3D environment.

The Design Modification propagation Control Service is a Java Servlet running in the Apache web server. It communicates with the design client through Java Applet downloaded in the client. Through Java Applet, the design client can submit the information to the Java Servlet and receive the processing results from the Servlet.

In the design client, when the designer inputs the design parameters of the feature in a feature-based design model, these parameters can be written into an XML file formatted as in List 1 through the XML writer in the client. In addition, during assembly modeling, the assembly information of each feature can be retrieved, and the XML files defining the feature and the assembly information formatted as in List 2 can also be written through the XML writer in the client when the assembly modeling is completed. When a feature is modified in one client, these two kinds of XML files and the XSL file defining the parsing requirement by the designer are submitted to the Apache web server for design modification propagation control service. The result, the XML/HTML files embedded with the design modification information will be published and browsed by the client due to the features designed by it are affected by the modification of other features. The detailed design modification propagation control mechanism has been illustrated previously in Section 3.3.

3.5 Case study

In this section, a simplified gearbox assembly is used to demonstrate the design modification process in a co-assembly design environment.

Figure 3.15 shows the simplified gearbox assembly displayed in one design client- Client 1. In this assembly, Part 1 to Part 10 are designed respectively in ten clients- Client 1 to Client 10. The design of the assembly can be finished by any design client through the modeling server when the design of those ten parts are finished. The assembly can also be displayed in other design clients.

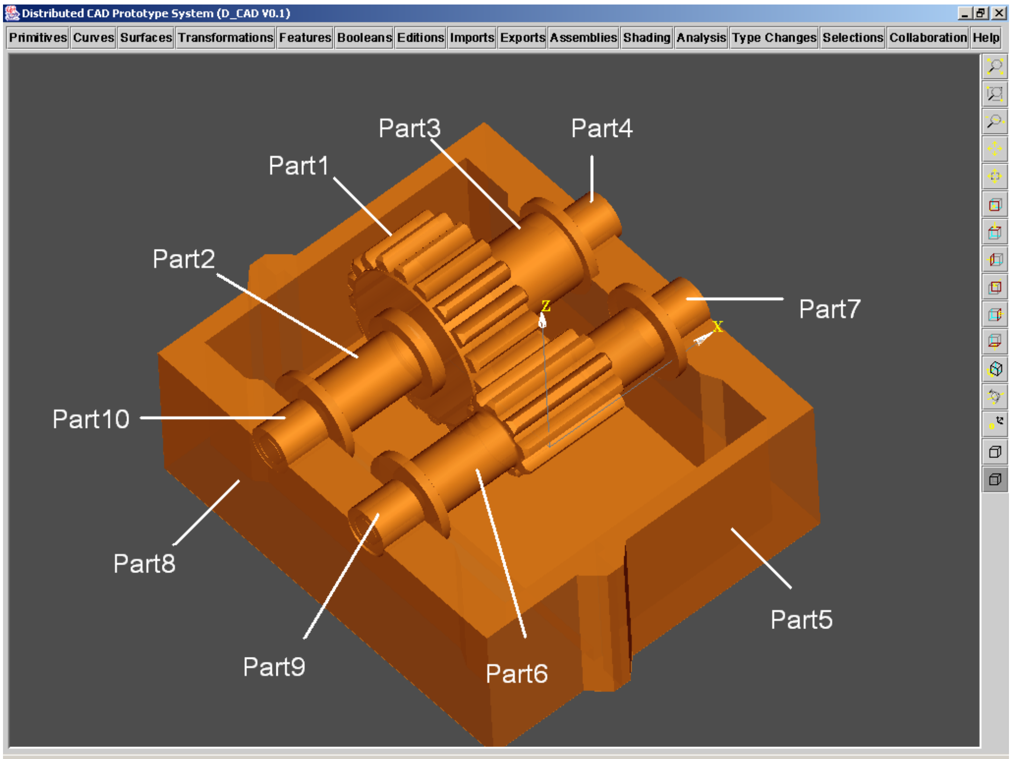


Figure 3.15 Simplified gearbox assembly displayed in design Client 1

Figure 3.16 shows some features (only some features with mating constraints are marked) of each part designed by different designers geographically dispersed.

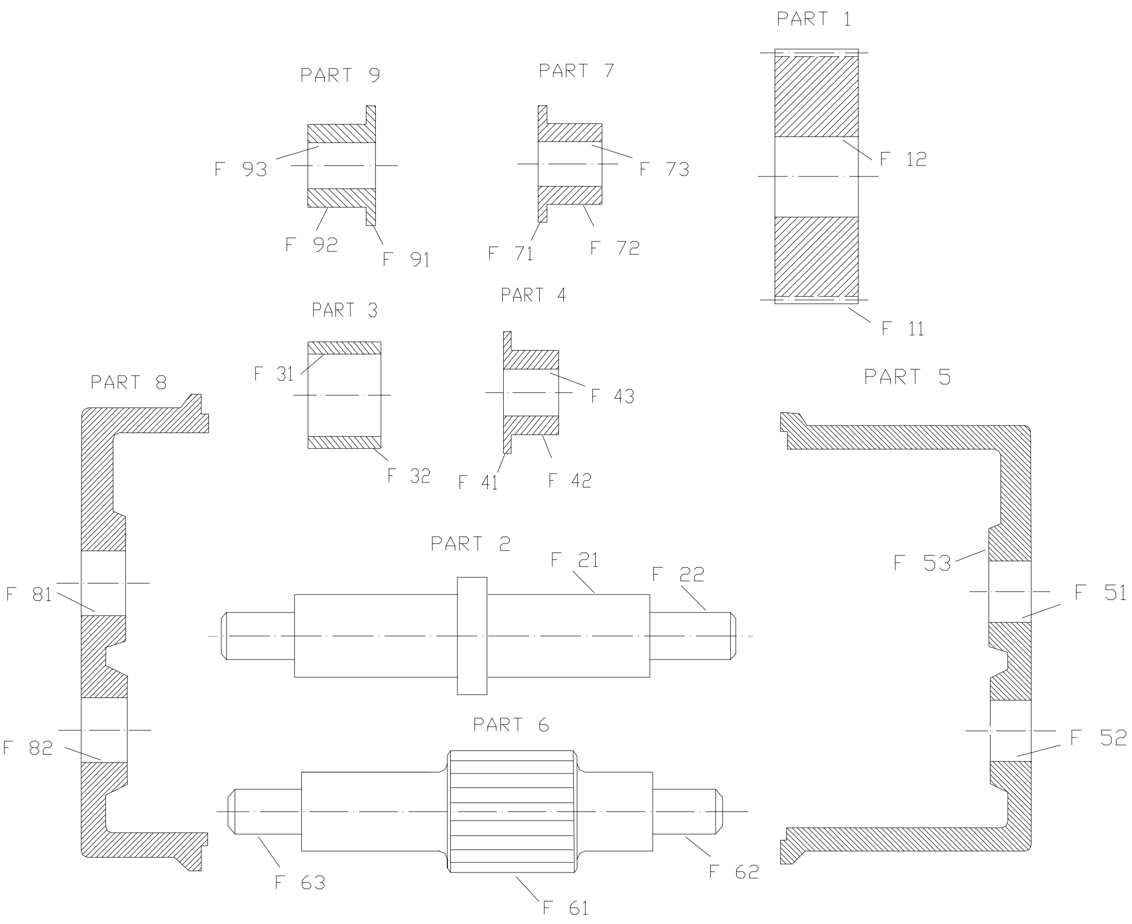
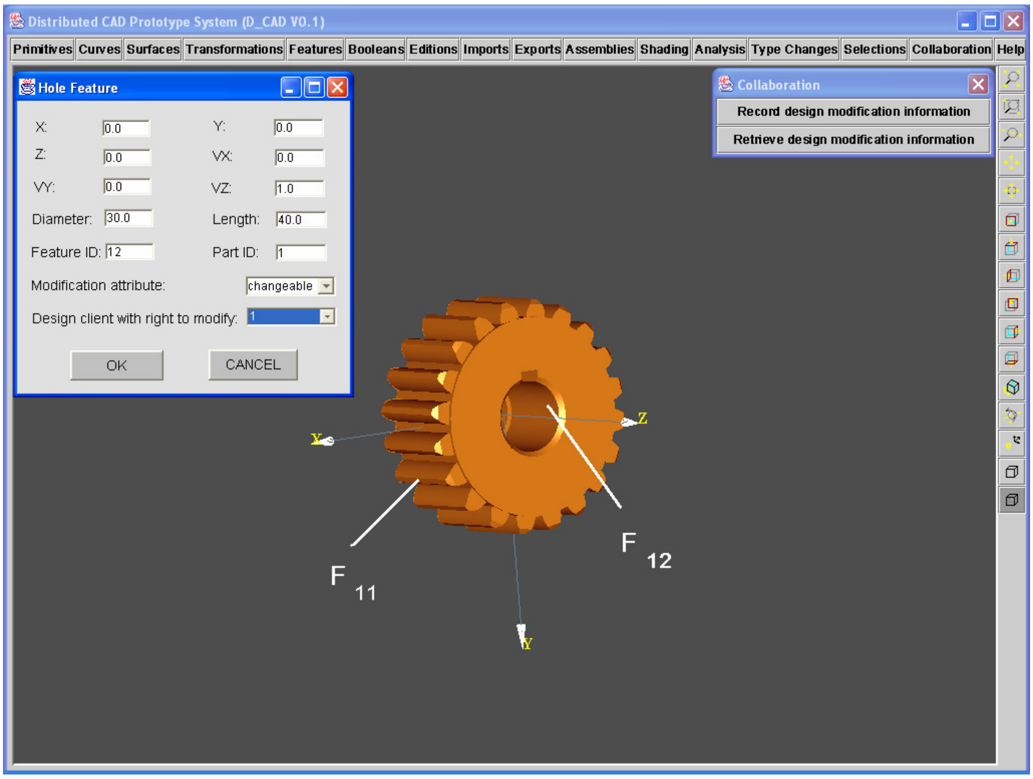


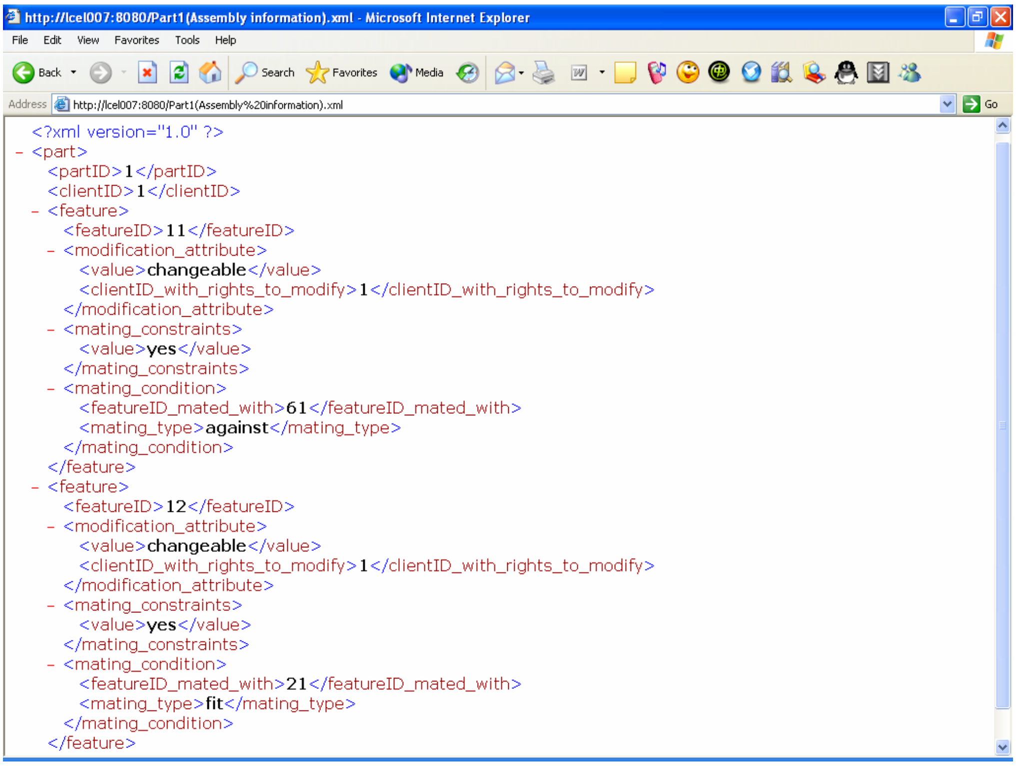
Figure 3.16 Some features of each part

Figure 3.17 shows the Part 1 designed in design Client 1. When the designer in Client 1 modifies the design of Part 1 after the assembly is finished, for example, if feature F_{11} and F_{12} are modified by increasing the teeth number and diameter respectively (the designer can also change the “modification attribute” or “design client with right to modify” of the features), the updated design information can be written into two XML files: one XML file defining the assembly information of each feature in Part 1 (in Figure 3.18), and the other defining the design parameters of each feature in Part 1(in Figure 3.19).



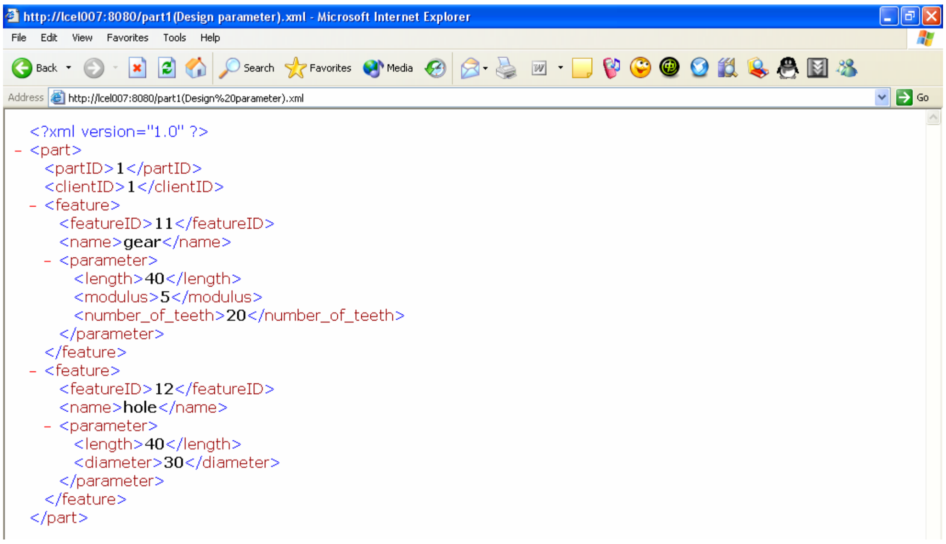
X: 0.0 Y: 0.0
Z: 0.0 VX: 0.0
VY: 0.0 VZ: 1.0
Diameter: 30.0 Length: 40.0
Feature ID: 12 Part ID: 1
Modification attribute: changeable
Design client with right to modify: 1

Figure 3.17 Part 1 in design Client 1



```
  <?xml version="1.0" ?>
- <part>
    <partID>1</partID>
    <clientID>1</clientID>
  - <feature>
      <featureID>11</featureID>
    - <modification_attribute>
        <value>changeable</value>
        <clientID_with_rights_to_modify>1</clientID_with_rights_to_modify>
      </modification_attribute>
    - <mating_constraints>
        <value>yes</value>
      </mating_constraints>
    - <mating_condition>
        <featureID_mated_with>61</featureID_mated_with>
        <mating_type>against</mating_type>
      </mating_condition>
    </feature>
  - <feature>
      <featureID>12</featureID>
    - <modification_attribute>
        <value>changeable</value>
        <clientID_with_rights_to_modify>1</clientID_with_rights_to_modify>
      </modification_attribute>
    - <mating_constraints>
        <value>yes</value>
      </mating_constraints>
    - <mating_condition>
        <featureID_mated_with>21</featureID_mated_with>
        <mating_type>fit</mating_type>
      </mating_condition>
    </feature>
```

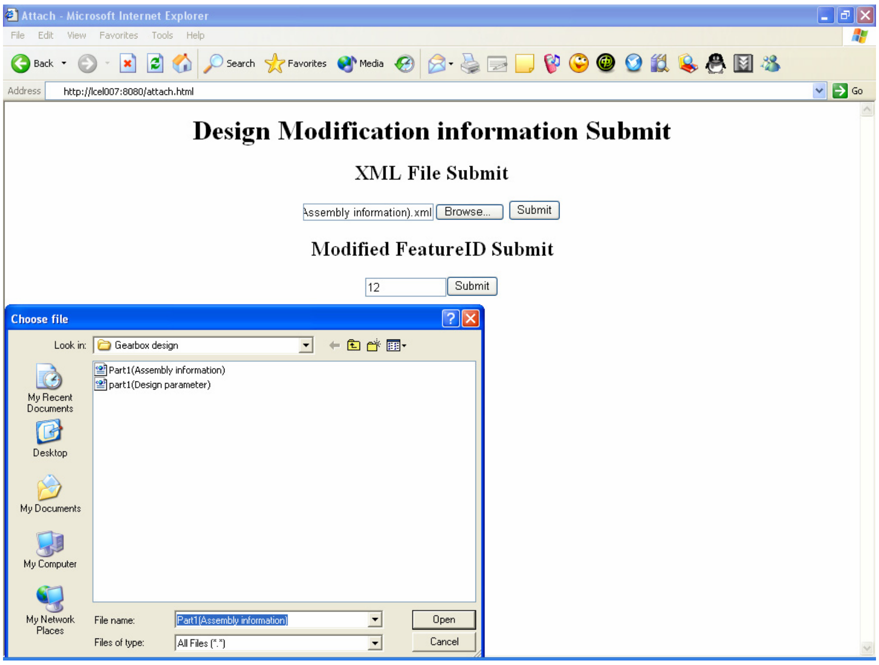
Figure 3.18 XML file 1



```
  <?xml version="1.0" ?>
- <part>
    <partID>1</partID>
    <clientID>1</clientID>
  - <feature>
      <featureID>11</featureID>
      <name>gear</name>
    - <parameter>
        <length>40</length>
        <modulus>5</modulus>
        <number_of_teeth>20</number_of_teeth>
      </parameter>
    </feature>
  - <feature>
      <featureID>12</featureID>
      <name>hole</name>
    - <parameter>
        <length>40</length>
        <diameter>30</diameter>
      </parameter>
    </feature>
  </part>
```

Figure 3.19 XML file 2

Figure 3.20 shows a Java Applet browsed in Client 1 for submitting the design modification information. Each time when designer modifies the design of features, he can browse this applet in the client and submit the corresponding XML files and modified featureID to the Apache web server for design modification propagation control service.



Design Modification information Submit

XML File Submit

Assembly information).xml

Modified FeatureID Submit

12

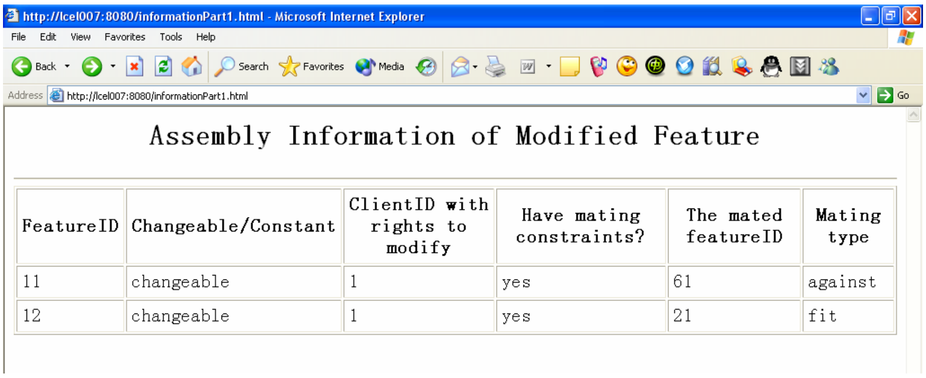
Part1(Assembly information)
part1(Design parameter)

File name: Part1(Assembly information)

Files of type: All Files (*.*)

Figure 3.20 Java Applet browsed in Client 1 for submitting design modification information

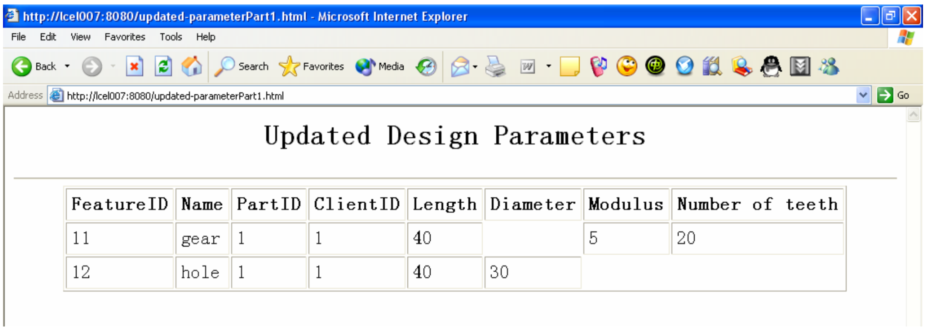
When the Design Modification Propagation Control Service in Apache web server receives these two files, it will execute the parsing function according to the parsing mechanism. The parsing result- two clear readable HTML web pages will be generated automatically in Apache web server and be published on the web. These two web pages, one is embeded with the assembly information of modified features (shown in Figure 3.21), and the other with the updated design parameters of modified features (shown in Figure 3.22).



Assembly Information of Modified Feature

FeatureID	Changeable/Constant	ClientID with rights to modify	Have mating constraints?	The mated featureID	Mating type
11	changeable	1	yes	61	against
12	changeable	1	yes	21	fit

Figure 3.21 Web page 1

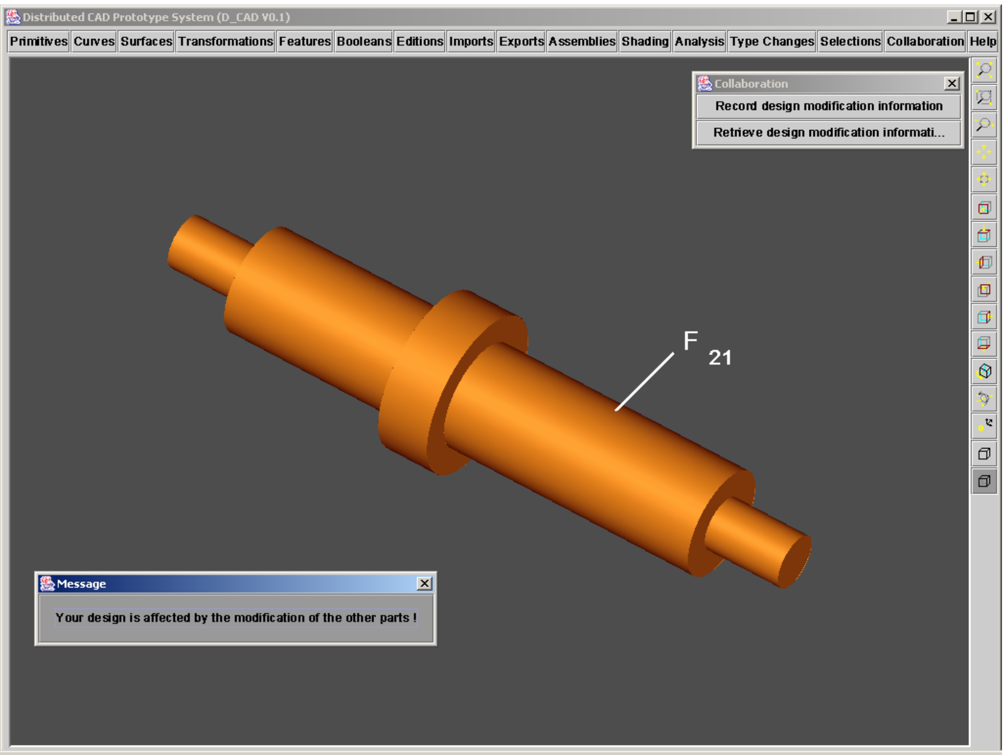


Updated Design Parameters

FeatureID	Name	PartID	ClientID	Length	Diameter	Modulus	Number of teeth
11	gear	1	1	40		5	20
12	hole	1	1	40	30		

Figure 3.22 Web page 2

When the affected featureID was retrieved, for example, F_{21}, the corresponding client ID can be extracted and a message will be sent to the Client 2 to inform the designer about the modification, as shown in Figure 3.23. Then the designer can browse the above web pages to check the detailed information.



Your design is affected by the modification of the other parts !

Figure 3.23 Part 2 in design Client 2

In the following step, the designer in Client 2 modifies feature F_{21} by increasing its diameter, then this modification will also trigger the design modification propagation control service. In this case, the designer in Client 3 (in Figure 3.24) sets the “modification attribute” of feature F_{31} as “constant”, based on the design modification propagation control mechanism, the Design Modification Propagation Control Service in Apache web server cannot propagate this modification to F_{31}, and will feedback the message “Your modification is not permitted, please reconsider!” to Client 2. Then the designer should remodify the design of feature F_{21} based on the assembly information he checked, in this case, feature F_{21} is remodified to a stepped cylinder to adapt to the modification of feature F_{12}, and still maintain the constant assembly condition with feature F_{31}, as shown in Figure 3.25.

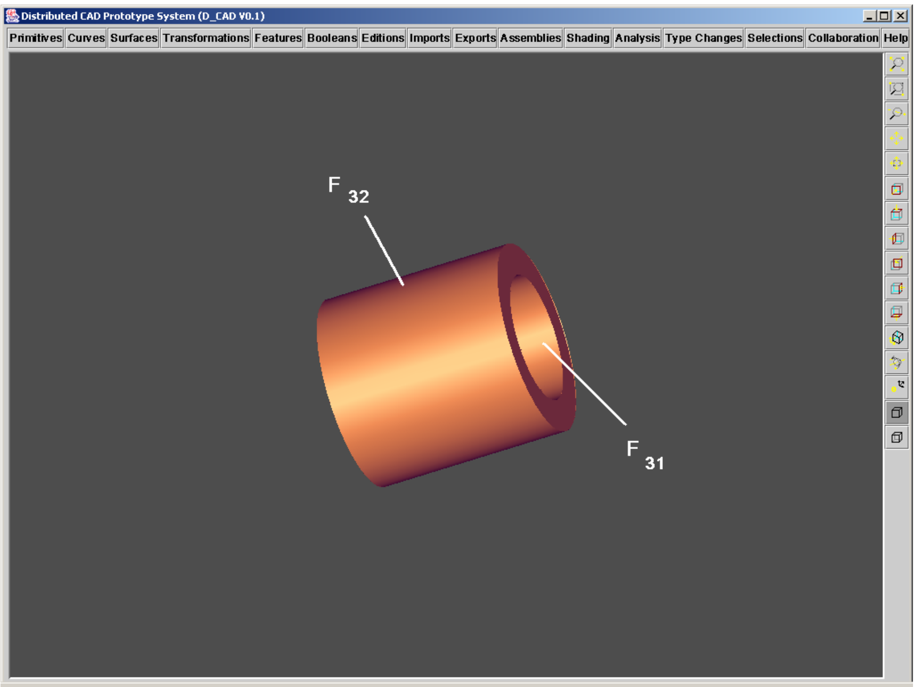
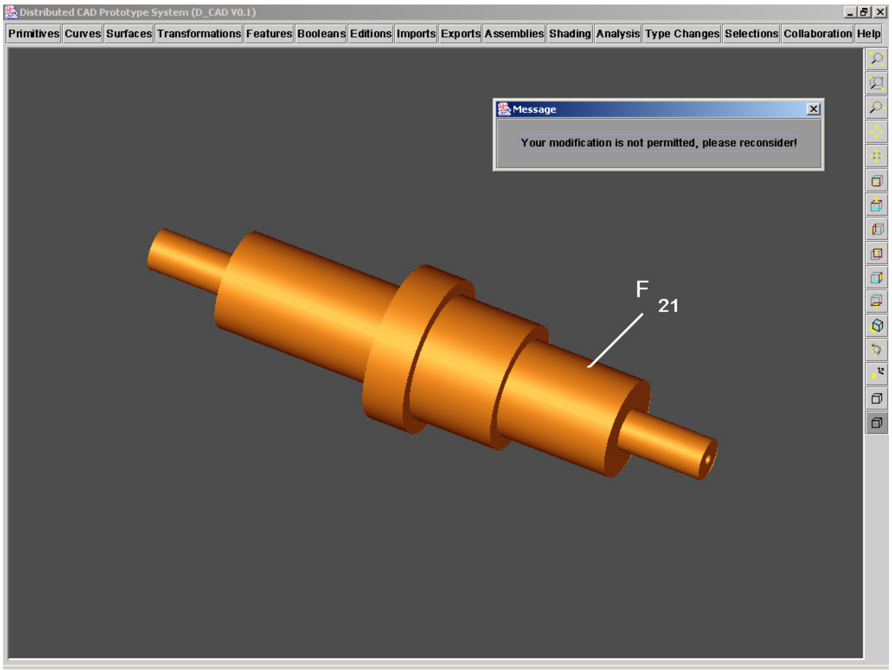


Figure 3.24 Part 3 in design Client 3



Your modification is not permitted, please reconsider!

Figure 3.25 Modified Part 2 in design Client 2

As the same step, the design modification of feature F_{11} will also trigger the design modification propagation control service. If we assume the modification attribute of all features are changeable, then the overall design modification propagation routes are shown in Figure 3.26. Figure 3.27 shows the updated gearbox assembly displayed in Client 1 when the modification of all affected parts is finished and the parts are reassembled together.

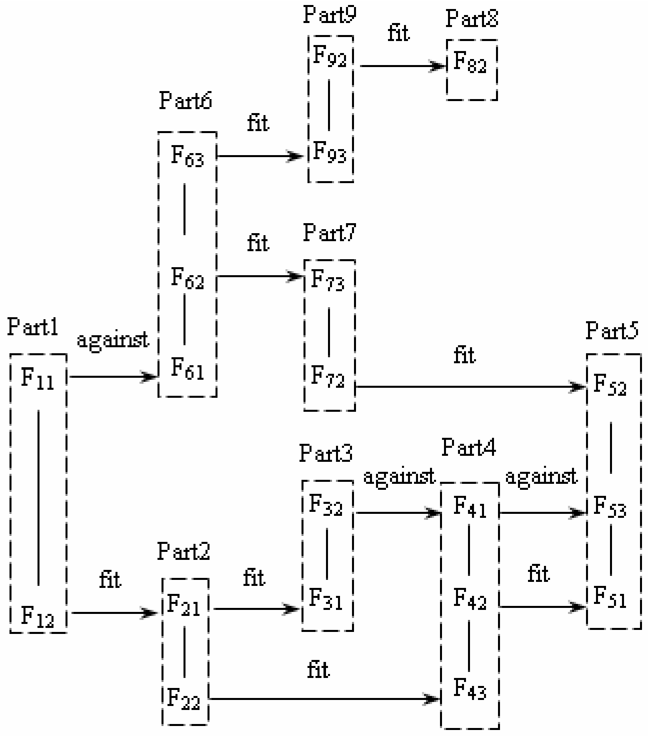


Figure 3.26 The design modification propagation triggered by modification of F_{11} & F_{12}

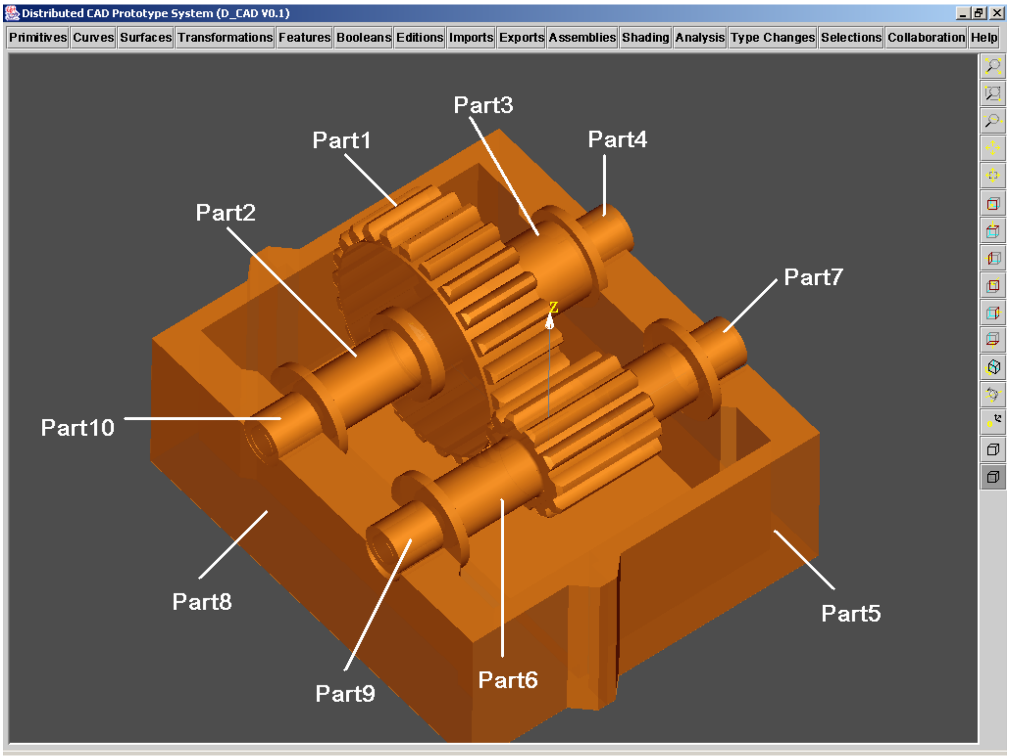


Figure 3.27 Updated gearbox assembly in design Client 1

3.6 Summary

This chapter discusses the design modification issues in a co-assembly design environment. A set of methodologies have been developed to avoid design conflicts and to maintain the validity and consistency of the whole assembly, so that the design modification in a co-assembly design environment can be realized.

Through a co-assembly representation model and a new definition of the assembly feature, the assembly relationship between different parts and the network-based working relationship between geographically dispersed designers can be built up. In addition, the defined assembly feature can help to decide the design modification propagation routes in the co-assembly design process. Based on the proposed co-assembly representation model, an XML schema is proposed to transfer the assembly design information, and a design modification propagation control mechanism has been developed to realize the design modification propagation control. The system framework suitable for realizing the design modification in a co-assembly design environment is proposed and demonstrated.

Chapter 4 Evaluation of Product Assemblability in Different Assembly Sequences

This chapter investigates an approach to evaluate product assemblability in different assembly sequences considering the influence of tolerance and assembly clearance. This approach will be used to help the downstream assembly planning system find optimal assembly sequences with good assemblability, and can also help the designer find the design problems. Section 4.1 discusses the tolerance categorization and representation. A concept called Sensitive Tolerance in Assembly is proposed and its influence on the assembly is investigated. Section 4.2 discusses the clearance zone in assembly and its conversion to geometric deviations in assembly. Section 4.3 proposed an approach using transformation matrices to determine the geometric deviations of mating features caused by the tolerance and assembly clearance, and their propagation and accumulations in the different assembly sequences. A case study is given in Section 4.4 to illustrate the approaches to derive the relative assemblability of a product for comparing different assembly sequences. Section 4.5 summarizes the chapter.

4.1 Tolerance categorization and representation

In order to study the influence of tolerance on assembly, it is necessary to first represent the tolerance in a suitable way that can determine its influence on the feature deviations in the assembly.

4.1.1 Tolerance categorization

Generally tolerance can be categorized into two types: dimensional tolerance and geometric tolerance. Dimensional tolerance directly designates an allowable positional or orientation deviation of a feature in the design space, such as length, diameter or angle.

Apart from dimensional tolerance, geometric tolerance can also play an important role to influence the feature deviations. From Foster [Foster, 1992], ANSI Y 14.5M-1982 [ANSI, 1982] further categorized geometric tolerance into five groups: form, profile, orientation, location and run-out.

4.1.2 Sensitive tolerance in assembly

In a three-dimensional space, each feature in a part can have six degrees of freedom (DOF): three translational DOFs along three axes- X, Y, Z, and three rotational DOFs about those three axes, respectively. According to the six DOFs, the feature deviations are represented as d_x, d_y, d_z and $\delta_x, \delta_y, \delta_z$, where the d_x, d_y, d_z are translational deviations along X, Y, Z axes, respectively, and $\delta_x, \delta_y, \delta_z$ are the corresponding rotational deviations about X, Y, Z axes.

Once a part is assembled with other parts or subassemblies, some DOFs of the mating features will be restricted. In assembly, the tolerance that can result in the geometric deviation of features along the restricted DOF consequently propagate the deviation to the mated features in other parts. We define such tolerance as *Sensitive Tolerance in Assembly* (*STA*) in this paper. The other tolerance that can only cause the geometric deviation of features along the unrestricted DOF in assembly will not propagate the deviation to the mated features in other parts, and accordingly we define such tolerance as *Non-Sensitive tolerance in Assembly* (*NSTA*).

According to the above definition, we analyze the *STA* in some common features in assembly. The first is a cylindrical feature. Before it is assembled into the hole (as shown in Figure 4.1), it has six DOFs and the corresponding geometric deviations are d_x, d_y, d_z and $\delta_x, \delta_y, \delta_z$. However when it is assembled into the hole of the base part, four DOFs- translations along *X*, *Y* and rotations about *X*, *Y* will be restricted, while the two DOFs- translation along *Z* and rotation about *Z* are maintained, as shown in Figure 4.2. Then the geometric deviations $d_x, d_y,$ and δ_x, δ_y of the cylindrical feature will be propagated to the mated hole feature in assembly. According to the definition of *STA*, the tolerances of the cylindrical feature that can cause the geometric deviations- $d_x, d_y,$ and δ_x, δ_y are *STA* in this case.

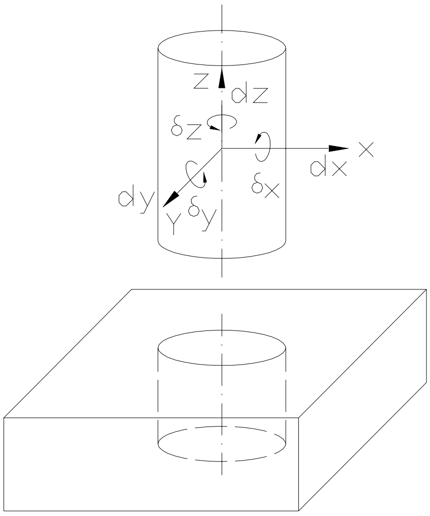


Figure 4.1. Geometric deviation in six DOFs of the cylindrical feature

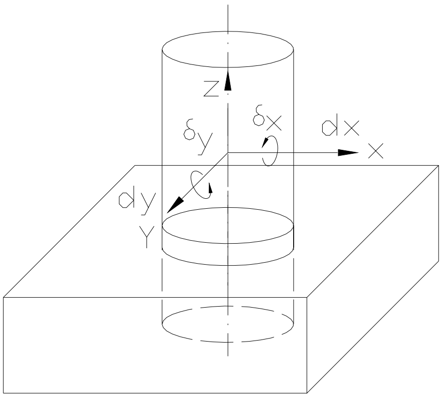
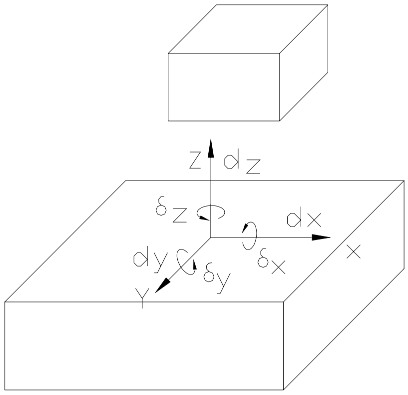


Figure 4.2 Geometric deviation in restricted DOF of the cylindrical feature in assembly

Another common feature used in assembly is the planar feature, as shown in Figure 4.3. Before it is mated against another planar feature, it has six DOFs. However when it is mated, three DOFs- translation along *Z* and rotation about *X* and *Y*, will be restricted, while the other DOFs- translation along *X* and *Y* and rotation about *Z* are maintained, as shown in Figure 4.4. Then the tolerances of the planar feature that can cause the geometric deviations d_z and $\delta_{x,}$ δ_y are *STA* in this case.



Fgure 4.3 Geometric deviation in six DOFs of the planar feature

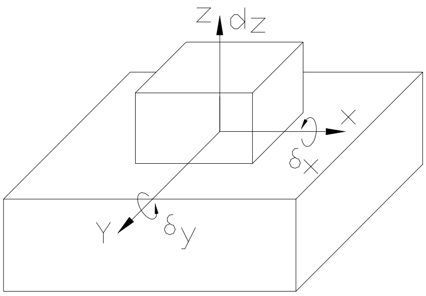


Figure 4.4 Geometric deviation in restricted DOFs of the planar feature in assembly

4.1.3 Converting the STA of features to geometric deviations

In tolerance design, most dimensional and geometric tolerances are set on the feature surface or axis, or use feature surfaces or axes as datum. In order to establish the influence of the dimensional or geometric tolerance of one feature to others in the assembly, the first thing to do is to identify the *STA* in assembly and convert them to geometric deviations. This section uses an example to illustrate the above process.

Case 1:

Figure 4.5 shows the perpendicularity tolerance of a planar feature B against the datum axis A. From this tolerance, we can get a tolerance zone from which the planar feature B can deviate, as shown in Figure 4.6. According to the *STA*, when using the cylindrical feature A as the locating surface in assembly and feature B mated against other planar features, then perpendicularity tolerance 0.1 of planar feature B would be the *STA*. It will cause the geometric deviations- d_z and δ_x, δ_y in the restricted DOF (*X, Y* axis in the plane perpendicular to *Z* Axis).

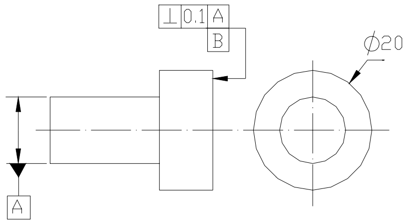


Figure 4.5 Perpendicularity tolerance of planar feature from datum A

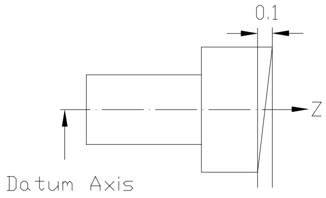


Figure 4.6 Tolerance zone of a planar feature

According to the tolerance zone in Figure 4.6, the deviations d_z is in the range [-0.05, +0.05], δ_x is in the range [-0.1/20, +0.1/20], δ_y is in the range [-0.1/20, +0.1/20], and the other deviations $d_x = d_y = \delta_z = 0$.

Case 2:

In this case, Figure 4.7 indicates a perpendicularity tolerance of axis A of the cylindrical feature from datum surface B. From this tolerance, we can obtain a tolerance zone from which the axis A can deviate, as shown in Figure 4.8. According to the *STA*, when using the planar feature B as the locating surface in assembly and the cylindrical feature A mated with other hole features, then perpendicularity tolerance 0.1 of axis A would be the *STA*, and it will cause the geometric deviations – d_x, d_y and δ_x, δ_y in the restricted DOF. However, the dimensional tolerance ±0.1 is *NSTA*, because it cannot cause geometric deviations in the restricted DOF.

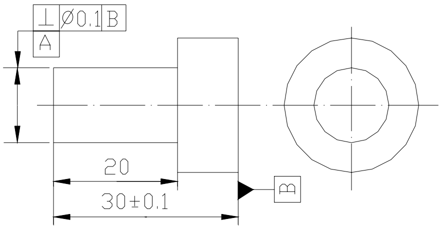


Figure 4.7 Perpendicularity tolerance of cylindrical feature from datum B

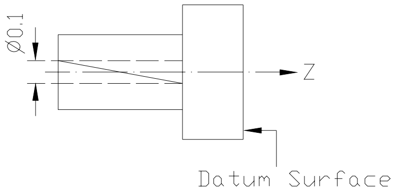


Figure 4.8 Tolerance zone of the axis of cylindrical feature

Then, according to the tolerance zone in Figure 4.8, the deviations d_x is in the range [-0.05, +0.05]，d_y is in the range [-0.05, +0.05]，δ_x is in the range [-0.1/20, +0.1/20], δ_y is in the range [-0.1/20, +0.1/20], and the other deviations $d_z = \delta_z = 0$.

From the above two cases, it can be seen that different dimensional or geometric tolerances and datum can result in different geometric deviations in assembly. For other types of geometric tolerance, using the method described above, we can also similarly determine the corresponding geometric deviations, considering the mating conditions in assembly.

4.2 Clearance in assembly and representation

4.2.1 The role of clearance in assembly

In assembly design, the dimension and geometry of the part usually take into consideration the required clearance between the surfaces of the mated features, such as that in the peg-hole mating condition. Clearance has two main functions: to ensure that the mated parts can be assembled easily, and to compensate for the geometric deviations of mated parts from their nominal dimensions or shapes due to the manufacturing process. For example, Figure 4.9 shows a four-part assembly. The clearance between the surface of the cylindrical feature (in Part 3 and Part 4) and the hole feature (in Part 1 and Part 2) will ensure that these four parts can be assembled together successfully.

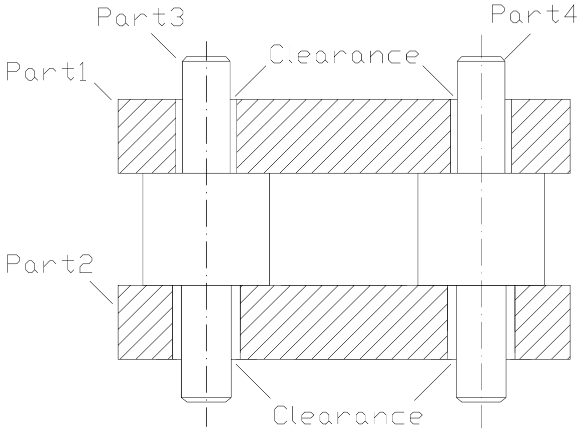


Figure 4.9 Clearance in assembly

However, the resulting clearance can also cause the positional or orientation deviations of the mating features in assembly. For example, as shown in Figure 4.9, the axis of the cylindrical feature of Part 3 can easily deviate from the axis of hole feature of Part 1 in the assembly process. These positional or orientation deviations can propagate and accumulate during the assembly process.

4.2.2 Representation of the clearance zone

In order to establish the influence of a specified clearance in assembly, we need to identify its clearance zone. Clearance in assembly is determined by the corresponding mated features. For example, as shown in Figure 4.9, the clearance between Part 1 and Part 3 is determined by the hole feature of Part 1 and cylindrical feature of Part 3. The clearance zone is then determined by the tolerance zone of these two features.

4.2.2.1 Normal distribution of the tolerance zone

Generally, in a stable manufacturing process, the dimensional and geometric tolerance zone can be assumed to have a normal distribution. A normally distributed variate X with mean μ and standard deviation σ would have the probability density function given by Eq.(4.1), in which $X \in (-\infty, \infty)$, as shown in Figure 4.10.

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)} \tag{4.1}$$

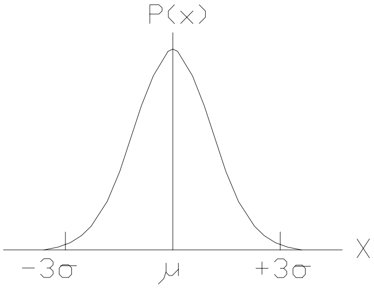


Figure 4.10 Probability of normal distribution

The percentage probability that X lies outside the range $\mu \pm 3\sigma$ is much less than 1%. Hence, we can assume that X is in the range [μ-3σ, μ+3σ]. Thus, for a tolerance $L_i \pm \Delta_i$, we can get $\mu_i = L_i$, $3\sigma_i = \Delta_i$, $\sigma_i = \Delta_i / 3$.

The distribution of a sum or difference of two normally distributed variates is also a normal distribution. For a normal sum distribution, the probability density function can be given by Eq. (4.2):

$$P_{X+Y}(u) = \frac{1}{\sqrt{2\pi(\sigma^2{}_x + \sigma^2{}_y)}} e^{-[\mu-(\mu_x+\mu_y)]^2/[2(\sigma^2{}_x+\sigma^2{}_y)]} \quad \textbf{(4.2)}$$

where X and Y are two normally distributed variates with mean and standard deviation (μ_x, σ_x) and (μ_y, σ_y) respectively. And we can get $\mu_{(x+y)} = \mu_x + \mu_y$, $\sigma^2{}_{(x+y)} = \sigma^2{}_x + \sigma^2{}_y$. For a normal difference distribution, the probability density function is in Eq.(4.3):

$$P_{X-Y}(u) = \frac{1}{\sqrt{2\pi(\sigma^2{}_x + \sigma^2{}_y)}} e^{-[\mu-(\mu_x-\mu_y)]^2/[2(\sigma^2{}_x+\sigma^2{}_y)]} \quad \textbf{(4.3)}$$

And we can get $\mu_{(x-y)} = \mu_x - \mu_y$, $\sigma^2{}_{(x-y)} = \sigma^2{}_x + \sigma^2{}_y$.

4.2.2.2 Normal distribution of the clearance zone

In assembly, the clearance zone is the difference of two normally distributed tolerance zones of mated features, thus the clearance zone is also normally distributed. For example, in the peg-hole mating condition, with the hole diameter $D \pm \Delta_D$ and peg diameter $d \pm \Delta_d$, the hole diameter is normally distributed with the mean and standard deviation (μ_D, σ_D), during which $\mu_D = D$, $\sigma_D = \Delta_D/3$. The peg diameter is normally distributed with the mean and standard deviation (μ_d, σ_d), during which $\mu_d = \mathrm{d}$, $\sigma_d = \Delta_d/3$. Then the clearance zone is normally distributed with the mean and standard deviation (μ_c, σ_c):

$$\mu_\mathrm{c} = \mu_D - \mu_d = D - d, \quad \sigma_c^2 = \sigma_D^2 + \sigma_d^2,$$

then $\sigma_c = [(\Delta_D/3)^2 + (\Delta_d/3)^2]^{1/2} = [(\Delta_D^2 + \Delta_d^2)^{1/2}]/3$, $3\sigma_c = (\Delta_D^2 + \Delta_d^2)^{1/2}$. Thus, as shown in Figure 4.11, the clearance zone is normally distributed in the range:

$$[D - d - (\mathit{\Delta}_D^{\ 2} + \mathit{\Delta}_d^{\ 2})^{1/2},\ \ D - d + (\mathit{\Delta}_D^{\ 2} + \mathit{\Delta}_d^{\ 2})^{1/2}]$$

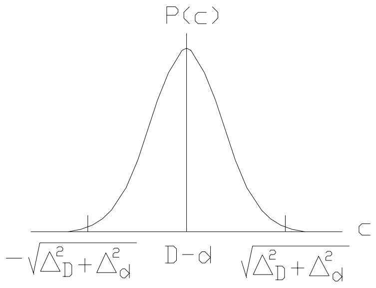


Figure 4.11 Normally distributed clearance zone

4.2.3 Converting the clearance zone to geometric deviations

Clearances in the assembly can also cause translational and rotational deviations of the mating features in assembly. The deviations will be determined by the shape of the clearance zone and the mating condition. In this chapter, we will discuss two types of clearance zone: one is the cylindrical clearance zone in the peg-hole mating condition; the other is the cuboid clearance zone in the rectangular key-hole mating condition.

4.2.3.1 Peg-hole mating condition

The peg-hole mating condition can be subdivided into two conditions as follows:

Condition 1a:

The first condition is to use the cylindrical surface as the locating surface in assembly (as shown in Figure 4.12). The cylindrical clearance zone is shown in Figure 4.13, with diameter *C* and length *L*.

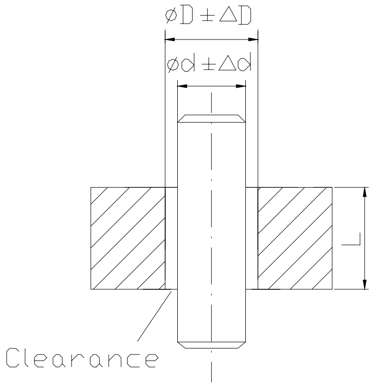


Figure 4.12 Peg-hole mating condition 1a

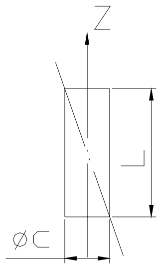


Figure 4.13 Clearance zone 1a

The clearance zone represents the three-dimensional space where the axis of the peg can deviate from the axis of the hole in assembly. In the clearance zone shown in Figure 4.13, the X and Y axes are perpendicular to the Z axis. Similar to the tolerance conversion discussed in section 4.1.3, we can convert this clearance zone into geometric deviations of the peg axis from the hole axis as follows:

The deviation d_x is in the range of $[-C/2, +C/2]$，d_y is in the range $[-C/2, +C/2]$，δ_x is in the range $[-C/L, +C/L]$ and δ_y is in the range $[-C/L, +C/L]$. C is normally distributed in the range: $[D - d - (\Delta_D^2 + \Delta_d^2)^{1/2}, D - d + (\Delta_D^2 + \Delta_d^2)^{1/2}]$. The other deviations are $d_z = \delta_z = 0$.

Condition 2a:

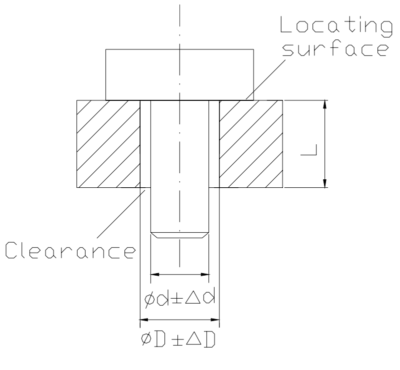


Figure 4.14 Peg-hole mating condition 2a

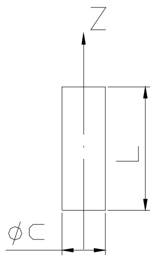


Figure 4.15 Clearance zone 2a

The second condition is to use another surface as the locating surface in assembly. As shown in Figure 4.14, a planar surface is used as the locating surface, with the clearance zone shown in Figure 4.15. In this condition, the locating surface influences the rotational deviations of the peg axis about the hole axis, and the clearance zone can only influence the translational deviations. Thus we can convert this clearance zone into the geometric deviations of the peg axis from the hole axis. The deviation d_x is in the range $[-C/2, +C/2]$，and d_y is in the range $[-C/2, +C/2]$. C is normally distributed in the range:$[D - d - (\Delta_D^2 + \Delta_d^2)^{1/2},\ D - d + (\Delta_D^2 + \Delta_d^2)^{1/2}]$. The other deviations $d_z = \delta_z = 0$. However, δ_x and δ_y will be determined by the perpendicularity tolerance of the peg axis and the hole axis with respect to the locating surface, as in case 2 discussed in section 4.1.3.

4.2.3.2 Rectangular key-hole mating condition

Similar to the peg-hole mating condition, the rectangular key-hole mating condition can also be subdivided into two conditions.

Condition 1b:

The first condition is to use the planar surface of the key as the locating surface in assembly, as shown in Figure 4.16, with the cuboid clearance zone shown in Figure 4.17.

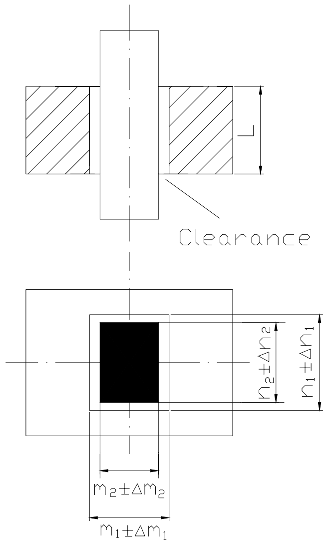


Figure 4.16 Rectangular key-hole mating condition 1b

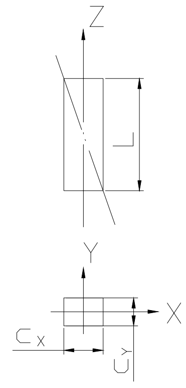


Figure 4.17 Clearance zone 1b

In this condition, because the key is cuboid, when it is mated with the rectangular hole, the restricted DOFs are two translational DOFs along the *X* and *Y* axes and three rotational DOFs rotated about the *X, Y* and *Z* axes respectively. Accordingly, the geometric deviations caused by the clearance zone are d_x and d_y, and δ_x, δ_y and δ_z.

From Figure 4.17, we can convert the clearance zone to the geometric deviations as follows:

The deviation d_x is in the range $[-C_x/2, +C_x/2]$, d_y is in the range $[-C_y/2, +C_y/2]$, $d_z = 0$, δ_x is in the range $[-C_x/L, +C_x/L]$, and δ_y is in the range $[-C_y/L, +C_y/L]$.

C_x is normally distributed in the range:

$$[\, m_1 - m_2 - (\Delta m_1^2 + \Delta m_2^2)^{1/2},\ m_1 - m_2 + (\Delta m_1^2 + \Delta m_2^2)^{1/2}\,],$$

and C_y is normally distributed in the range:

$$[\, n_1 - n_2 - (\Delta n_1^2 + \Delta n_2^2)^{1/2},\ n_1 - n_2 + (\Delta n_1^2 + \Delta n_2^2)^{1/2}\,].$$

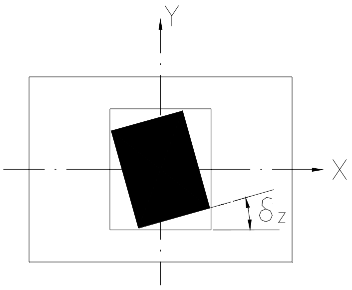


Figure 4.18 Geometric deviation around *Z* axis

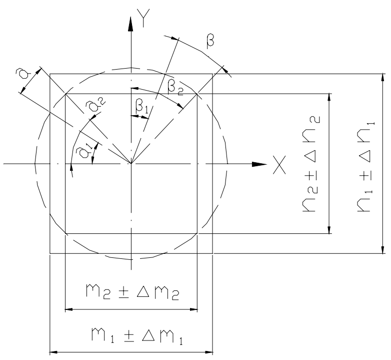


Figure 4.19 Possible maximum δ_Z

From Figure 4.18 and Figure 4.19, we can convert the clearance zone to the geometric deviation δ_Z, $\delta_Z = [-\min[\alpha, \beta], +\min[\alpha, \beta]]$, where, $\alpha=\alpha_2-\alpha_1$, $\alpha_2 = \cos^{-1}[m_2'/[(m_2')^2+(n_2')^2]^{1/2}]$, $\alpha_1=\cos^{-1}[m_1'/[(m_2')^2+(n_2')^2]^{1/2}]$; m_2' is normally distributed in $[m_2-\Delta m_2, m_2+\Delta m_2]$, and m_1' is normally distributed in $[m_1-\Delta m_1, m_1+\Delta m_1]$; $\beta=\beta_2-\beta_1$, $\beta_2=\cos^{-1}[n_2'/[(m_2')^2+(n_2')^2]^{1/2}]$, and $\beta_1=\cos^{-1}[n_1'/[(m_2')^2+(n_2')^2]^{1/2}]$; n_2' is normally distributed in $[n_2-\Delta n_2, n_2+\Delta n_2]$, and n_1' is normally distributed in $[n_1-\Delta n_1, n_1+\Delta n_1]$.

Condition 2b:

The second condition is to use another surface as the locating surface in assembly. As shown in Figure 4.20, a planar surface is used as the locating surface, with the clearance zone shown in Figure 4.21.

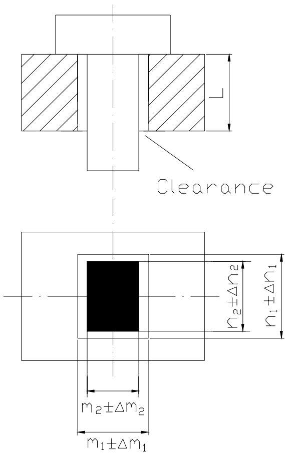


Figure 4.20 Rectangular key-hole mating condition 2b

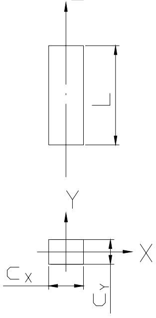


Figure 4.21 Clearance zone 2b

The difference of this condition from the first condition is that the clearance zone can only influence the translational deviations d_x and d_y, and rotational deviation δ_z. The determination of these three deviations is the same as those for condition 1b. For δ_x and δ_y, they are determined by the perpendicularity tolerance of the key axis and hole axis with respect to the locating surface. The other deviation $d_z = 0$ because the translation along *Z* axis is an unrestricted DOF.

4.3 Using transformation matrices to conclude the propagation and accumulation of the geometric deviations

In order to study the propagation and accumulation of the geometric deviations caused by the tolerance and clearance during assembly process, we should consider the geometric deviations of the assembly features in a single coordinate frame.

4.3.1 Transformation matrix

The relationship between different coordinate frames can be specified by a 4×4 homogeneous transformation matrix, which defines the translation and rotation transformation between the coordinate frames. In this chapter, we classify the transformation matrix into nominal transformation matrix, differential transformation matrix [Paul, 1981] and variational transformation matrix.

Nominal transformation matrix

This defines the nominal relationship between coordinate frames specified by the nominal dimensions of geometric features. When we adopt the same *X, Y* and *Z* directions of each coordinate frame, the nominal transformation matrix can be given as follows:

$$T^n = \begin{bmatrix} 1 & 0 & 0 & D_x \\ 0 & 1 & 0 & D_y \\ 0 & 0 & 1 & D_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.4}$$

Where, D_x, D_y *and* D_z are the translation transformation between two coordinate frames.

Differential transformation matrix

This defines the small geometric deviation between two coordinate frames caused by the tolerance and clearance. If we use Δ to represent a differential transformation matrix, then

$$\Delta = \text{Trans}(d_x, d_y, d_z)\text{Rot}(x, \delta_x)\text{Rot}(y, \delta_y)\text{Rot}(z, \delta_z) - I \tag{4.5}$$

where I is the identity matrix, and Trans(d_x, d_y, d_z) is the translation matrix that defines the translation relationship between two coordinate frames, and d_x, d_y, d_z are the translation deviations of assembly features along X, Y and Z axes arising from the tolerance and clearance. Rot(x, δ_x), Rot(y, δ_y) and Rot(z, δ_z) are the rotation matrices that define the rotation relationships between two coordinate frames about X, Y and Z axes respectively, and δ_x, δ_y, δ_z are the rotational deviations about X, Y and Z axes arising from the tolerance and clearance. We assume that δ_x, δ_y, δ_z are small enough, so that $\sin\delta_x = \delta_x$, $\sin\delta_y = \delta_y$, $\sin\delta_z = \delta_z$, and $\cos\delta_x = \cos\delta_y = \cos\delta_z = 1$, and the second order of δ_x, δ_y, δ_z can be considered as zero.

Then the differential transformation matrix Δ can be given as

$$\Delta = \begin{bmatrix} 0 & -\delta_z & \delta_y & d_x \\ \delta_z & 0 & -\delta_x & d_y \\ -\delta_y & \delta_x & 0 & d_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \textbf{(4.6)}$$

When the small geometric deviations are given with respect to the base coordinate frame, the change in the nominal transformation matrix can be given as:

$$dT^{\mathrm{n}} = \Delta T^{n} \qquad \textbf{(4.7)}$$

Variational transformation matrix

The variational transformation matrix consists of the nominal transformation matrix and the differential transformation matrix. It defines the actual relationship between coordinate frames considering the geometric deviations caused by tolerance and clearance.

If we use T^v to represent a variational transformation matrix, then T^v can be given as:

$$T^{v} = T^{n} + dT^{n} = T^{n} + \Delta T^{n} \qquad \textbf{(4.8)}$$

If we set the coordinate frames at the surface or axis of geometric features in assembly, then T^v represents the actual transformation relationship between the features considering the tolerance and clearance influence in assembly.

4.3.2 Coordinates conversion between coordinate frames

If $T_{1,2}$ is a transformation matrix that transforms coordinate frame F_1 to F_2, O_1 and O_2 are the respective origins of these two frames, P_1 and P_2 are the coordination of point P in frames F_1 and F_2 respectively, then

$$P_1 = T_{1,2}\, P_2$$

Thus, for n frames F_1, F_2, ……F_n, and n transformation matrices $T_{1,2}$, $T_{2,3}$, ……$T_{n-1,n}$, given the coordinate of point P in frames F_1, F_2, ……F_n as P_1, P_2……P_n,

$$P_1 = T_{1,2}\, P_2,\ P_2 = T_{2,3}\, P_3,\ \ldots\ldots,\ P_{n-1} = T_{n-1,n}\, P_n,$$

Then,

$$P_1 = T_{1,2}\, T_{2,3} \ldots\ldots T_{n-1,n}\, P_n \qquad \textbf{(4.9)}$$

Using Eq. (4.9), we can conclude the coordinates of a point on the assembly features in different coordinate frames set at the surface or axis of geometric features in assembly.

4.4 Assemblability evaluation in different assembly sequences

In this section, we use an example to discuss the approach of evaluating the product assemblability in different assembly sequences. Figure 4.22 shows an assembly that joins the Part 1, Part 2, Part 3 and Part 4 by screwing with Part 5, Part 6, Part 7 and Part 8.

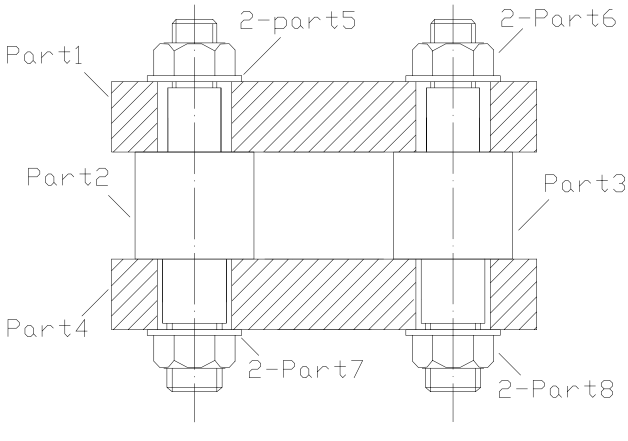


Figure 4.22 Assembly consisting of 12 parts

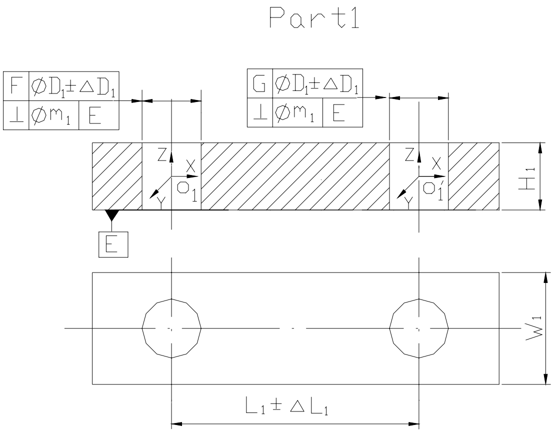


Figure 4.23 Tolerance design in Part 1

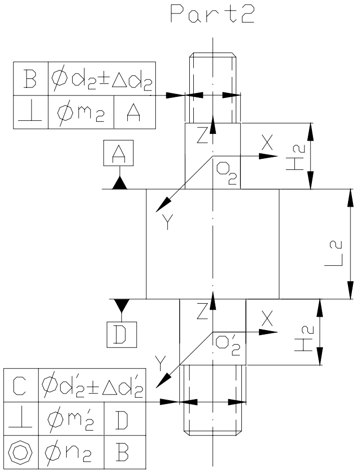


Figure 4.24 Tolerance design in Part 2

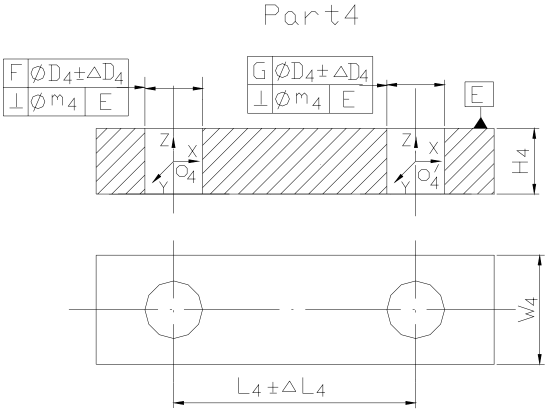


Figure 4.25 Tolerance design in Part 4

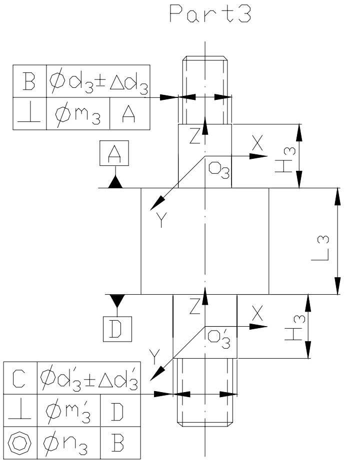


Figure 4.26 Tolerance design in Part 3

Figure 4.23, Figure 4.24, Figure 4.25 and Figure 4.26 shows the tolerance design of Part 1, Part 2, Part 4 and Part 3, respectively. The dimension and tolerance of each feature in these four parts are given in Table 4.1:

Table 4.1 The dimension and tolerance of each feature in assembly

Part1	L_1	ΔL_1	D_1	ΔD_1	m_1	W_1	H_1		
	60	0.1	10	0.1	0.1	40	20		
Part2	L_2	H_2	d_2	Δd_2	d_2'	Δd_2'	m_2	m_2'	n_2
	40	20	9	0.1	14.8	0.05	0.1	0.1	0.05
Part3	L_3	H_3	d_3	Δd_3	d_3'	Δd_3'	m_3	m_3'	n_3
	40	20	9	0.1	14.8	0.05	0.1	0.1	0.05
Part4	L_4	ΔL_4	D_4	ΔD_4	m_4	W_4	H_4		
	60	0.1	15	0.1	0.1	40	20		

Given assembly sequence 1: (number represents part number)

1-2-5-6-3-5-6-4-7-8-7-8

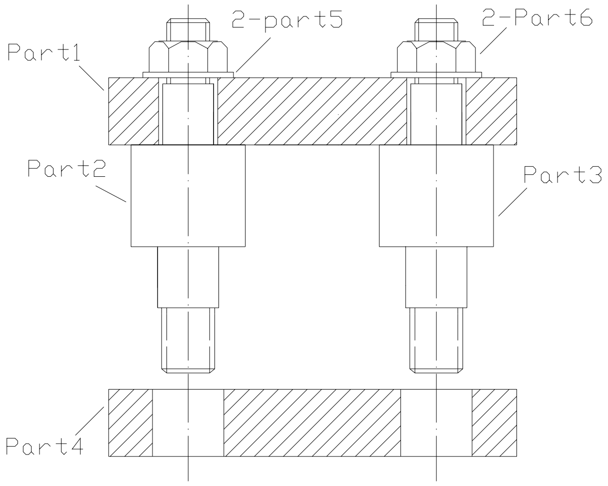


Figure 4.27 Assembly sequence 1

In the above assembly sequence (shown in Figure 4.27), given the design tolerance, we can ensure that Part 1 can be assembled with Part 2, Part 3, Part 5 and Part 6 properly. However the tolerance and clearance propagation and accumulation in the subassembly – Part 1, Part 2, Part 3, Part 5 and Part 6 will influence the assemblability between this subassembly with Part 4. This assemblability is concluded in following steps:

Step 1: Calculate the geometric deviations of cylindrical feature *C* in Part 3 from the hole feature *F* in Part 1

(1) Calculate $T^v_{(1,1')}$

Set global coordinate frame F_1 on the axis of hole feature *F* in Part 1. O_1 is the origin of F_1, and it is located at the center of the hole. Set the local coordinate frame $F_{1'}$ on the axis of hole feature *G* in Part 1. $O_{1'}$ is the origin of $F_{1'}$, and it is located at the center of the hole (shown in Figure 4.23).

1) For the nominal transformation matrix $T^n_{(1,1')}$ that transforms the coordinate frame F_1 to $F_{1'}$, $D_x=L_1$, $D_y = D_z = 0$, then $T^n_{(1,1')}$ is as:

$$T^n_{(1,1')} = \begin{bmatrix} 1 & 0 & 0 & L_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2) For the differential transformation matrix $\Delta_{(1,1')}$ that transforms the coordinate frame F_1 to $F_{1'}$, $d_x = [-\Delta L_1, \Delta L_1]$, $d_y = d_z = 0$, $\delta x = \delta y = \delta z = 0$, then $\Delta_{(1,1')}$ is as:

$$\Delta_{(1,1')} = \begin{bmatrix} 0 & 0 & 0 & [-\Delta L_1, \Delta L_1] \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

3) Then the variational transformation matrix $T^v_{(1,1')}$ that transforms the coordinate

frame F_1 to $F_{1'}$ is as:

$$T^{v}_{(1,1')} = T^{n}_{(1,1')} + dT^{n}_{(1,1')} = T^{n}_{(1,1')} + \Delta_{(1,1')}\, T^{n}_{(1,1')}$$

$$= \begin{bmatrix} 1 & 0 & 0 & [L_1 - \Delta L_1, L_1 + \Delta L_1] \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(2) Calculate $T^{v}_{(1',3)}$

Set the coordinate frame $F_{1'}$ with the origin $O_{1'}$ in Part 1 as the base coordinate frame (shown in Figure 4.23). Set the local coordinate frame F_3 on the axis of cylindrical feature *B* in Part 3. O_3 is the origin of F_3, and it is located at the center of the cylindrical feature *B*, as shown in Figure 4.26.

1) For the nominal transformation matrix $T^{n}_{(1',3)}$ that transforms the coordinate frame $F_{1'}$ to F_3, $D_x = D_y = D_z = 0$, then $T^{n}_{(1',3)}$ is as:

$$T^{n}_{(1',3)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2) For the peg-hole mating between the hole feature *G* and the cylindrical feature *B*, a clearance exists. As discussed in the section 4.2.3, this clearance can cause the geometric deviations as follows:

d_x is in the range [-*C*/2, *C*/2], d_y is in the range [-*C*/2, *C*/2], and *C* is normally distributed in the range $[D_1 - d_3 - (\Delta D_1^2 + \Delta d_3^2)^{1/2},\ D_1 - d_3 + (\Delta D_1^2 + \Delta d_3^2)^{1/2}]$. In this case, we let $C = D_1 - d_3 + [(\varDelta D_1)^2 + (\varDelta d_3)^2]^{1/2}$ for the worst case analysis. δ_x and δ_y are determined by the perpendicularity tolerance of the axis of hole feature G and cylindrical feature B with respect to the locating surfaces E and A, respectively. As discussed in section 4.1.3, $\delta x = \delta y = [-(m_1/H_1+m_3/H_3), (m_1/H_1+m_3/H_3)]$. The other deviations $d_z = \delta_z = 0$.

Then the differential transformation matrix $\varDelta_{(1',3)}$ that transforms the coordinate frame $F_{1'}$ to F_3 can be given as (let $m = m_1/H_1+m_3/H_3$):

$$\varDelta_{(1',3)} = \begin{bmatrix} 0 & 0 & [-m,m] & [-C/2,C/2] \\ 0 & 0 & -[-m,m] & [-C/2,C/2] \\ -[-m,m] & [-m,m] & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

3) Hence, the variational transformation matrix $T^v_{(1',3)}$ that transforms the coordinate frame $F_{1'}$ to F_3 can be given as:

$$T^v_{(1',3)} = T^n_{(1',3)} + dT^n_{(1',3)} = T^n_{(1',3)} + \varDelta_{(1',3)}\ T^n_{(1',3)}$$

$$= \begin{bmatrix} 1 & 0 & [-m,m] & [-C/2,C/2] \\ 0 & 1 & -[-m,m] & [-C/2,C/2] \\ -[-m,m] & [-m,m] & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(3) Calculate $T^v_{(3,3')}$

Set the local coordinate frame $F_{3'}$ in Part 3 (shown in Figure 4.26), and the origin $O_{3'}$ are set at the center of the cylinder feature C.

1) For the nominal transformation matrix $T^n_{(3,3')}$ that transforms the coordinate

frame F_3 to $F_{3'}$, $D_x = D_y = 0$, $D_z = -(H_3+L_3)$, $\delta x = \delta y = \delta z = 0$, then $T^{n}_{(3,3')}$ is as:

$$T^{n}_{(3,3')} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -(H_3+L_3) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2) For the differential transformation matrix $\varDelta_{(3,3')}$ that transforms the coordinate frame F_3 to $F_{3'}$ (shown in Figure 4.28), $d_x = d_y = [-n_3/2, n_3/2]$, $d_z = 0$, $\delta x = \delta y = [-2n_3/H_3, 2n_3/H_3]$, $\delta z = 0$, then $\varDelta_{(3,3')}$ is as: $\varDelta_{(3,3')} =$

$$\begin{bmatrix} 0 & 0 & [-2n_3/H_3, 2n_3/H_3] & [-n_3/2, n_3/2] \\ 0 & 0 & -[-2n_3/H_3, 2n_3/H_3] & [-n_3/2, n_3/2] \\ -[-2n_3/H_3, 2n_3/H_3] & [-2n_3/H_3, 2n_3/H_3] & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

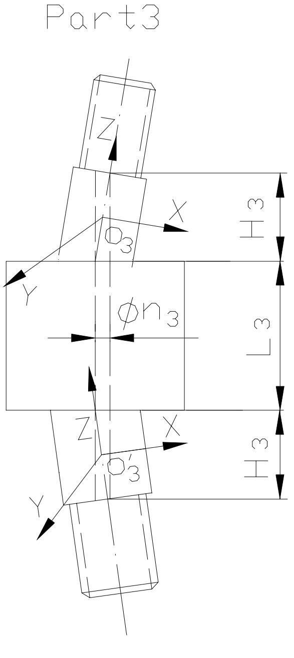


Figure 4.28 Concentricity in Part 3

3) Then the variational transformation matrix $T^{v}_{(3,3')}$ that transforms the coordinate

frame F_3 to $F_{3'}$ can be given as:

$$T^v_{(3,3')} = T^n_{(3,3')} + dT^n_{(3,3')} = T^n_{(3,3')} + \Delta_{(3,3')} T^n_{(3,3')} = (I + \Delta_{(3,3')})\, T^n_{(3,3')},$$

By using Normal Sum Distribution and Normal Difference Distribution to the above matrix, $T^v_{(3,3')} =$

$$\begin{bmatrix} 1 & 0 & [-2n_3/H_3, 2n_3/H_3] & [-Z_1, Z_1] \\ 0 & 1 & -[-2n_3/H_3, 2n_3/H_3] & [-Z_1, Z_1] \\ -[-2n_3/H_3, 2n_3/H_3] & [-2n_3/H_3, 2n_3/H_3] & 1 & -(H_3 + L_3) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where, $Z_1 = \{(n_3/2)^2 + [2n_3(H_3+L_3)/H_3]^2\}^{1/2}$.

(4) Calculate the coordinate of $O_{3'}$ in the global coordinate frame F_1

Let the coordinate of $O_{3'}$ in the local coordinate frame $F_{3'}$ be (0,0,0). Then the coordinate of $O_{3'}$ in global coordinate frame F_1 can be calculated as follows:

$$O_{3'}\Big|_{F_1} = T^v_{(1,3')}\, O_{3'}\Big|_{F_{3'}} = T^v_{(1,1')}\, T^v_{(1',3)}\, T^v_{(3,3')}\, O_{3'}\Big|_{F_{3'}}$$

Using the Normal Sum Distribution and Normal Difference Distribution with the above matrix, and the second order of tolerance approximated to zero, then

$$O_{3'}\Big|_{F_1} =$$

$$\begin{bmatrix} [L_1 - \{Z_1^2 + (H_3 + L_3)^2 m^2 + (C/2)^2 + (\Delta L_1)^2\}^{1/2}, L_1 + \{Z_1^2 + (H_3 + L_3)^2 m^2 + (C/2)^2 + (\Delta L_1)^2\}^{1/2}] \\ [-\{Z_1^2 + (H_3 + L_3)^2 m^2 + (C/2)^2\}^{1/2}, \{Z_1^2 + (H_3 + L_3)^2 m^2 + (C/2)^2\}^{1/2}] \\ -(H_3 + L_3) \\ 1 \end{bmatrix}$$

<u>Step 2</u>: Calculate the geometric deviations of cylindrical feature C in Part 2 from the hole feature F in Part 1

Using similar steps as above, when we set the local coordinate frame $F_{2'}$ in the cylindrical feature C in Part 2, and the origin $O_{2'}$ is at the center of the cylindrical feature C (shown in Figure 4.24), we can calculate the coordinate of $O_{2'}$ in global coordinate frame F_1 as follows:

$$O_{2'}\Big|_{F_1} = T^{v}_{(1,2')}\, O_{2'}\Big|_{F_{2'}} = T^{v}_{(1,2)}\, T^{v}_{(2,2')}\, O_{2'}\Big|_{F_{2'}}$$

$$= \begin{bmatrix} [-\{Z_2^{\,2} + (H_2+L_2)^2 m'^2 + (C'/2)^2\}^{1/2}, \{Z_2^{\,2} + (H_2+L_2)^2 m'^2 + (C'/2)^2\}^{1/2}] \\ [-\{Z_2^{\,2} + (60m')^2 + (C'/2)^2\}^{1/2}, \{Z_2^{\,2} + (60m')^2 + (C'/2)^2\}^{1/2}] \\ -(H_2+L_2) \\ 1 \end{bmatrix}$$

where, $m' = m_1/H_1 + m_2/H_2$, $C' = D_1 - d_2 + [(\Delta D_1)^2 + (\Delta d_2)^2]^{1/2}$, and

$Z_2 = \{(n_2/2)^2 + [2n_2(H_2+L_2)/H_2]^2\}^{1/2}$.

Step 3: Compare the geometric deviations of cylindrical feature C in Part 2 and Part 3 with the worst-case assembly clearance between Part 2, Part 3 and Part 4.

For the worst-case analysis, we should calculate the minimum and maximum distances between $O_{2'}$ and $O_{3'}$ and the minimum and maximum distances between O_4 and $O_{4'}$ in Part 4 in the X direction. Let $\Delta X_{(O2')} = \{Z_2^2 + (H_2+L_2)^2 m'^2 + (C'/2)^2\}^{1/2}$, $\Delta X_{(O3')} = \{Z_1^2 + (H_3+L_3)^2 m^2 + (C/2)^2 + (\Delta L_1)^2\}^{1/2}$, then, as shown in Figure 4.29, the minimum distance between $O_{2'}$ and $O_{3'}$ is:

$$Min\ D_{x\,(O2'\,O3')} = L_1 - \Delta X_{(O2')} - \Delta X_{(O3')},$$

and the maximum distance between $O_{2'}$ and $O_{3'}$ is:

$$Max\ D_{x\,(O2'\,O3')} = L_1 + \Delta X_{(O2')} + \Delta X_{(O3')}$$

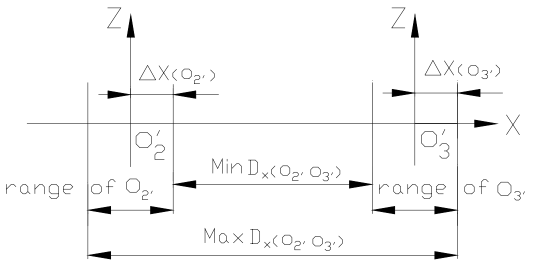


Figure 4.29 Distance between $O_{2'}$ and $O_{3'}$

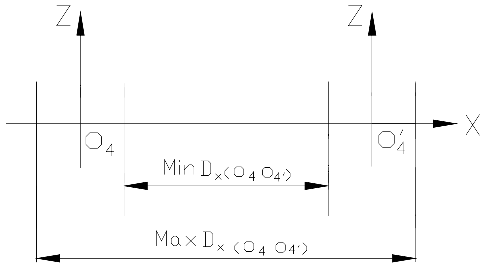


Figure 4.30 Distance between O_4 and $O_{4'}$

From Figure 4.30, the minimum distance between O_4 and $O_{4'}$ in Part 4 is:

$$Min\ D_{x\,(O4\,O4')} = L_4 - \Delta L_{4,}$$

and the maximum distance between O_4 and $O_{4'}$ in Part 4 is:

$$Max\ D_{x\,(O4\,O4')} = L_4 + \Delta L_4$$

Worst case 1: the distance between $O_{2'}$ and $O_{3'}$ is minimum, and the distance between O_4 and $O_{4'}$ in part 4 is maximum

Worst case 2: the distance between $O_{2'}$ and $O_{3'}$ is maximum, and the distance between O_4 and $O_{4'}$ in part 4 is minimum.

The minimum clearance between hole feature *F* in Part 4 and cylindrical feature *C* in Part 2 can be given as: $C_{1\min} = D_4 - \Delta D_4 - d_2' - \Delta d_2'$. The minimum clearance between hole feature *G* in Part 4 and cylindrical feature *C* in Part 3 can be given as: $C_{2\min} = D_4 - \Delta D_4 - d_3' - \Delta d_3'$. In both worst cases, to ensure that the hole features *F* and *G* in Part 4 can be assembled with the cylindrical features *C* in Part 2 and Part 3 respectively, the minimum clearances C_{1min} and C_{2min} should be enough to compensate the geometric deviations in *X* direction of the above cylindrical features in assembly subject to the following condition:

$$Min(C_{1\min}, C_{2\min}) \geq Max\, [\, \Delta X_{(O2')},\ \ \Delta X_{(O3')}] + \Delta L_4 =$$

$$Max\, [\{Z_2{}^2 + (H_2+L_2)^2 m'^2 + (C'/2)^2\}^{1/2},\ \{Z_1{}^2 + (H_3+L_3)^2 m^2 + (C/2)^2 + (\Delta L_1)^2\}^{1/2}] + \Delta L_4 \quad \textbf{(4.10)}$$

In the inequality given by Eq. (4.10),

$C_{1\min} = D_4 - \Delta D_4 - d_2' - \Delta d_2'$; $C_{2\min} = D_4 - \Delta D_4 - d_3' - \Delta \mathrm{d}_3'$;

$C = D_1 - d_3 + [(\Delta D_1)^2 + (\Delta d_3)^2]^{1/2}$; $C' = D_1 - d_2 + [(\Delta D_1)^2 + (\Delta d_2)^2]^{1/2}$;

$m = m_1/H_1 + m_3/H_3$; $m' = m_1/H_1 + m_2/H_2$;

$Z_1 = \{(n_3/2)^2 + [2n_3(H_3+L_3)/H_3]^2\}^{1/2}$; $Z_2 = \{(n_2/2)^2 + [2n_2(H_2+L_2)/H_2]^2\}^{1/2}$.

If the condition given by Eq. (4.10) is not met, the assemblability of the assembly sequence 1: 1-2-5-6-3-5-6-4-7-8-7-8 cannot be ensured, and an alternative assembly sequence is needed.

Consider another assembly sequence 2:

4-2-7-8-3-7-8-1-5-6-5-6

To ensure good assemblability in sequence 2, using the method mentioned before, we can derive the following inequality conditions:

$$Min(C_{1\min}, C_{2\min}) \geq Max\, [\, \Delta X_{(\mathrm{O2})},\ \ \Delta X_{(\mathrm{O3})}] + \Delta L_1 =$$

$$Max\, [\{(Z_2')^2 + (60m')^2 + (C'/2)^2\}^{1/2},\ \{(Z_1')^2 + (60m)^2 + (C/2)^2 + (\Delta L_4)^2\}^{1/2}] + \Delta L_1 \quad \textbf{(4.11)}$$

In Eq.(4.11),

$C_{1\min} = D_1 - \Delta D_1 - d_2 - \Delta d_2$; $C_{2\min} = D_1 - \Delta D_1 - d_3 - \Delta d_3$;

$C = D_4 - d_3' + [(\Delta D_4)^2 + (\Delta d_3')^2]^{1/2}$; $C' = D_4 - d_2' + [(\Delta D_4)^2 + (\Delta d_2')^2]^{1/2}$,

$m = m_4/H_4 + m_3'/H_3$; $m' = m_4/H_4 + m_2'/H_2$;

$Z_1' = \{(n_3/2)^2 + [2n_3(H_3+L_3)/H_3]^2\}^{1/2}$; $Z_2' = \{(n_2/2)^2 + [2n_2(H_2+L_2)/H_2]^2\}^{1/2}$.

Comparison between the above two assembly sequences:

Given the dimension and tolerance of each feature as shown in Table 4.1:

For assembly sequence 1:

$Min(C_{1\min}, C_{2\min}) = 0.05$, $Max\ [\Delta X_{(O2')}, \Delta X_{(O3')}] + \Delta L_4 = 0.986$. Thus, the inequality condition according to Eq.(4.10) is not met, and the assemblability of this sequence is not ensured.

For assembly sequence 2:

$Min(C_{1\min}, C_{2\min}) = 0.8$, $Max\ [\Delta X_{(O2)}, \Delta X_{(O3)}] + \Delta L_1 = 0.796$. Thus the inequality condition according to Eq.(4.11) is satisfied, and the assemblability of this sequence is ensured.

From the above comparison, it can be seen that the assembly sequence affects assemblability. A relative assemblability (*RA*) can be given as the ratio between the two sides of the above inequation. The *RA* of the assembly sequence 1 is as follows:

$$RA = Min(C_{1\min}, C_{2\min}) / \{ Max\ [\Delta X_{(O2')}, \Delta X_{(O3')}] + \Delta L_4 \} = 0.051,$$

and the *RA* of the assembly sequence 2 is as follows:

$$RA = Min(C_{1\min}, C_{2\min}) / \{ Max\ [\Delta X_{(O2)}, \Delta X_{(O3)}] + \Delta L_1 \} = 1.005$$

If $RA > 1$, then the assemblability of the assembly sequence can be ensured; otherwise, a smaller *RA*, indicates worse assemblability for the sequence.

4.5 Summary

This chapter presents an approach to evaluate product assemblability in different assembly sequences. By analyzing the influence of tolerance and assembly clearance on assembly, transformation matrices are used to conclude the geometric deviations of mating features and their propagations and accumulations in different assembly sequences. Through comparison between the geometric deviations and the assembly clearance at the final assembly process, the relative assemblability of the product in different assembly sequences can be determined. This approach has been verified and demonstrated through an example, and it can be integrated into an assembly planning system to generate more feasible assembly sequences, which will be discussed in Chapter 5.

Chapter 5 An Enhanced Assembly Planning Approach Using a Multi-objective Genetic Algorithm

This chapter proposes an enhanced assembly planning approach using a multi-objective genetic algorithm (GA). The influence of tolerance and clearance on product assemblability in different assembly sequences is considered and used as a constraint in assembly planning. For more comprehensive search for feasible non-dominated solutions, this chapter proposes a multi-objective genetic algorithm which establishes different fitness functions through a fuzzy weight distribution algorithm. It also considers the experience of the decision maker.

The organization of this chapter is as follows: Section 5.1 introduces the tolerance-based constraint considered in this work; Section 5.2 discusses the fuzzy weight distribution algorithm proposed in this chapter; based on the aforementioned algorithm, Section 5.3 presents a multi-objective genetic algorithm with multiple search directions; Section 5.4 investigates the fitness function consideration; the cases study are given in the Section 5.5; and the Section 5.6 summarizes the chapter.

5.1 Tolerance-based constraint in assembly planning

The influence of tolerance is very important in assembly planning. As discussed in Chapter 4, the allowable manufacturing tolerance can cause the geometric deviations during the assembly process, and the different assembly sequence results in different propagations and accumulations of the geometric deviations. The assembly sequence is not feasible when the accumulated geometric deviation exceeds the limits. Associated with the tolerance is the assembly clearance between the mating features, which can be used to adjust the position of the mating features in the assembly process. So clearance can be used to compensate some of the geometric deviations caused by the tolerance. However, the assembly clearance can also cause the geometric deviations of the mating features in assembly.

As discussed in Chapter 4, the relative assemblability (*RA*) was proposed to evaluate the product assemblability in different assembly sequence. If *RA* >1, then the assemblability of the assembly sequence can be ensured; otherwise, if *RA*<1, the assemblability of the sequence cannot be ensured, and the bigger the *RA*, the better is the assemblability of the sequence. Therefore, *RA* can be used to evaluate the assemblability of different assembly sequences in assembly planning, and it will be used as a constraint when building the fitness function in assembly planning, which will be discussed in the Section 5.4.

The detailed approach to conclude the *RA* of an assembly product in different assembly sequences has been discussed in Chapter 4.

5.2 Genetic search directions with fuzzy weights distribution

Assembly planning is a typical multi-objective optimization problem. Generally, in order to reduce the assembly cost, we need to simultaneously minimize the number of assembly orientation change, number of tool (gripper) change and number of assembly operation change. However, these objectives are usually conflicting in the assembly process. For example, some assembly sequences can reduce the number of assembly orientation change, but they increase the number of tool change at the same time. Generally the existing GA-based approaches build the fitness function by assigning constant weights to different objectives and combine them into one fitness function in an evolution trial. For example, fitness function $Z(x)$ in Eq. (5.1) is defined by assigning weights W_1, W_2 and W_3 to objective functions $f_1(x)$, $f_2(x)$ and $f_3(x)$, respectively. If we cannot get the optimal or near-optimal solutions, the weights may need to be changed to start another evolution trial.

$$Z(x) = W_1 f_1(x) + W_2 f_2 (x) + W_3 f_3 (x) \tag{5.1}$$

The obvious drawback of this approach is that the weights are set arbitrarily by the decision maker, which can affect the successful search for the optimal or near-optimal solutions. In addition, the constant weights determine the search direction, and thus the solution space, affecting the search for other non-dominated solutions.

To overcome the limitation of the above problem, in this section, an algorithm is proposed to derive the fuzzy weights that are used in the multi-objective decision problem. The objective of using the fuzzy weights is to find as many non-dominated solutions as possible, while considering the preference of the weight distribution to different objectives based on the experience of the decision maker.

5.2.1 Non-dominated solutions

Due to the conflicting attributes among the different objectives in assembly planning, there exists a set of solutions in the solution space; in which none of them is superior to the others according to each objective. These solutions are usually called non-dominated solutions or Pareto optimal solutions. These non-dominated solutions can be regarded as the best trade-off solutions in the multi-objective optimization problem.

Generally, the non-dominated solutions can be defined as follows: Given a multi-objective optimization problem with n objectives to be minimized: minimize $f_1(x)$, $f_2(x)$,……$f_n(x)$, $X \in \Omega$, where $f_i(x)$ represents the different objectives, $i \in \{1, 2, \ldots\ldots n\}$, and Ω represents the feasible solution space. For two solutions X_1, X_2, if

$$\begin{cases} f_t(x_1) < f_t(x_2), \text{ for some } t \in \{1, 2, \ldots\ldots n\} \\ f_t(x_1) \leq f_t(x_2), \text{ for all } t \in \{1, 2, \ldots\ldots n\} \end{cases}$$

then solution X_2 is dominated by solution X_1. In the feasible solution space Ω, if there does not exist any solution which can dominate solution X, then we call solution X as non-dominated solution.

In the multi-objective optimization problem, a set of non-dominated solutions form the Pareto frontier. An example is shown in Figure 5.1, where the solid circles represent the non-dominated solutions while the hollow circles represent the dominated solutions. This is a two-objective optimization problem, with the goal to minimize those two objectives, i.e. to search for the non-dominated solutions located along the Pareto frontier.

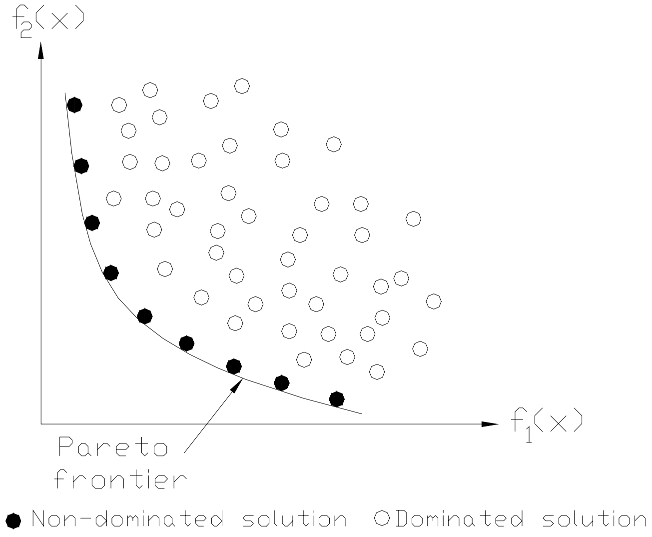


Figure 5.1 Non-dominated solutions in a two-objective optimization problem

5.2.2 Search directions in a multi-objective optimization problem

Existing GA-based assembly planning approaches fix the search direction by using constant weights to build the fitness function. With reference to Figure 5.2, if we set constant weights $W_1 = W_2 =$ some constant value to two objectives $f_1(x)$ and $f_2(x)$, we can get a fixed search direction say d_3. Along this direction, we find the non-dominated solutions X_5 but miss the other non-dominated solutions, such as X_4 or X_6. Because the constant weights are set by the decision maker through experience, it is difficult to set the weights exactly; thus we need to provide more feasible non-dominated solutions to the decision maker for choices.

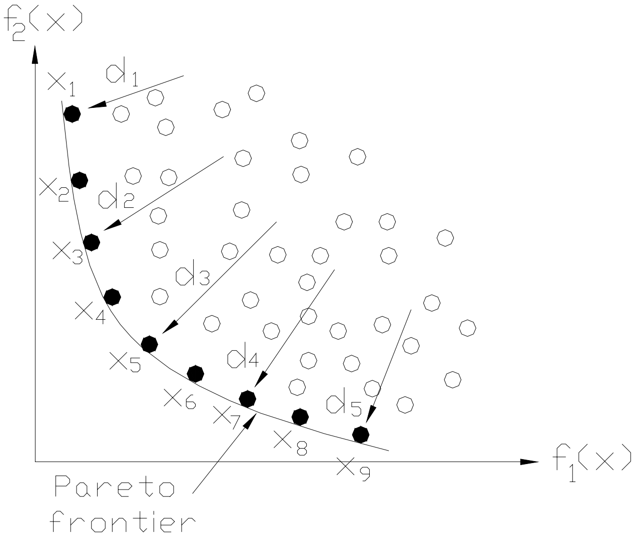


Figure 5.2 Search directions toward Pareto frontier

In order to expand the search directions toward the Pareto frontier and get more non-dominated solutions, some recent works using GA to resolve multi-objective optimization problems can be summarized as follows:

Ishibuchi and Murata [1998] proposed an approach using random weights assigned to different objectives when a pair of parent solutions are selected. The fitness function is then built using the weighted sum of different objectives; thus the fitness functions of each parent solution have different weight vectors. This approach can search multiple random directions toward Pareto frontier. However, because of the randomness of the search directions, it can easily miss some non-dominated solutions. As shown in Figure 5.2, the multiple random search directions may include d_1 and d_5, but may miss directions d_2, d_3 and d_4, and thereby not find the non-dominated solutions X_3, X_4, X_5, X_6 and X_7. In addition, this approach does not consider the preference of different objectives, and so the obtained non-dominated solutions may not suit the desired condition of the optimization problem.

Leung et al. [2000] and Solimanpur et al. [2004] proposed an approach using uniform design to select the weight vector, which is used to build different fitness functions, to enable the search directions to be scattered uniformly toward the Pareto frontier. As shown in Figure 5.2, using this approach, we can search the non-dominated solutions X_1 to X_9 along directions d_1, d_2, d_3, d_4 and d_5. However, this approach does not consider the preference of different objectives either, and therefore some non-dominated solutions found may not be suitable for the optimization problem. This search procedure could cost more time to find more uniformly scattered non-dominated solutions.

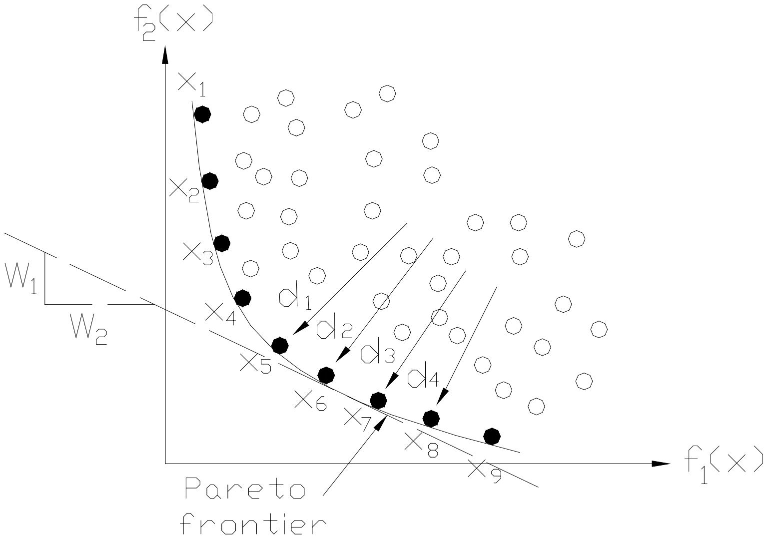


Figure 5.3 Optimized search directions toward Pareto frontier

In view of the aforementioned problems, in this work, we need to develop a new search mechanism, enabling multiple search directions toward Pareto frontier. For example, in a two-dimensional optimization problem, as shown in Figure 5.3, if objective $f_2(x)$ is more important than objective $f_1(x)$, w_2 should be larger than w_1. Then through the tangent point between the Pareto frontier and the line with slope of w_1/w_2, we can find the non-dominated solutions x_6 or x_7 to be the reasonable non-dominated solutions representing the relative importance of the different objectives. However, due to the uncertainty of the weights set by the decision maker, we also need find more non-dominated solutions near to x_6 or x_7, such as x_5 and x_8. Therefore we use the search directions d_1, d_2, d_3 and d_4 toward the Pareto frontier, while the search directions toward other non-dominated solutions, such as x_1 or x_2 can be ignored.

In order to obtain a set of weights that determine suitable search directions towards the Pareto frontier and find more reasonable non-dominated solutions, an algorithm using linear membership functions is proposed to derive the fuzzy weights for the multi-objective optimization problem.

5.2.3 Using linear membership functions to derive the fuzzy weights

In a multi-objective optimization problem, through estimating the relative importance of different objectives according to desired conditions, the decision maker typically makes some rough estimate to the weight that is assigned to each objective based on his judgment and experience to represent the preference of the objective.

In order to deal with the uncertainty in the weight setting and expand the search directions toward the Pareto frontier, this section proposes the use of fuzzy weights to build the fitness function. Firstly, rough weights are given to the objectives by the decision maker to represent the preferences of the respective objectives. Secondly, each weight is given a range that can be set by the decision maker through his experience. The range of each weight sets the boundary for the weight to fluctuate. For example, in the assembly planning of a product consisting of heavy and large parts, the objective related to orientation change should be given larger weight as, it would cost more time or labor to change the orientation of a large and heavy product during the assembly process. Thus the objective related to orientation change can be given a larger weight (e.g. weight =0.6) than the other objectives. And if the range of this weight is set at ± 0.1, then the weight can fluctuate in the range of [0.5, 0.7].

According to the rough weight and weight range set to each objective by the decision maker, the linear membership functions are used to derive the fuzzy weight of each objective, and the weighted sum is used to build a fitness function. The linear membership functions are discussed as follows, with reference to Figure 5.4.

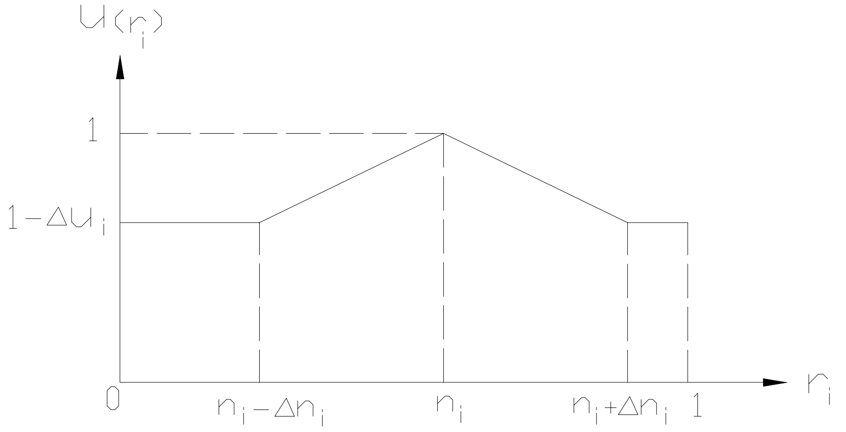


Figure 5.4 Linear membership functions to derive the fuzzy weights

Consider a multi-objective optimization problem with k objectives, where initial weight $w_i = n_i$ ($n_i \in [0,1]$), with a range of $\pm n_i \Delta\mu_i$ is set for objective i ($i \in$ [1, 2,.….,

k]). $\Delta\mu_i \in [0,1]$ decides the maximum fluctuation of the weight w_i. The variable $\mu(r_i)$ is the coefficient of the weight w_i and is decided by a randomly generated number r_i ($r_i \in [0,1]$) and the linear membership functions according to Eq. 5.2. $\Delta n_i \in [0,1]$ sets the probability of fluctuation of w_i. Larger Δn_i, implies easier fluctuation of w_i.

$$\mu(r_i) = \begin{cases} 1-\Delta\mu_i, & \text{if } r_i < n_i - \Delta n_i \\ 1+\Delta\mu_i(r_i - n_i)/\Delta n_i, & \text{if } n_i - \Delta n_i \le r_i < n_i \\ 1, & \text{if } r_i = n_i \\ 1+\Delta\mu_i(n_i - r_i)/\Delta n_i, & \text{if } n_i < r_i \le n_i + \Delta n_i \\ 1-\Delta\mu_i, & \text{if } n_i + \Delta n_i < r_i \end{cases} \tag{5.2}$$

The coefficient $\mu(r_i)$ can represent the fuzzy fluctuation of the weight. Using $\mu(r_i)$, the fuzzy weight w_i can be calculated as follows:

$$w_i = \begin{cases} n_i\,\mu(r_i), & \text{if } r_i < n_i - \Delta n_i \\ n_i\,\mu(r_i), & \text{if } n_i - \Delta n_i \le r_i < n_i \\ n_i, & \text{if } r_i = n_i \\ n_i + n_i[1-\mu(r_i)], & \text{if } n_i < r_i \le n_i + \Delta n_i \\ n_i + n_i[1-\mu(r_i)], & \text{if } n_i + \Delta n_i < r_i \end{cases} \tag{5.3}$$

The algorithm of deriving the fuzzy weight for each objective is as follows:

***Algorithm 1*: Deriving the fuzzy weight of each objective**

For a k-objective optimization problem,

Step (1): Set i=1, $i \in [1, 2, \ldots.., k]$;

Step (2): Set the initial rough value n_i ($n_i \in [0,1]$) for the weight w_i as determined by the decision maker;

Step (3): Set $\Delta\mu_\mathrm{i}$ and Δn_i for the weight w_i by the decision maker, and let $\Delta\mu_i \in [0,1]$, $\Delta n_i \in [0,1]$;

Step (4): Randomly generate a number r_i for the weight w_i, $r_i \in [0,1]$;

Step (5): Derive the coefficient $\mu(r_i)$, according to Eq. 5.2;

Step (6): Derive the fuzzy weight w_i, according to Eq. 5.3;

Step (7): If $i \neq k$-1, then $i = i$+1, go to step (2); else

$w_k = 1- (w_1 + w_2 + \ldots\ldots + w_{k\text{-}1})$, stop.

Using the above algorithm, the weight for each objective can be derived, with the sum of all weights equals to 1.

5.3 Multi-objective genetic algorithm with multiple search directions

In this section, a multi-objective genetic algorithm with multiple search directions is proposed. The multiple search directions are determined by the fuzzy weights derived (according to the algorithm described in section 5.2).

5.3.1 Initial population generation

Firstly, we randomly generate *N* solutions. Then, we adopt *n* search directions by deriving the fuzzy weights to different objectives to build n different fitness functions. According to the fitness function built from an initial set of fuzzy weights, we select *N/n* solutions based on a roulette wheel selection algorithm; i.e. we get a set of *N/n* solutions for the search direction based on the initial set of fuzzy weights. Repeating the above steps, we derive a new set of fuzzy weights and build a new fitness function for the *N* solutions, and obtain another set of *N/n* solutions. The above iterative steps of deriving a new set of fuzzy weights and the corresponding fitness function to search for *N/n* solutions are continued until n sets of solutions are obtained.

Using this approach, we can have n sets of solutions as the initial population. Each set of *N/n* solutions has been obtained by a different search direction in the solution space. The detailed initial population generation algorithm is summarized as follows:

***Algorithm 2*: Initial population generation**

Step (1): Randomly generate N solutions, let $i = 1$;

Step (2): Use Algorithm 1 to derive the fuzzy weight of each objective; then build the corresponding fitness function.

Step (3): Using the fitness function, calculate the selection probability of each solution through the roulette wheel selection algorithm as proposed by Goldberg [1989], i.e.:

$$P(x) = [f(x)-f_{\min}(\Psi)] \,/\, \Sigma\{f(x)-f_{\min}(\Psi)\}, \quad \text{where } x \in \Psi$$

From this selection of probability, we select *N/n* solutions in a set with set number i;

Step (4): If $i < n$, then $i = i+1$, go to step (2); else, stop.

5.3.2 Population evolution

Crossover and mutation are two main operators to realize the population evolution in GA. Through the crossover to a pair of solutions, two new offspring can be generated, which can inherit some properties of their parents. Through mutation to each solution, some new properties can be brought to the offspring, and it can help to overcome the premature convergence.

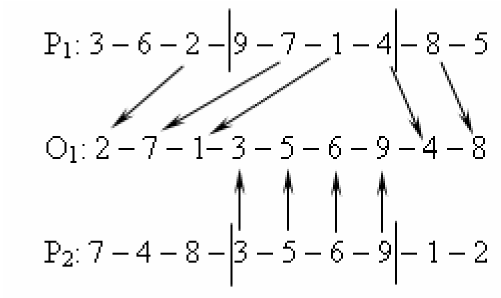


Figure 5.5 Order Crossover procedure

In this work, the Order Crossover (OX) [Michalewicz, 1996] is used as the crossover operator. Consider the two parent solutions P_1 and P_2 with two cut points, as shown in Figure 5.5, when generating the offspring, the gene segment between two cut points is firstly copied from one parent to the offspring. The genes are next copied from another parent in the same sequence to the offspring by deleting the genes that already exist. The crossover process to generate one offspring O_1 is illustrated in Figure 5.5. Using the similar method, the other offspring can also be generated. Using the Order Crossover, the offspring can inherit some properties of their parents, while ensuring that each gene can occur only once in each solution. Thus Order Crossover is suitable for use in assembly planning, where each part occurs only once in a sequence.

After crossover, the mutation operator is applied to the offspring. In this work, we have found experimentally that the insertion mutation operator is suitable for obtaining an assembly sequence without geometric precedence interference. For example, consider the offspring O_1, when applying insertion mutation, one gene is randomly selected and inserted into a position randomly selected in the offspring, as shown in Figure 5.6.

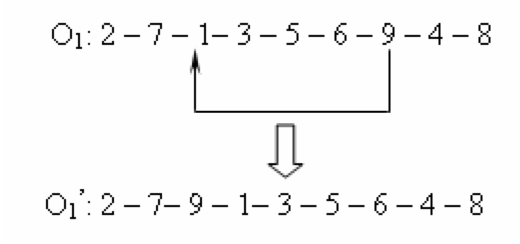


Figure 5.6 Insertion mutation procedure

According to the n solution sets in the initial population generated through Algorithm 2, we generate the new offspring by crossover to each randomly selected pair of solutions in each set under a certain crossover probability, and then evolve the offspring by mutation under a certain mutation probability.

The detailed population evolution algorithm is summarized as follows:

***Algorithm 3*: Population evolution**

Step (1): Let i =1; i is the number of solution sets;

Step (2): For the N/n solutions in set i, randomly select one pair of solutions to crossover under a given crossover probability to generate two offsprings; then apply the mutation to the offspring under a given mutation probability. Using the same method, generate the other offspring from the left pairs of solutions in set i.

Step (3): If $i < n$, then $i = i+1$, go to step (2), else, stop.

5.3.3 Population selection

After the crossover and mutation operations, we use the selection algorithm to select the next new generation from the parent and the offspring solutions generated. The selection algorithm is as follows:

***Algorithm 4*: Population selection**

Step (1): Let $i = 1$; i is number of solution sets;

Step (2): Use Algorithm 1 to derive the fuzzy weight of each objective; then build the corresponding fitness function;

Step (3): Use the fitness function, to calculate the selection probability of the parent solutions and the offspring solutions generated through the roulette wheel selection algorithm [Goldberg, 1989]:

$$P(x) = [f(x) - f_{\min}(\Psi)] / \Sigma\{f(x) - f_{\min}(\Psi)\}, \text{ where } x \in \Psi.$$

From this selection probability, we select the N/n solutions in a set with the set number i;

Step (4): If $i < n$, then $i = i+1$, go to step (2); else, stop.

From the above algorithm, we generate a new generation including n sets of solutions; each set has N/n solutions with the same search direction, with the different set having different search directions from other sets. The overall population size is N.

5.3.4 Overall multi-objective genetic algorithm

The overall multi-objective genetic algorithm with multiple search directions proposed in this work can be summarized as follows:

Step (1): Using algorithm (1) to derive the fuzzy weight for each objective;

Step (2): Using algorithm (2) to generate the initial population;

Step (3): Let $N_{\text{gen}}=1$, where N_{gen} is the generation number;

Step (4): Using algorithm (3) to evolve the population;

Step (5): Using algorithm (4) to select the new generation of populations;

Step (6): If $N_{\text{gen}} < \text{Max}(N_{\text{gen}})$ ($\text{Max}(N_{\text{gen}})$ is given as the maximum generation number), then $N_{\text{gen}} = N_{\text{gen}} + 1$, go to step (4); else, stop.

5.4 Building the fitness function for assembly planning

5.4.1 Objectives in assembly planning

The goal of assembly planning is to minimize the assembly cost or assembly time. Usually the assembly cost or time can be determined by the number of assembly orientation changes, tool changes and changes in assembly operation types. In the case of automated assembly system, robots are used to grasp and place the parts with assembly tools. The selection of the assembly tool is determined by the geometric shape, dimension, weight, etc. of the part, and generally can include magnetic, vacuum, adhesive grippers, etc. [Delchambre, 1992]. A change of the assembly orientation or assembly tool incurs time and usually can increase the assembly cost. Different types of assembly operations are needed to complete the assembly process, such as pressing, screwing, riveting, etc. Changes of the assembly operations can also require tool change, and thus increase the assembly time and cost.

Hence, in assembly planning, the above three objectives pertaining to assembly orientation changes, tool changes and changes in assembly operation types should be minimized to reduce the assembly time and cost.

5.4.2 Constraints for feasibility evaluation of the assembly sequence

In this work, two constraints are used to evaluate the feasibility of the assembly sequence. The first is precedence constraint, which is used to avoid collision during the assembly process. The second is tolerance-based constraint. The latter is used to evaluate the product assemblability, which considers the propagation and accumulation of geometric deviations caused by tolerance and clearance in a given assembly sequence.

5.4.2.1 Using interference matrix for precedence feasibility evaluation and determination of assembly orientation changes

The interference matrix was first proposed by Dini et al. [1992] in assembly planning, and can be derived from the geometric assembly relationship. An interference matrix I_d (d represents the assembly direction) for an assembly consisting of n parts can be represented as follows:

$$I_d= \begin{array}{c|cccc|} & P_1 & P_2 & \cdots\cdots & P_\mathrm{n} \\ P_1 & P_{11} & P_{12} & \cdots\cdots & P_{1n} \\ P_2 & P_{21} & P_{22} & \cdots\cdots & P_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ P_\mathrm{n} & P_{n1} & P_{n2} & \cdots\cdots & P_{nn} \end{array}$$

P_1, ….. P_n represent the n parts in the assembly, $P_{ij}=1$ ($i\in[1, n], j\in[1, n]$) if part P_i collides with P_j when P_i is moved along the direction d to the assembly position; otherwise $P_{ij}=0$. $P_{ii}= 0$, as the part cannot collide with itself. Because P_{ij} in the $-d$ direction is equal to P_{ji} in the $+d$ direction, we can use three interference matrices $I_{+X,}$ $I_{+Y,}$ I_{+Z} to conclude the precedence feasibility of the given assembly sequence (Generally, in large scale automated assembly system, the six axes- $\pm X$, $\pm Y$, $\pm Z$ are the principal axes along which the components are assembled to the position, so we only consider these six directions in this work. In practice, the feasible assembly directions may be fewer without considering other assembly directions, such as through rotation about one or more of the X, Y, and Z axes. These other directions will be considered in the future work.).

For a given assembly sequence, when part P_i is assembled after a subassembly S_m consisting of m parts, then the feasible assembly direction of part P_i to S_m can be derived as follows:

For assembly direction d, $d\in\{\pm X, \pm Y, \pm Z\}$, let $P_j\in S_m$, determine $D_d(P_i\ S_m)= \sum P_{ij}$ (P_{ij} is the element in I_d). If $D_d(P_i\ S_m)=0$, then direction d is the feasible assembly direction of part P_i to S_m; otherwise, direction d is infeasible. If none of the six directions is feasible, then the assembly sequence is infeasible; otherwise, the assembly sequence is precedence feasible for part P_i to S_m.

Using the above approach, only when the assembly sequence is precedence feasible for all parts P_i ($i \in [1, n]$), the assembly sequence is precedence feasible; otherwise, this sequence should be discarded.

Using the interference matrix, we can also determine the number of assembly orientation changes in a precedence feasible assembly sequence. For a feasible assembly sequence consisting of n parts, we can get the feasible assembly directions D'_i ($i \in [1, n]$) for part P_i to subassembly S_m which has been assembled before part P_i. For an assembly sequence P_1, P_2, P_3, ……., P_n, the number of assembly orientation changes can be determined as follows:

Step (1): Let i=1, q=1, w=0, w represents the number of assembly orientation changes;

Step (2): If $D'_i \cap D'_{i+1} \cap \ldots\ldots \cap D'_{i+q} \neq \emptyset$, but $D'_i \cap D'_{i+1} \cap \ldots\ldots \cap D'_{i+q} \cap D'_{i+q+1} = \emptyset$, then assembling part P_{i+q+1} needs an orientation change, let $w = w+1$, go to step (3); otherwise, $q=q+1$, and if $i+q+1<n$, then reiterate step (2); else, go to step (4);

Step (3): Let $i = i+q+1$, q=1, if $i < n$, go to step (2); otherwise, go to step (4)

Step (4): End.

In the following, the above approaches are illustrated by a simple example of an assembly consisting of seven parts, as shown in Figure 5.7.

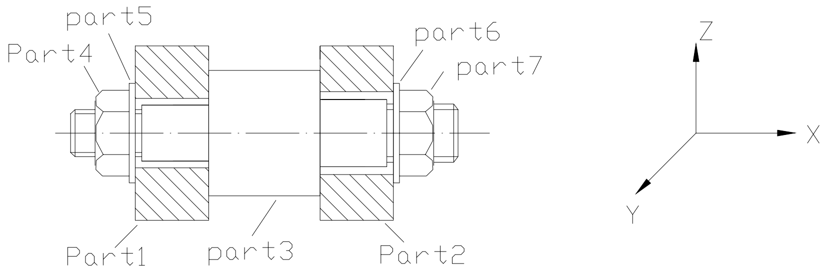


Figure 5.7 Assembly consists of 7 parts

The three interference matrices for this assembly are derived as follows:

$$I_{+X}=\begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 2 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 3 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 6 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 7 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{array} \qquad I_{+Y}=I_{+Z}=\begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 3 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 4 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 6 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 7 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array}$$

For a given assembly sequence: 1-3-4-5-2-6-7, when assembling part 4 to subassembly S_2 consisting of part 1 and part 3, we calculate $D_{+X}(P_4\,S_2) = P_{4\,1}(\mathrm{I}_{+X}) + P_{4\,3}(I_{+X}) = 0$, $D_{-X}(P_4\,S_2) = P_{4\,1}(I_{-X}) + P_{4\,3}(I_{-X}) = 1+1 \neq 0$, $D_{+Y}(P_4\,S_2) = P_{4\,1}(I_{+Y}) + P_{4\,3}(I_{+Y}) = 0+1 \neq 0$, and $D_{-Y}(P_4\,S_2) \neq 0$, $D_{+Z}(P_4\,S_2) \neq 0$, $D_{-Z}(P_4\,S_2) \neq 0$, then the feasible assembly direction of part 4 to subassembly S_2 is $+X$. When assembling part 5 to subassembly S_3 consisting of part 1, part 3 and part 4, we calculate $D_d(P_5\,S_3) \neq 0$, for all six directions $d \in \{\pm X, \pm Y, \pm Z\}$. Therefore, this assembly sequence is infeasible.

For another assembly sequence: 1-5-3-2-6-7-4, we can calculate the feasible assembly directions D'_i as follows ($i \in [1, 7]$):

D'_1 for part 1: $\{\pm X, \pm Y, \pm Z\}$;

D'_2 for part 5 to part 1: $\{+X, \pm Y, \pm Z\}$;

D'_3 for part 3 to subassembly S_2 consisting of part 1 and part 5: $\{-X\}$;

D'_4 for part 2 to subassembly S_3 consisting of part 1, part5 and part 3: $\{-X\}$;

D'_5 for part 6 to subassembly S_4 consisting of part 1, part 5, part 3 and part 2: $\{-X\}$;

D'_6 for part 7 to subassembly S_5 consisting of part 1, part 5, part 3, part 2 and part 6: $\{-X\}$;

D'_7 for part 4 to subassembly S_6 consisting of part 1, part 5, part 3, part 2, part 6 and part 7: $\{+X\}$;

Obviously this sequence is feasible considering the precedence constraint. In this sequence, because $D'_1 \cap D'_2 \neq \emptyset$, but $D'_1 \cap D'_2 \cap D'_3 = \emptyset$, so one assembly orientation change is needed to assemble part 3; because $D'_3 \cap D'_4 \cap D'_5 \cap D'_6 \neq \emptyset$, but $D'_3 \cap D'_4 \cap D'_5 \cap D'_6 \cap D'_7 = \emptyset$, then another assembly orientation change is needed to assemble part 7. Therefore, there are totally two assembly orientation changes in this sequence.

5.4.2.2 Tolerance-based constraint in assembly planning

After an assembly sequence is generated by the genetic operators, the relative assemblability (*RA*) of the sequence can be concluded using the approach introduced in chapter 4. The *RA* of the assembly sequence is used as a constraint to ensure that the assembly sequence is feasible, by avoiding the assembly sequence with significant accumulation of geometric deviations caused by the tolerance and clearance during the assembly process.

In this work, if $RA \geq 1$, the assembly sequence can be considered feasible if we only consider the tolerance-based constraint. So if $RA \geq 1$, we let $RA = 1$, then $RA \in (0, 1]$.

5.4.3 Formulation of the fitness function

As mentioned earlier, to build the fitness function for assembly planning, three objectives are used: number of assembly orientation change, number of tool change and number of assembly operation type change. These are considered as follows:

N_{or} — Number of assembly orientation change;

N_t — Number of assembly tool change;

N_{op} — Number of assembly operation type change;

For a given assembly sequence, N_{or} can be calculated using the approach described in Section 5.4.2.1, N_t and N_{op} can be easily derived if we assign the assembly tool and assembly operation type for each part in advance.

Besides the above three objectives, two constraints, the precedence constraint and the tolerance-based constraint, are used in the fitness function. The precedence constraint is used to evaluate the precedence feasibility, as mentioned in Section 5.4.2.1, and the tolerance-based constraint is used to evaluate the RA of the assembly sequence, as mentioned in Section 5.4.2.2.

For an assembly consisting of m parts, given an assembly sequence, if the sequence is infeasible considering the precedence constraint, i.e. the assembly interference number $\geqslant 1$, then the fitness function is given by:

$$F = (2m - W_1 N_{or} - W_2 N_t - W_3 N_{op}) * RA / (2Z); \tag{5.4}$$

where, the W_1, W_2, W_3 are the weights derived using Algorithm 1. $RA \in (0,1]$ represents its relative assemblability. Z represents the assembly interference number in the given assembly sequence. If the assembly sequence is also infeasible considering the tolerance constraint, i.e. $RA \in (0, 1)$, then in this fitness function, we use RA and Z to penalize the infeasible solution, while allowing the infeasible solutions to be selected and evolved under a low probability. This serves to avoid pre-mature convergence. If the assembly sequence is feasible considering the tolerance-based constraint, i.e. $RA = 1$, then only Z will be used to penalize this infeasible solution.

In another condition, if the given assembly sequence is feasible considering the precedence constraint, i.e. $Z = 0$, then the fitness function is given as follows:

$$F = (2m - W_1 N_{or} - W_2 N_t - W_3 N_{op}) * RA \tag{5.5}$$

In this condition, if the given assembly sequence is infeasible considering the tolerance-based constraint, i.e. $RA \in (0, 1)$. RA is used to penalize this solution, allowing the solution with a larger RA to be selected with a higher probability.

Finally, if the assembly sequence is feasible considering both precedence and tolerance constraints, i.e. $RA=1$, $Z=0$, then the fitness function is given as:

$$F = 2m - W_1 N_{or} - W_2 N_t - W_3 N_{op} \tag{5.6}$$

The evolving steps of the fitness function for a solution are illustrated in Figure 5.8.

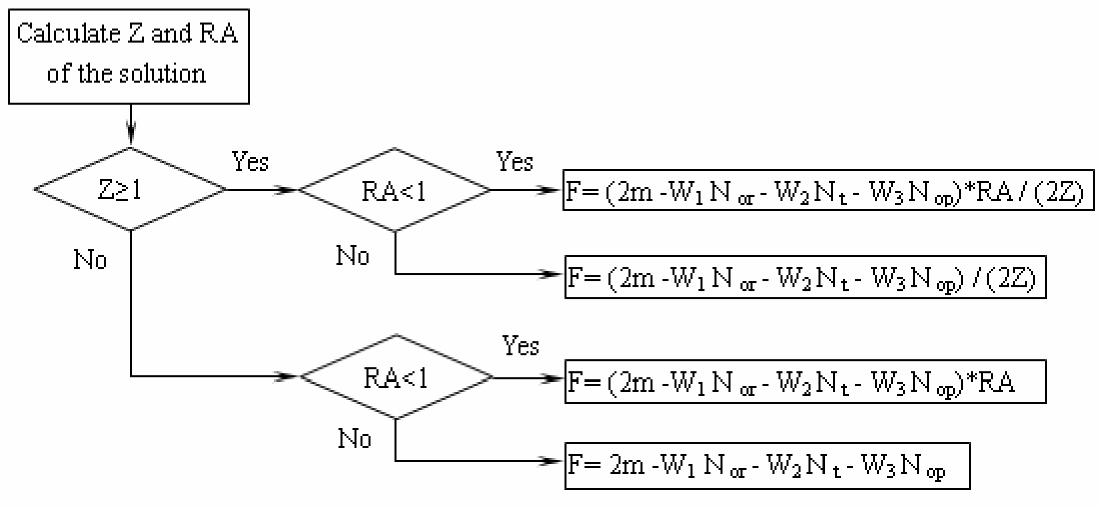


Figure 5.8 Evolving steps of the fitness function for a solution

5.5 Case study

The assembly planning system with the proposed approaches has been implemented using Visual C++ 6.0. In this section, two cases are used to validate the proposed approaches.

5.5.1 Case study 1

Case 1 (Figure 5.9) shows an assembly product consisting of 22 parts. Using the proposed approaches, the test steps and results are introduced as follows.

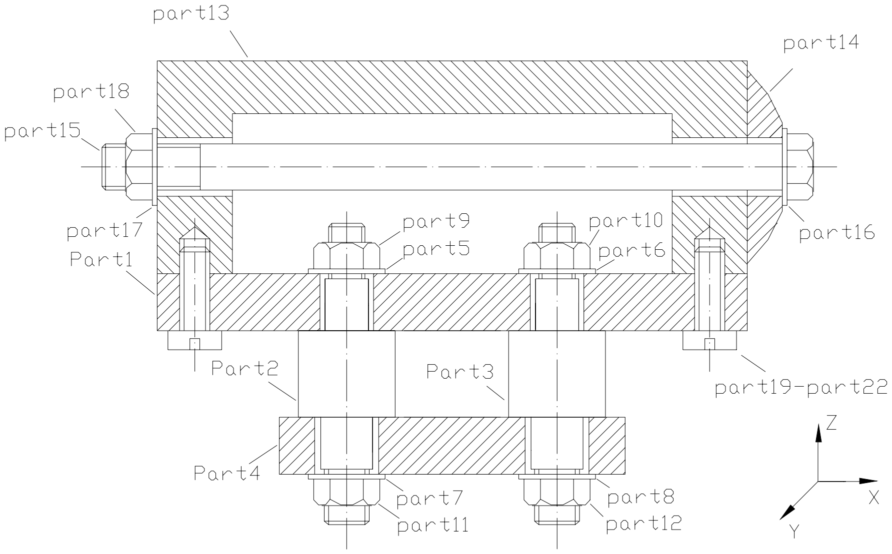
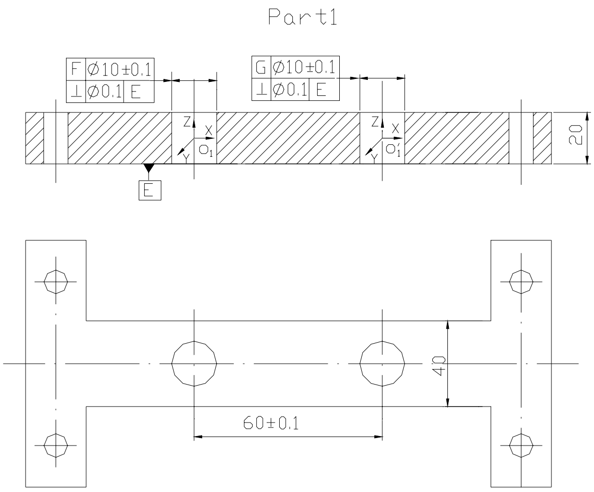


Figure 5.9 An assembly consisting of 22 parts

Step 1: Calculation of Relative Assemblability of each assembly sequence

In this case, the inappropriate assembly sequence for part 1 to part 12 may result in the assembly interference due to the influence of tolerance and clearance. Actually, only the dimensional and geometric tolerance of part 1, part 2, part 3, and part 4 can result in the assembly interference in some assembly sequences. Therefore, we need only consider the dimensional and geometric tolerance of these four parts, as shown in Figure 5.10, Figure 5.11, Figure 5.12 and Figure 5.13, respectively. The relative assemblability (*RA*) in different assembly sequences can be calculated using the approach presented in Chapter 4.



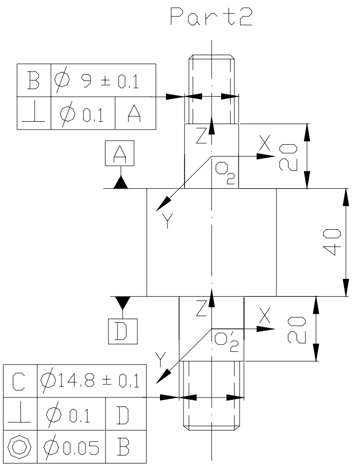
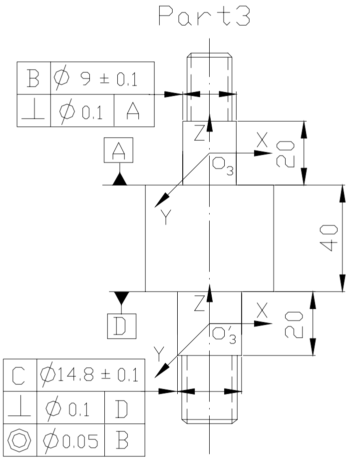


Figure 5.10 Part 1 **Figure 5.11 Part 2**



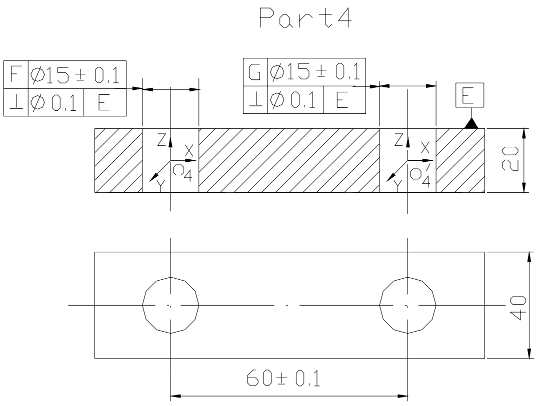
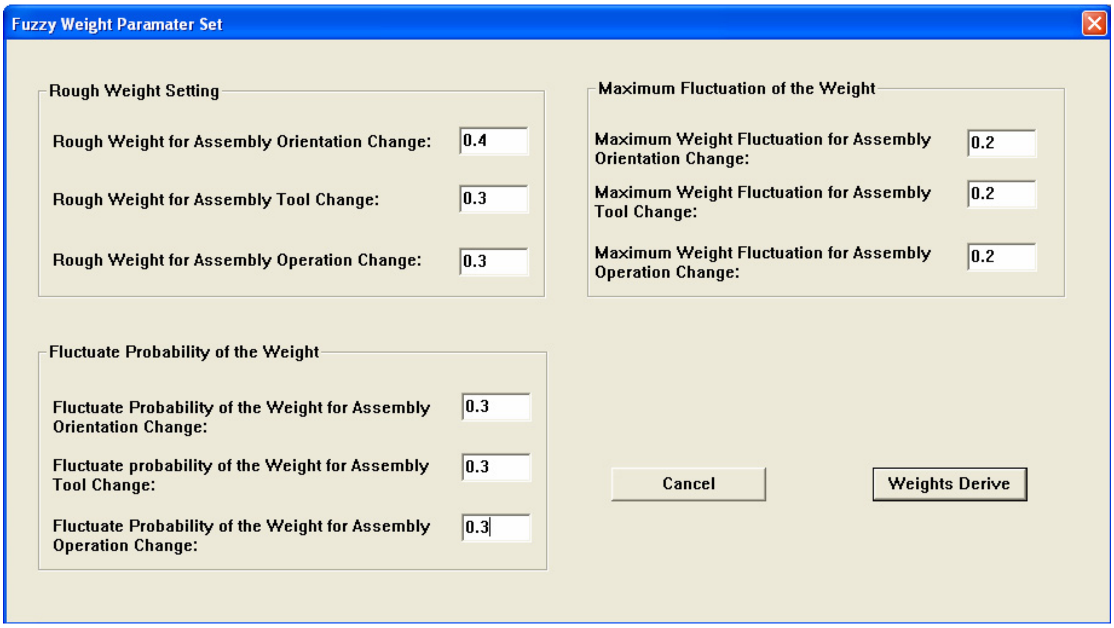


Figure 5.12 Part 3 **Figure 5.13 Part 4**

Step 2: Derivation of fuzzy weight for each objective

Figure 5.14 shows the dialog box for the decision maker to input the fuzzy weight parameters. Through this dialog box, the decision maker inputs the rough weights, maximum fluctuation of the weight, and fluctuation probability of the weight for each of the three objectives based on his experience and the desired working condition. These parameters will be used to derive the fuzzy weight of each objective, including assembly orientation change, assembly tool change and assembly operation change, as discussed in section 5.2.3. In practice, the decision maker only needs to input the weight parameters for any two objectives, the weight of the third objective can be calculated from the equation $W_1+W_2+W_3=1$.



Fuzzy Weight Paramater Set

Rough Weight Setting

Rough Weight for Assembly Orientation Change: 0.4

Rough Weight for Assembly Tool Change: 0.3

Rough Weight for Assembly Operation Change: 0.3

Maximum Fluctuation of the Weight

Maximum Weight Fluctuation for Assembly Orientation Change: 0.2

Maximum Weight Fluctuation for Assembly Tool Change: 0.2

Maximum Weight Fluctuation for Assembly Operation Change: 0.2

Fluctuate Probability of the Weight

Fluctuate Probability of the Weight for Assembly Orientation Change: 0.3

Fluctuate probability of the Weight for Assembly Tool Change: 0.3

Fluctuate Probability of the Weight for Assembly Operation Change: 0.3

Figure 5.14 Fuzzy weight parameter input dialog box

Step 3: Selection of population size, number of sets, and parameters of genetic operators

In this case, we adopt the population size N=80, number of sets n=10, then each set includes 8 solutions. Each generation will have 10 search directions toward the Pareto frontier determined by the fuzzy weights derived. For the parameters of genetic operators, crossover probability P_c=0.8, mutation probability P_m= 0.2, and maximum generation numbers N_{Gen}= 500.

Step 4: Evolution test and results

In this case, the number of tool (gripper) types and operation types of each part in the assembly are given in Table5.1.

Table 5.1 Tool type and operation type of each part in the assembly

Part No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Tool type	1	2	2	1	3	3	3	3	4	4	4	4	1	5	2	3	3	4	6	6	6	6
Operation type	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	1	2	2	2	2

According to the existing assembly planning system with GA, when considering a complex assembly with many precedence constraints, the initial population is generally required to include one or more feasible precedence assembly sequences [Guan et al., 2002, Senin et al., 2000]. Otherwise, the randomly generated initial population usually cannot converge to a precedence feasible sequence.

In our test, we adopt the randomly generated initial population by the assembly planning system, and use two different weight distribution schemes: traditional constant weights distribution and the fuzzy weight distribution as proposed in this work.

(1) Test 1:

In test 1, we adopt traditional constant weight distribution scheme, setting W_1=0.4 W_2=0.3, W_3=0.3, and the GA parameters are listed in Table 5.2.

Table 5.2 GA parameters in test 1

GA Parameters	Population size	Crossover probability	Mutation probability	Maximum generation
Value	80	0.8	0.2	500

For the test results of 20 trials, the solutions in 15 trials converge to feasible sequences, and the solutions in 5 other trials converge to infeasible sequences. During the 15 trials with convergence to feasible sequences, only 2 trials get more than 1 non-dominated solution, as shown in Table 5.3.

Table 5.3 20 trial results in Test 1

Total trials	**Feasible trials**	**Infeasible trials**	**Trials with more than 1 non-dominated solutions**	**Non-dominated solutions found in each trial**
20	15	5	2	1-2

The non-dominated solutions found in a trial are shown in Table 5.4.

Table 5.4 Test results of a trial in Test 1

Non-dominated solution No.	**Interference number**	**Relative assemblability**	**Orientation changes**	**Tool changes**	**Operation changes**
1	0	1	7	6	5
2	0	1	5	6	6

(2) Test 2:

In test 2, we adopt the proposed fuzzy weight distribution scheme, with the GA parameters listed in Table 5.5, and the fuzzy weight parameters listed in Table 5.6.

Table 5.5 GA parameters in test 2

GA Parameters	Population size	Set number	Crossover probability	Mutation probability	Maximum generation
Value	80	10	0.8	0.2	500

Table 5.6 Fuzzy weight parameters in test 2

Fuzzy weight Parameters	Rough weight			Maximum fluctuation			Fluctuate probability		
	n_1	n_2	n_3	$\Delta\mu_1$	$\Delta\mu_2$	$\Delta\mu_3$	Δn_1	Δn_2	Δn_3
Value	0.4	0.3		0.3	0.3		0.3	0.3	

The test results of 20 trials are shown in Table 5.7.

Table 5.7 20 trial results for fuzzy weights setting with $\Delta\mu_1 = \Delta\mu_2 = 0.3$

Total trials	**Feasible trials**	**Infeasible trials**	**Trials with more than 1 non-dominated solutions**	**Non-dominated solutions found in each trial**
20	16	4	5	1-2

The non-dominated solutions found in a trial are shown in Table 5.8.

Table 5.8 Test results of a trial in Test 2

Non-dominated solution No.	**Interference number**	**Relative assemblability**	**Orientation changes**	**Tool changes**	**Operation changes**
1	0	1	5	7	4
2	0	1	4	7	5

(3) Test 3:

In test 3, we adopt the fuzzy weight distribution scheme, setting the fuzzy weight parameter $\Delta\mu$=0.5 (as shown in Table 5.9), and using the same GA parameters in Table 5.5.

Table 5.9 Fuzzy weight parameters in test 3

Fuzzy weight Parameters	Rough weight			Maximum fluctuation			Fluctuate probability		
	n_1	n_2	n_3	$\Delta\mu_1$	$\Delta\mu_2$	$\Delta\mu_3$	Δn_1	Δn_2	Δn_3
Value	0.4	0.3		0.5	0.5		0.3	0.3	

The test results of 20 trials are shown in Table 5.10.

Table 5.10 20 trial results for fuzzy weights setting with $\Delta\mu_1 = \Delta\mu_2 = 0.5$

Total trials	**Feasible trials**	**Infeasible trials**	**Trials with more than 1 non-dominated solutions**	**Non-dominated solutions found in each trial**
20	18	2	11	1-3

The non-dominated solutions found in a trial are shown in Table 5.11.

Table 5.11 Test results of a trial in Test 3

Non-dominated solution No.	**Interference number**	**Relative assemblability**	**Orientation changes**	**Tool changes**	**Operation changes**
1	0	1	4	6	7
2	0	1	2	8	9
3	0	1	3	7	9

(4) Test 4:

In test 4, we also adopt the fuzzy weight distribution scheme, setting the fuzzy weight parameter $\Delta\mu$=0.8 (as shown in Table 5.12), and using the same GA parameters in Table 5.5.

Table 5.12 Fuzzy weight parameters in test 4

Fuzzy weight Parameters	Rough weight			Maximum fluctuation			Fluctuate probability		
	n_1	n_2	n_3	$\Delta\mu_1$	$\Delta\mu_2$	$\Delta\mu_3$	Δn_1	Δn_2	Δn_3
Value	0.4	0.3		0.8	0.8		0.3	0.3	

The test results of 20 trials are shown in Table 5.13.

Table 5.13 20 trial results for fuzzy weights setting with $\Delta\mu_1 = \Delta\mu_2 = 0.8$

Total trials	**Feasible trials**	**Infeasible trials**	**Trials with more than 1 non-dominated solutions**	**Non-dominated solutions found in each trial**
20	18	2	16	1-4

The non-dominated solutions found in a trial are shown in Table 5.14.

Table 5.14 Test results of a trial in Test 4

Non-dominated solution No.	Interference number	Relative assemblability	Orientation changes	Tool changes	Operation changes
1	0	1	5	6	3
2	0	1	2	7	5
3	0	1	4	7	3
4	0	1	3	6	5

The evolution convergence performance of the above 4 tests are given as follows. For test 1, we record the fitness value of the best solution in different generations in a feasible trial. For test 2, test 3 and test 4, we record the fitness value of the best solution in set No.1 in different generations in a feasible trial, respectively. The results are shown in Figure 5.15, where curves 1, 2, 3, 4 represent the fitness value in different generations in the above four tests, respectively.

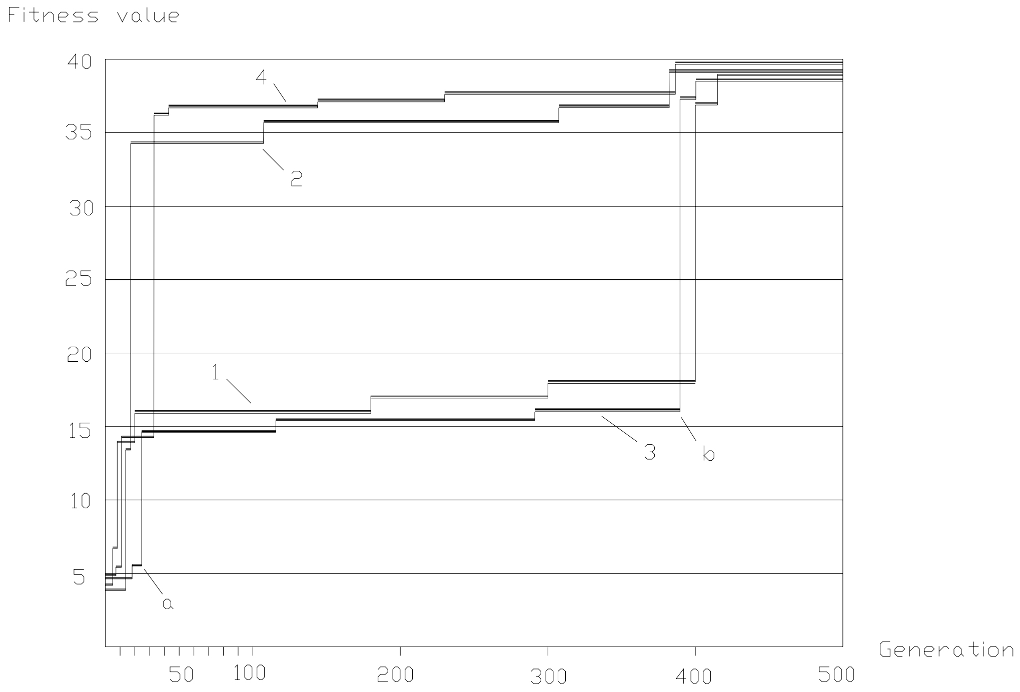


Figure 5.15 Fitness value in different generations in four tests

From the above tests, the evolution results are the feasible non-dominated solutions. For example, the evolution result in non-dominated solution no.1 in Test 2 is given as follows:

4-2-3-1-5-6-8-7-16-17-10-9-13-14-15-18-11-12-21-20-19-22

This sequence has the orientation change number 5, tool change number 7, operation change number 4, and the fitness value is 38.82. This assembly sequence is feasible according to the geometric precedence constraint and the tolerance-based constraint.

5.5.2 Case study 2

Case 2 [Li et al, 2003] is an industrial drive assembly consisting of 21 parts, as shown in Figure 5.16.

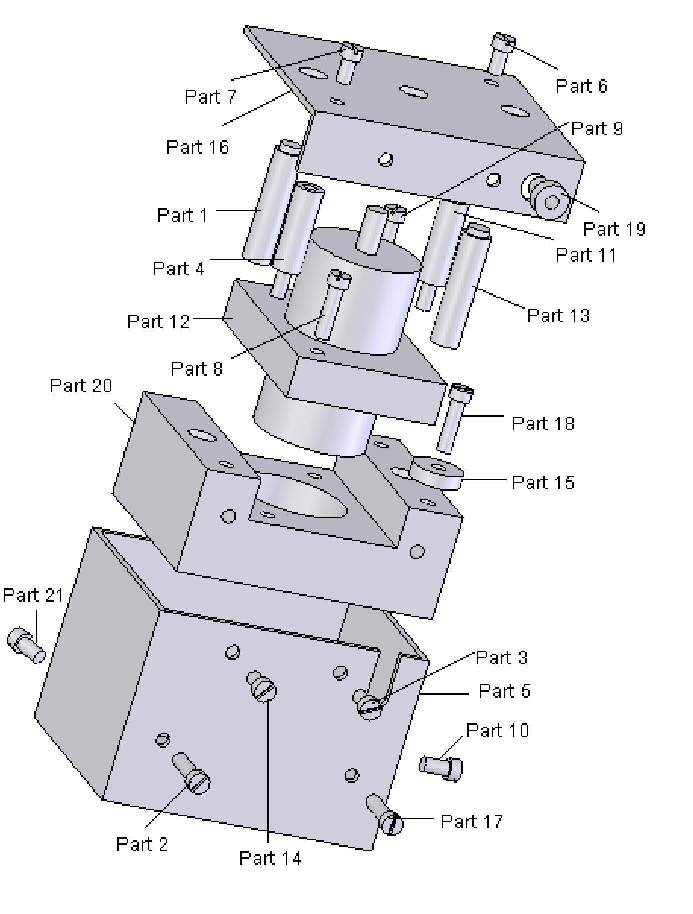


Figure 5.16 A drive assembly consisting of 21 parts

The tolerance design of Part 5, Part 20, Part 12, Part 16, Part 4 and Part 1, which may influence the *RA* of the product in different assembly sequences, is given, as shown in Figure 5.17, Figure 5.18, Figure 5.19, Figure 5.20, Figure 5.21 and Figure 5.22, respectively.

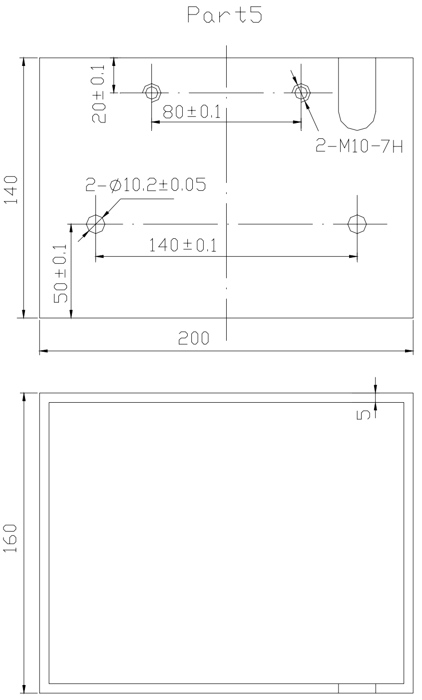


Figure 5.17 Part 5

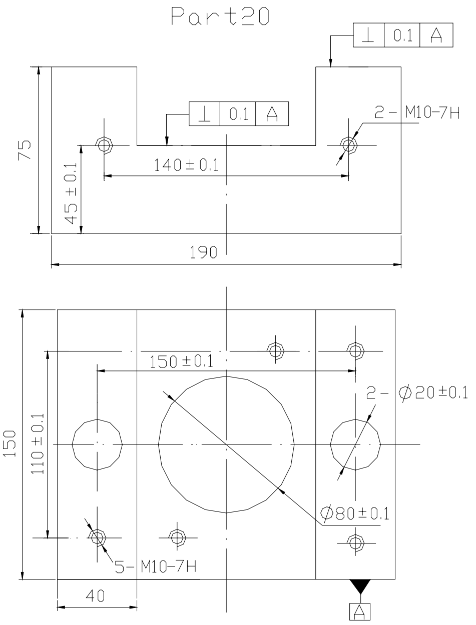


Figure 5.18 Part 20

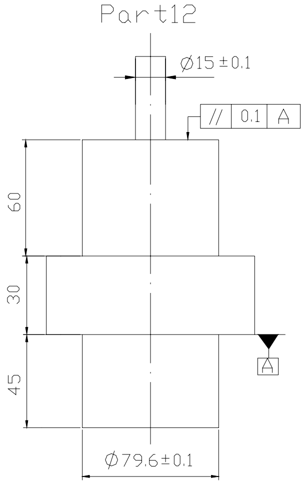


Figure 5.19 Part 12

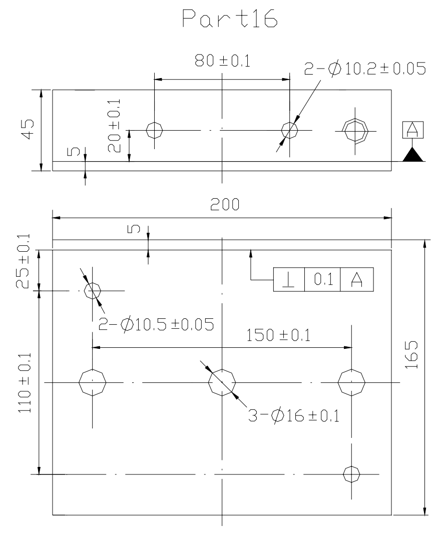


Figure 5.20 Part 16

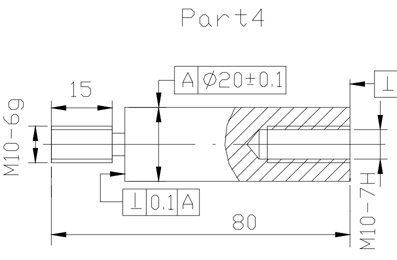


Figure 5.21 Part 4

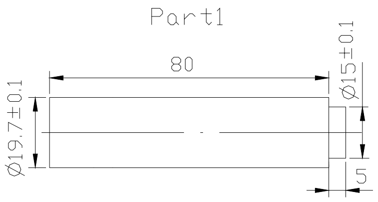


Figure 5.22 Part 1

The number of tool (gripper) types and operation types of each part in the assembly are given in Table 5.15.

Table 5.15 Tool type and operation type of each part in the assembly

Part No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Tool type	5	6	6	5	1	6	6	6	6	6	5	3	5	6	5	4	6	6	6	2	6
Operation type	0	1	1	2	0	1	1	1	1	1	2	0	0	1	0	0	1	1	2	0	1

We also adopted the four test schemes respectively, as that described in case study 1, and the results for each test are shown in Table 5.16.

Table 5.16 20 trial results for four tests

Test No.	**Total trials**	**Feasible trials**	**Infeasible trials**	**Trials with more than 1 non-dominated solutions**	**Non-dominated solutions found in each trial**
1	20	17	3	0	1
2	20	19	1	6	1-2
3	20	20	0	11	1-2
4	20	20	0	17	1-3

The non-dominated solutions found in a trial in Test 4 are shown in Table 5.17.

Table 5.17 Test results of a trial in Test 4

Non-dominated solution No.	Interference number	Relative assemblability	Orientation changes	Tool changes	Operation changes
1	0	1	5	7	6
2	0	1	6	7	5
3	0	1	6	6	6

where, the non-dominated solution No.1 in Test 4 is given as follows: 20 -12-15-18-8-9-4-11-13-1-16-5-17-2-14-3-19-21-10-7-6, it is a feasible non-dominated solution, with the orientation change number 5, tool change number 7, and operation change number 6. This assembly sequence is feasible according to the geometric precedence constraint and the tolerance-based constraint.

5.5.3 Discussions

In each case study, the different tests show the different evolution results. Compared to the conventional assembly planning approach using GA with constant weights, the evolution approach with the fuzzy weight distribution algorithm can find more feasible non-dominated solutions easily, and this can provide the decision maker more choices on the assembly process. In the evolution approach with fuzzy weight distribution, through increasing the maximum fluctuation magnitude of each weight, more feasible non-dominated solutions can be found in a trial. This phenomenon may be analyzed as follows: in a multi-objective genetic evolution, the weights with larger fluctuation can expand the search direction more widely towards Pareto-frontier, and thus can help to find more non-dominated solutions. However, the fluctuation magnitude of each weight should be set by the decision maker considering the desired condition of the assembly product and the assembly process.

As shown in Figure 5.15, different tests show different convergence performances. Some evolution trials cost more time to find feasible sequences, such as Test 1 and Test 3; some evolution trials can find feasible sequences rapidly, and the fitness value increased quickly, such as Test 2 and Test 4. The difference in convergence performance in the above tests is probably due to the randomness of the initial population generated in the computer, and the stochastic property of the GA evolution process. In addition, in each evolution test, there are two significant changes of the fitness value, such as at points *a* and *b* on curve 3, as shown in Figure 5.15. Because we set two constraints in the fitness function, the tolerance-based constraint and the geometric precedence constraint, then the fitness value would have a significant increase when the sequence is evolved from an infeasible sequence to a feasible sequence according to anyone of the above two constraints. Therefore, those two points such as *a* and *b* on curve 3 may be analyzed as the transition points where the sequence is evolved from an infeasible sequence to a feasible sequence according to the two constraints, respectively.

5.6 Summary

This chapter presents an enhanced assembly planning approach using a multi-objective genetic algorithm. In assembly planning, the influence of tolerance and clearance on product assemblability in different assembly sequences is firstly considered, and used as a constraint in assembly planning, resulting in more feasible assembly planning solutions.

Due to the uncertainty in weight setting, and to provide the decision maker greater choice on the final assembly process, a fuzzy weight distribution algorithm is proposed. Based on the fuzzy weight distribution, a multi-objective genetic algorithm with multiple search directions has been developed. This approach builds the different fitness functions through the fuzzy weights derived, which direct the search towards the Pareto frontier in the different directions in an evolution process, allowing more non-dominated solutions to be found while considering the preference of the weight distribution to different objectives by the decision maker.

Through the cases study and evolution tests, the validity of the proposed assembly planning approach has been verified.

Chapter 6 Evaluation of Assembly Design from Assembly Planning and Redesign Suggestion

This chapter discusses the potential design problems which can be identified through the evaluation of the assembly planning results, and further proposes redesign guidelines to help the designer to make the proper design modification or redesign considering the detailed assembly process in the design stage. Section 6.1 discusses the potential design problems identified from the assembly planning results; Section 6.2 proposes the overall redesign guidelines from the assembly planning results; Section 6.3 is the summary of the chapter.

6.1 The design problems identified from the assembly planning results

One of the objectives of assembly planning is to identify possible design problems by evaluating the assembly planning results, and help the designer to make necessary design modification or redesign in the design stage. The assembly planning results, including the different non-dominated assembly sequences, the corresponding assembly interference number, relative assemblability, and the different objective value, including the number of assembly orientation changes, number of assembly tool changes, and number of assembly operation type changes of each non-dominated assembly sequence, can be provided to the designer for design evaluation. From these assembly planning results, some design problems would be identified.

From the assembly planning results, the designer can evaluate several design issues as follows:

1) The feasibility of geometry and dimension design of each part for assembly process
2) The feasibility of tolerance distribution in each part for assemblability during the assembly process
3) The difficulty of the whole assembly process

From the above evaluation, the designer can make some design modification or redesign to improve the product assemblability and reduce the assembly cost or time at the design stage.

In the following, the case 1 in Chapter 5 (shown in Figure 5.9) is used to illustrate the identification of potential design problems through the evaluation of the assembly planning results.

Through assembly planning using GA-based assembly planning approach proposed in Chapter 5, the assembly planning results of the product in an evolution trial can be given, as shown in Table 5.14.

There are four non-dominated solutions. In each non-dominated solution, the relative assemblability, assembly interference number, and three objective values-orientation change number, tool change number and operation change number are displayed.

From the assembly planning results, the designer can select an assembly sequence according to the relative assemblability, assembly interference number, and three objective values. The relative assemblability and assembly interference number are used to judge the feasibility of the assembly sequence. Only when the relative assemblability is 1 and the assembly interference number is 0, the assembly sequence is feasible and can be used; otherwise, it should be discarded. The three objective values- orientation change number, tool change number and operation change number will be determined by the assembly sequence selected by the designer.

In the selection of the assembly sequence, the designer should consider the detailed condition of the part design and the working requirement during the assembly process. For example, if the parts are very big and heavy or the working condition requires the fewer orientation changes, the designer can consider the assembly sequence with fewer number of assembly orientation change. From the assembly planning results shown in Table 5.14, the designer can select the non-dominated solution No. 2 – the assembly sequence with two assembly orientation changes. In another condition, if the designer hopes to reduce the number of assembly operation changes due to the working requirement, he can consider to select the non-dominated solution No. 1 – the assembly sequence with three assembly operation changes. However, if the designer hopes to select an assembly sequence with the fewer number of assembly orientation change and the number of assembly operation change simultaneously, he cannot find an ideal sequence from the assembly planning results. Therefore, some design modification or redesign works need to be done on some parts to realize the above objective. Besides the objectives- orientation change number and operation change number, the tool change number in the four non-dominated solutions are all between 6 to 7. If the designer hopes to reduce this number, some design modifications or redesign works need to be done no matter what assembly sequence is selected.

From the above, it is shown that the potential design problems can be identified through the evaluation of the assembly planning results, and the design modification or redesign of the parts should be made appropriately based on the assembly planning results according to the detailed condition of the parts and working requirement. The design modification or redesign in the design stage will then become more practical considering the assembly process represented by the detailed assembly sequence selected by the designer.

6.2 The overall redesign guidelines from the assembly planning results

In this section, the overall redesign guidelines are proposed according to the design problems identified through the evaluation of the assembly planning results. Compared with the most popular design for assembly (DFA) approach proposed by Boothroyd and Dewhurst [1989], the guidelines proposed in this work are more specific with regard to the detailed assembly process. These guidelines can be classified as two main categories: improving the assemblability of the product, and reducing the assembly cost of the product, as shown in Figure 6.1.

In order to improve the assemblability of the product, the redesign guidelines focus on the two following areas: to improve the relative assemblability (RA) caused by the influence of tolerance and clearance, and to reduce the assembly interference numbers. In order to reduce the assembly cost of the product, the redesign guidelines focus on the following areas: to reduce the assembly orientation change number, the assembly tool change number and the operation change number during the assembly process. The detailed redesign guidelines are discussed in the following sections.

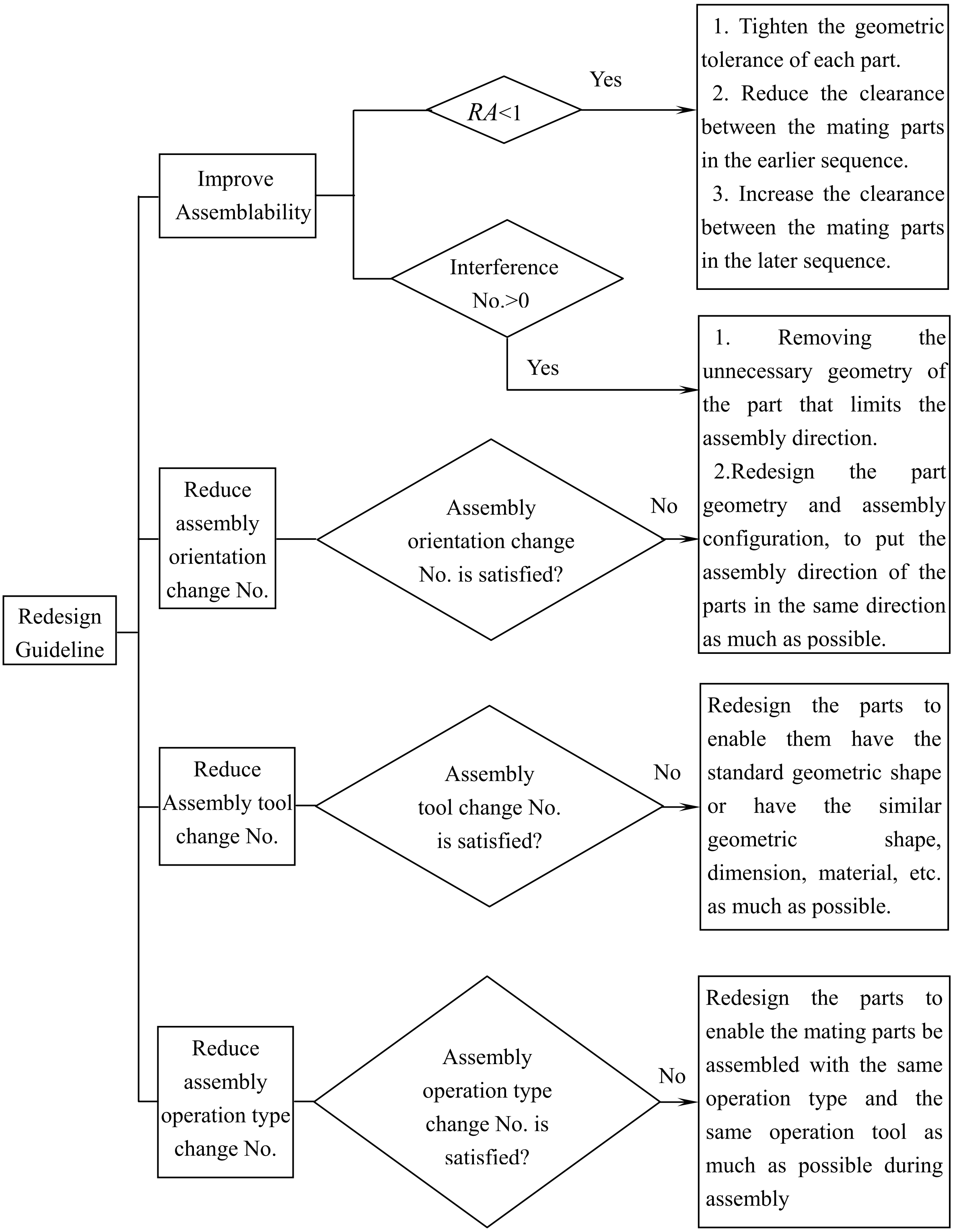


Figure 6.1 Redesign guidelines from the assembly planning results

6.2.1 Redesign suggestion from the assemblability evaluation

Through assembly planning, the product assemblability evaluation is made on the following two areas: the relative assemblability (*RA*) caused by the tolerance-based constraint and the assembly interference number.

6.2.1.1 Redesign suggestion from the relative assemblability

The relative assemblability (*RA*) represents the possibility to achieve a successful assembly product in a given assembly sequence considering the influence of tolerance and clearance. As discussed in the chapter 5, in a given assembly sequence, only when the $RA \geq 1$, can the assemblability of this sequence be ensured; otherwise, the smaller the *RA*, the worse the assemblability. So in a selected assembly sequence, if $RA<1$, the parts that can affect the value of *RA* should be identified and redesigned.

As shown in Figure 5.9, for this assembly product, from the assembly planning results, if the decision maker select some assembly sequences considering some factors, such as smaller assembly orientation change numbers, tool change numbers, etc., however, for these assembly sequences, if $RA<1$, then some parts in this product need be redesigned or modified. In this product, the geometric dimension and tolerance of Part 1, Part 2, Part 3 and Part 4 can affect the *RA* of the assembly sequence, so if $RA<1$ for the selected assembly sequence, then the designers for Part 1, Part 2, Part 3 or Part 4 should consider modify or redesign the parts, including the geometric dimension, tolerance or the geometric shape.

The detailed redesign schemes are summarized as follows:

Scheme 1: Tighten the geometric tolerance of each part which can affect the *RA*

The designer should tighten the geometric tolerance of each part which can affect the *RA* of the selected assembly sequence. The tightened geometric tolerance of the part can reduce the geometric deviations during the assembly process. For example, in the assembly as shown in Figure 5.9, if a selected sequence by the designer for Part 1 to Part 12 is as follows: 1-2-3-5-6-9-10-4-7-8-11-12, but the *RA* of this selected sequence is smaller than 1, then the designers for Part 1, Part 2, Part 3 and Part 4 can consider to tighten the geometric tolerance of each part, respectively, such as the perpendicularity tolerance Ø0.1 of the cylindrical feature *B* and *C* in Part 2, respectively, and the concentricity tolerance Ø0.05 in Part 2, as shown in Figure 5.11. The tightened geometric tolerance can result in the smaller geometric deviations during the assembly process, thus can increase the *RA* of the assembly sequence. However, the tightened geometric tolerance can generally result in the increase of the manufacturing cost. If the designers plan to maintain the manufacturability of the parts in some condition, then another redesign scheme can be considered as follows.

Scheme 2: Redesign the nominal dimension and dimensional tolerance of the assembly features

Firstly, the designer should redesign the nominal dimension and dimensional tolerance of the assembly features between the mating parts which are assembled in the earlier sequence, to reduce the mating clearance between the mating parts; secondly, the designer can also redesign the nominal dimension and dimensional tolerance of the assembly features between the mating parts which are assembled in the later sequence, to increase the mating clearance of the mating parts as long as the function of the parts or assembly will not be affected. Through the above two steps, the geometric deviations caused by the clearance between the mating parts in the earlier assembly sequence can be reduced during the assembly process; meanwhile, the increased clearance between the mating parts in the later assembly sequence can be used to compensate the geometric deviations during the assembly process.

For example, considering the selected assembly sequence (*RA*<1) by the designer for Part 1 to Part 12 (Figure 5.9) as follows: 1-2-3-5-6-9-10-4-7-8-11-12, the designer firstly can redesign the nominal dimension and dimensional tolerance of the assembly features between the mating parts Part 1, Part 2 and Part 3, i.e. Ø9 ± 0.1 of the cylindrical feature B of the Part 2 (Figure 5.11), Ø9 ± 0.1 of the cylindrical feature *B* of the Part 3 (Figure 5.12), Ø10 ± 0.1 of the two hole features *F* and *G* of the Part 1 (Figure 5.10), to reduce the mating clearance between these mating parts. Secondly, the designer can also redesign the nominal dimension and dimensional tolerance of the assembly features between the mating parts Part 2, Part 3 and Part 4, i.e. Ø14.8 ± 0.1 of the cylindrical feature *C* of the Part 2 (Figure 5.11), Ø14.8 ± 0.1 of the cylindrical feature *C* of the Part 3 (Figure 5.12), Ø15 ± 0.1 of the two hole features *F* and *G* of the Part 4 (Figure 5.13), to increase the mating clearance between these mating parts. The above redesign or modification schemes can effectively increase the *RA* of this selected assembly sequence.

6.2.1.2 Redesign suggestion from the assembly interference numbers

The assembly interference number represents the interference times happened during an assembly process in a given assembly sequence. Actually in a feasible assembly sequence, the assembly interference number must be zero; otherwise the assembly sequence is infeasible. So, from the assembly planning results which include different assembly sequences, only the assembly sequences with the zero assembly interference number can be used in the assembly planning process. This means the designer must select the assembly sequence with the zero assembly interference number. However, in some conditions, if we cannot get a feasible sequence without interference, or we can only get very few feasible sequences without interference from the assembly planning result, we should consider to redesign the parts to enable more feasible sequences can be derived, thus can provide the decision maker more choices on feasible sequence selection.

The proposed redesign guideline is to increase the feasible assembly directions in each step of the assembly process, and this guideline can also be used in reducing the number of assembly orientation change during assembly process. The detailed redesign guideline is discussed in Section 6.2.2.

6.2.2 Redesign suggestion from the number of assembly orientation change

As an important factor to evaluate the assembly time or assembly cost of an assembly product, the number of assembly orientation change in a selected assembly sequence can provide the designer the information on the necessity to redesign or modify the design.

Considering an assembly product, from the assembly planning results, where the number of assembly orientation change is too big no matter which assembly sequence is adopted. Because frequent assembly orientation change can increase the assembly cost and time, the design of the corresponding parts should be improved to reduce the number of assembly orientation change during the assembly process. Several conditions and schemes are discussed as follows.

6.2.2.1 Remove the unnecessary geometry of the part

Some parts with unnecessary geometry that limits the assembly direction of the mated parts should be identified, and the redesign is made accordingly to rectify it.

As shown in Figure 6.2, we assume Part 1 and Part 2 belong to a complex assembly product. The two steps of Part 1 limit the assembly direction of Part 2 from ± *X* directions. If these two steps are unnecessary for the function of the product, then they should be removed. Thus, the feasible assembly directions of Part 2 to Part 1 are increased, and the number of total assembly orientation change may be reduced during the whole assembly process.

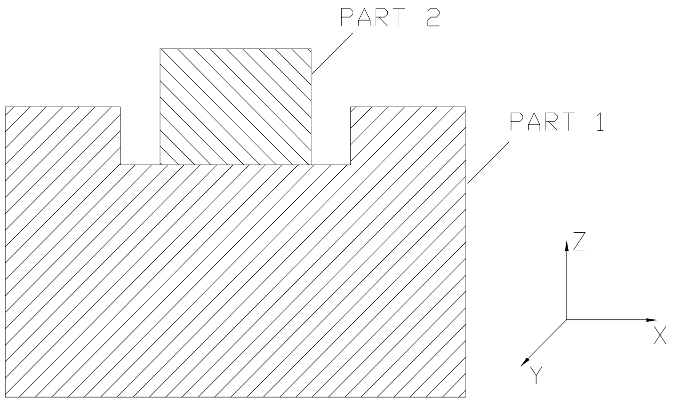


Figure 6.2 Limited assembly orientation of Part 2 to Part 1

Another case is the assembly product discussed in the chapter 5, as shown in Figure 5.9. From the assembly planning results, we assume that the designer selected an assembly sequence with a bigger number of assembly orientation change considering the other objective values. In order to reduce the number of assembly orientation change, the designer of the part should consider remove the unnecessary geometry of the part that limits the assembly orientation of the mated parts. In this case, the Part 13 limits the assembly direction of the Part 5, Part 6, Part 9 and Part 10 to the position in the assembly from –*Z* direction if Part 13 is assembled before these four parts. In the selected assembly sequence, if Part 13 is assembled before Part 5, Part 6 part 9 and Part 10, then the redesign of Part 13 may reduce the number of assembly orientation change.

The designer of Part 13 can redesign Part 13 into two single parts- Part 13a and Part 13b, as shown in Figure 6.3. After redesign, the function of the original Part 13 to hold Part 15 is maintained, and the feasible assembly directions of Part 5, Part 6 part 9 and Part 10 to the position in the assembly are increased, therefore the number of total assembly orientation change may be reduced during the whole assembly process.

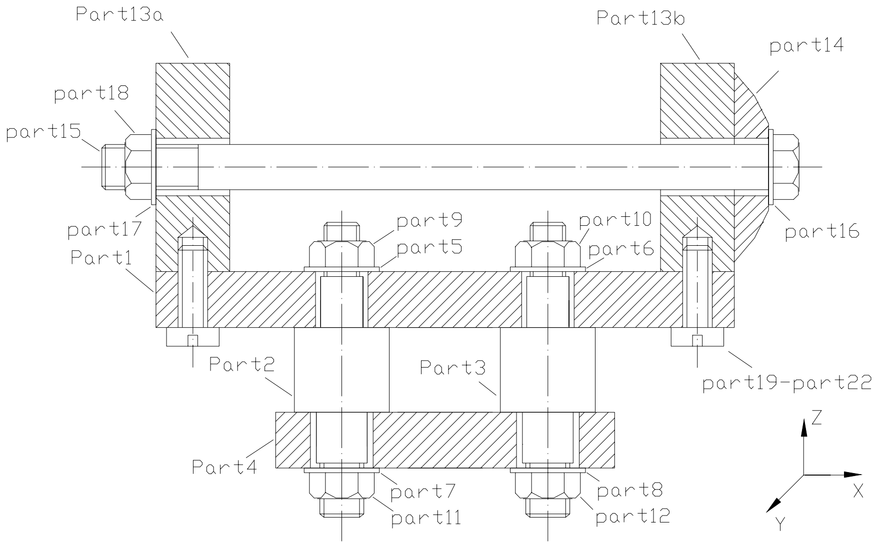


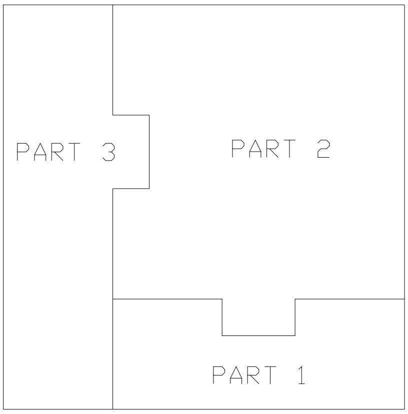
Figure 6.3 Redesigned Part 13a and Part 13b in assembly

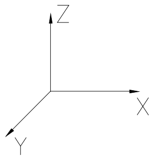
6.2.2.2 Redesign the part geometry and the assembly configuration

The objective of the redesign of the part geometry and the assembly configuration is try to put the assembly direction of the parts in the same direction as much as possible, thus to reduce the number of assembly orientation change during the assembly process.

A simple example is shown in Figure 6.4, we assume the parts cannot move along $\pm Y$ direction, then to assemble these three parts, two assembly directions are needed $+X$ or $-X$ and $+Z$ or $-Z$, and one assembly orientation change is needed. From the assembly planning results, if the designers hope to further reduce the number of assembly orientation change, the geometric shape of the parts should be redesigned. The designer of the Part 2 and Part 3 can redesign these two parts, respectively, as shown in Figure 6.5. Redesigned parts can be assembled together along $+Z$ or $-Z$

direction consecutively, and no assembly orientation change is needed.





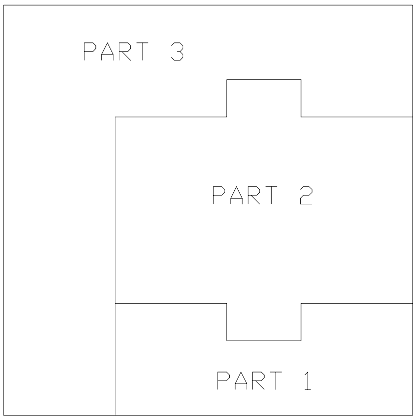


Figure 6.4 Assembly with one assembly orientation change

Figure 6.5 Assembly without assembly orientation change

6.2.3 Redesign suggestion from the number of assembly tool change

The assembly tools (grippers) are used to hold and deliver the parts to the assembly position during the assembly process. The type of the assembly tool is mainly determined by the geometric shape, dimension of the parts, generally the parts with the similar geometric feature can be hold and delivered by the same tool. For example, the part with the shin spherical shell can be hold and delivered by the vacuum clamping, and the cylindrical part can be hold by the chuck clamping, etc. Frequent change of assembly tools during assembly process will spend much time, if the number of tool change is big no matter what assembly sequence is selected or if the designer hopes to reduce the number of tool change in a selected assembly sequence, then some redesign works should be done to reduce the tool change number.

The redesign guideline is to enable the parts have the standard geometric shape or have the similar geometric shape, dimension, material, etc. as much as possible, therefore the same tool can be used to hold and deliver the parts with the similar geometric features. For example, as shown in Figure 6.6, the original design of Part 14 in Case 1 (Figure 5.9) has the quadratic curve surface, during assembly process, the vacuum clamping is needed to hold and deliver this part to the assembly position. If the designer redesign this part by adding a new cylindrical feature, as shown in Figure 6.7, then this part can be hold and delivered by the chuck clamping on the cylindrical feature, which can be also used to hold and deliver the Part 2 , Part 3 and Part 15 (Figure 5.9). This redesign scheme can reduce the assembly tool types used in holding and delivering the parts, therefore can effectively reduce the number of assembly tool change during the assembly process.

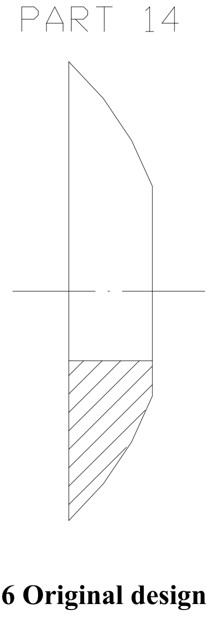


Figure 6.6 Original design of Part 14

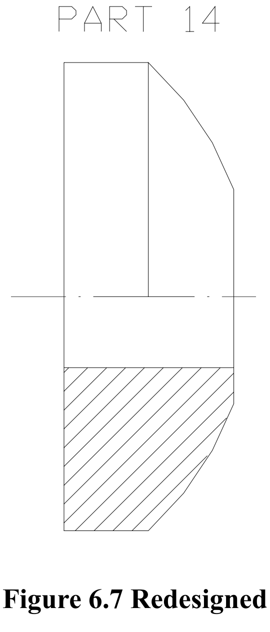


Figure 6.7 Redesigned Part 14

6.2.4 Redesign suggestion from the number of assembly operation change

Different assembly operations are used during the assembly process to fasten the mating parts, such as screwing, riveting, pressing, and so on. Different fastening operation needs different tools, such as screwdriver, spanner, rivet gun, etc. Frequent change of operation type needs more time to change the tools, thus should cost more time. If the working condition requires fewer assembly operation type changes during the assembly process, and the number of assembly operation type change cannot be reduced no matter what assembly sequence is selected by the designer, even if the designer assign a larger weight to this objective in the fitness function during the assembly planning evolution process, then the redesign works need to be done.

The redesign objective is to adopt the same operation tool as much as possible to carry out the same fastening operation of the mating parts during assembly. In the case as shown in Figure 5.9, although only two different operation types exist and two operation tools needed- spanner and screw driver, but different assembly sequence can result in different number of assembly operation type change. In order to further reduce the number of assembly operation type change in a selected assembly sequence, the designer can consider adopting only one operation type and one operation tool to carry out the fastening process during assembly. For example, the designer can change the fastening operation of the Part 13 and Part 1 by using the bolt and nut which is the same with the fastening operation between Part 2, Part 3 and Part 1, Part 4, as shown in Figure 6.8.

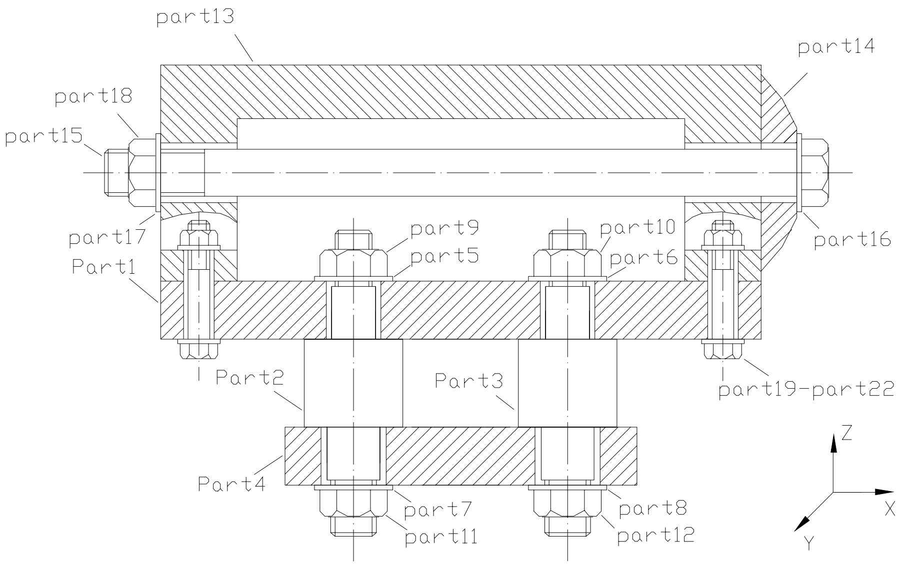


Figure 6.8 Redesigned assembly operation type

Accordingly, the designer of Part 13 should redesign the Part 13 (shown in Figure 6.9) to the new geometric shape, as shown in Figure 6.10.

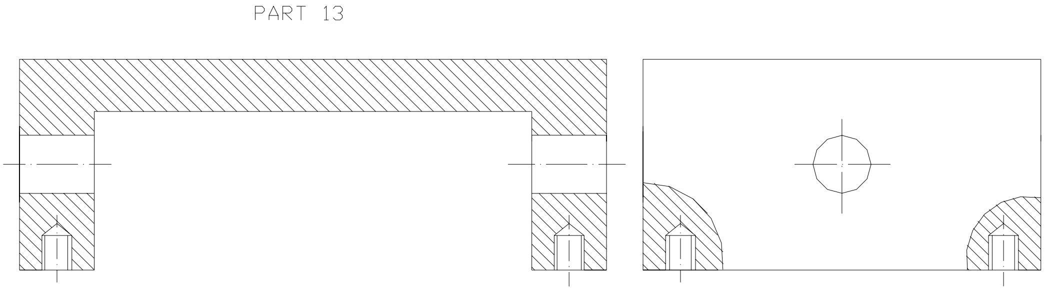


Figure 6.9 Original design of Part 13

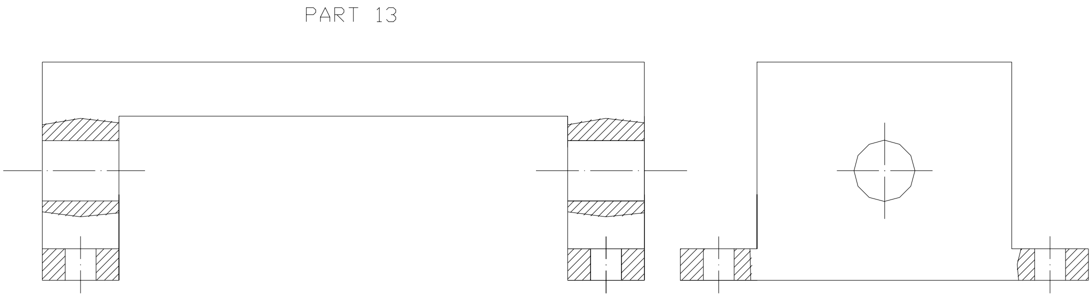


Figure 6.10 Redesigned Part 13

6.3 Summary

This chapter discusses the identification of potential design problems in the design stage through the evaluation of the assembly planning results, including different non-dominated assembly sequences, the corresponding assembly interference number, relative assemblability, and the different objective values, including the number of assembly orientation change, number of assembly tool change and number of assembly operation type change of each non-dominated assembly sequences.

According to the design problems, a set of redesign guidelines is proposed. These guidelines focus on the following two areas: Firstly, improve the assemblability of the product, by improving the relative assemblability (*RA*) caused by the influence of tolerance and clearance, and to reduce the assembly interference numbers. Secondly, reduce the assembly cost of the product, by reducing the assembly orientation change number, the assembly tool change number and the operation change number during the assembly process. These redesign guidelines can effectively help the designer improve the product design considering the detailed assembly process in the design stage; therefore, the design modification or redesign should be more practical and feasible.

Chapter 7 Collaborative Assembly Planning

A complex assembly product usually consists of many parts, and can be divided into several different subassemblies with specific assembly relationships and the functions. The assembly task for each subassembly can be carried out in different regions according to required assembly conditions and available facilities. The distribution of the assembly tasks can accelerate the assembly process in a parallel manner. The traditional single-user assembly planning approach is not suitable for the aforementioned distributed complex assembly, as it typically involves more than one planner for the various subassembly task assignments for the product at the different locations. The approach needs to be extended to a multi-user assembly planning environment, where several geographically dispersed planners can carry out the subassembly task assignments collaboratively, and complete the assembly planning for the assigned subassembly concurrently according to detailed assembly conditions, and available facilities at the different subassembly locations. The assembly planning process can thereby be accelerated, and the resulting assembly plan can then be more feasible and optimal.

This chapter presents a collaborative assembly planning approach. Section 7.1 discusses the system framework and working mechanism while Section 7.2 introduces the collaborative assembly planning procedure. A case study is given in Section 7.3 to illustrate the collaborative assembly planning process and Section 7.4 summarizes the chapter.

7.1 System framework and working mechanism

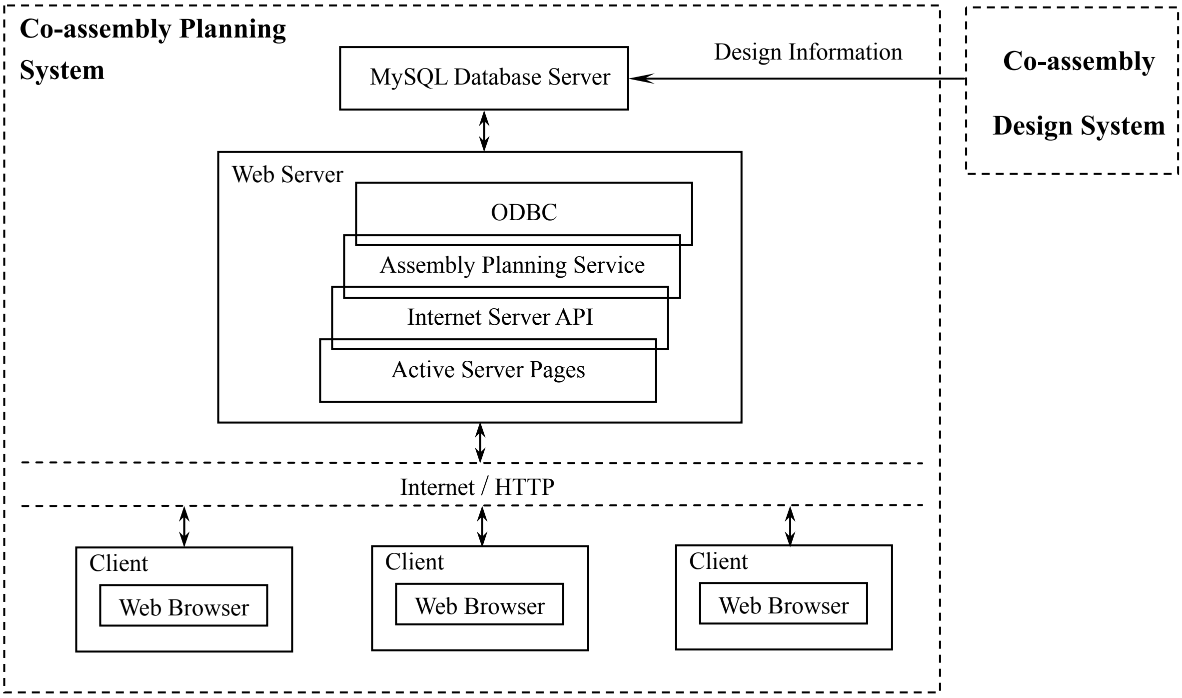


Figure 7.1 System framework for collaborative assembly planning

To realize the collaborative assembly planning in a distributed working environment, a system framework is proposed, as shown in Figure 7.1. It adopts a Browser/Server architecture with the assembly planning service application located in the web server, and it connects with the database server that stores the assembly design information and the assembly process information. The assembly design information comes from the co-assembly design system, and will be used to build the assembly interference matrix and conclude on the assembly interference number in an assembly sequence. In addition, the assembly design information will also be used to evaluate the assemblability of an assembly sequence by considering the tolerance and clearance influence (as discussed in chapter 4). The assembly process information comes from the web-based collaborative assembly planning system during the assembly planning process, as discussed in the following.

Different users can log onto the system through the web browser, and call the assembly planning service through the Internet Server API (ISAPI) and Active Server Pages (ASP). Through the ASP, the user can realize the active interaction with the database in the database server, i.e. to insert or delete the record in the database, and retrieve the record from the database. In the assembly planning process, the user needs to firstly check the assembly status of the product, such as the subassembly that has already been completed by the other users, and the parts included in each subassembly. The display of the assembly status can be updated by retrieving the updated information from the database. Secondly, through the ASP web page, the user can select the parts to assemble into a new subassembly, and assign a number for this subassembly. Thirdly, the user can indicate the detailed assembly requirement for the selected subassembly, and select the parameters for each part in the subassembly. At this stage, the user can assign fuzzy weight parameters for this subassembly, including the rough weight, maximum fluctuation of the weight, fluctuation probability of the weight for three objectives- number of orientation change, number of gripper change, and number of operation change, respectively. The fuzzy weight parameter selection is decided by the user by considering the working condition, the assembly facilities to carry out the assembly tasks of the selected subassembly, as well as the dimensions and weight of the parts. Besides the fuzzy weight parameters, the user also needs to indicate the gripper used to hold the part, and the operation type used to fasten the part during the assembly process. Once these parameters are assigned, they will be stored in the database for the use of the assembly planning service.

After the user has assigned the parameters and sent the request for assembly planning on the selected subassembly, the assembly planning application in the web server will be activated through the ISAPI extension, and the assembly planning results will be stored in the database, and can be fed back to the user as a record retrieved from the database.

7.2 Collaborative assembly planning procedure

In this section, the issues related to collaborative assembly planning will be discussed. These issues include the task assignment for the subassembly, the feasibility check on the subassembly task assignment, parameter selection for assembly planning, and assembly planning process.

The workflow for the whole collaborative assembly planning is discussed as follows:

7.2.1 The task assignment for the subassembly

The task assignment for the subassembly is an important step in collaborative assembly planning. Through task assignment, a complex assembly product is divided into several subassemblies with fewer parts. In each subassembly, the parts should be assembled together successfully, and the assembly cost should be lowered as much as possible.

In order to realize the above requirements, the assembly planners should consider some factors when they carry out the subassembly task assignment, as discussed in the following.

Factor 1: The assembly relationship among the parts

The assembly relationship among the parts must be considered. The assembly relationship represents the spatial configuration of the parts in a subassembly. Generally only the parts with the assembly relationship can be composed into a subassembly.

Factor 2: The facilities and conditions of the workshop

The assembly planner needs to select the parts according to the assembly facilities to carry out the assembly task of the selected subassembly. For example, if the facilities of the planner are capable of delivering and assembling large and heavy parts, then the planner can select large and heavy parts according to the assembly relationship among the parts. Meanwhile, the planner needs to try to reduce the number of assembly orientation changes especially of the big and heavy parts as these would cost more time and labor. Thus the planner would assign the larger rough weight to the objective of orientation change in the subsequent assembly planning process.

Factor 3: Three objectives of the assembly planning

During the task assignment, the planner should also consider the three objectives of assembly planning: number of assembly orientation changes, number of assembly gripper changes, and number of assembly operation changes. Based on the assembly relationship among the parts (Factor 1) and the assembly facilities and conditions (Factor 2), the planner should consider the following factors:

(1) Try to select the parts that can be assembled along the same direction into a subassembly as much as possible, to reduce the number of orientation changes during the assembly process;

(2) Try to select the parts that have the same or similar geometric shape in a subassembly as much as possible, to reduce the number of gripper changes during the assembly process;

(3) Try to select the parts which have the same fastening method into a subassembly as much as possible, to reduce the number of operation changes during the assembly process.

7.2.2 Feasibility check of the subassembly task assignment

During the collaborative assembly planning process, when the subassembly task assignment is completed, sometimes the parts in a subassembly can cause collision interference with the parts in other subassemblies when these subassemblies are delivered and assembled into the product. This is the collision interference among different subassemblies.

To avoid the collision interference among different subassemblies and ensure that the subassembly task assignment is feasible, a feasibility check algorithm is proposed as follows:

We assume a product is to be assembled by three planners collaboratively. Firstly, the subassembly S_A is selected by the planner A, with n_1 parts $\{P_{A1}, P_{A2}, \ldots\ldots P_{An1}\} \in S_A$; secondly, the subassembly S_B is selected by the planner B, with n_2 parts $\{P_{B1},$

P_{B2}, ……$P_{Bn2}\} \in S_B$; thirdly, the subassembly S_C is selected by the planner *C*, with n_3 parts $\{P_{C1}, P_{C2}, \ldots\ldots P_{Cn3}\} \in S_C$.

Algorithm: **Feasibility check on subassembly task assignment**

Step 1: To check the feasibility of task assignment for subassembly S_A according to subassembly S_B.

Step 1.1: After the subassembly S_A has been selected by planner *A*, planner *B* selects the parts for subassembly S_B, using the interference matrix approach (as discussed in chapter 5) to conclude the feasible assembly direction of each part in S_B to the position in the product according to all parts in S_A, represented as $d(P_{Bi}\, S_A)$, $i \in \{1, 2, \ldots\ldots n_2\}$; if $d(P_{Bi}\, S_A) \neq \emptyset$, then $d(P_{Bi}\, S_A) \in \{\pm X, \pm Y, \pm Z\}$.

Step 1.2: If $d(P_{Bi}\, S_A) \neq \emptyset$ for all $i \in \{1, 2, \ldots\ldots n_2\}$, and $d(P_{B1}\, S_A) \cap d(P_{B2}\, S_A) \cap \ldots\ldots d(P_{Bn2}\, S_A) \neq \emptyset$, then there is no collision interference between subassembly S_A and subassembly S_B. Go to step 1.3.

Else, if any $d(P_{Bi}\, S_A) = \emptyset$ for $i \in \{1, 2, \ldots\ldots n_2\}$, then the part P_{Bi} has collision interference with the subassembly S_A. The task assignment for subassembly S_A is infeasible, and the parts in subassembly S_A need be reassigned. Stop.

Else, if $d(P_{Bi}\, S_A) \neq \emptyset$ for all $i \in \{1, 2, \ldots\ldots n_2\}$, but $d(P_{B1}\, S_A) \cap d(P_{B2}\, S_A) \cap \ldots\ldots d(P_{Bn2}\, S_A) = \emptyset$, then there is collision interference between subassembly S_B and subassembly S_A. The task assignment for subassembly S_B is infeasible, and the parts in subassembly S_B need be reassigned. Stop.

Step 1.3: Using the similar steps described in step 1.1 and step 1.2 to check the feasibility of task assignment for subassembly S_A according to subassembly S_C, if it is feasible, then the task assignment for subassembly S_A is feasible. Go to step 2. Otherwise, reassign the parts in subassembly S_A or S_C, as discussed in step 1.2.

Step 2: Using the similar steps described in step 1, to check the feasibility of task assignment for subassembly S_B according to subassembly S_C.

Using the above algorithm, the planner can check the feasibility of the subassembly task assignment, and this algorithm can be extended and suitable for the feasibility check on more subassembly task assignments by more different planners.

7.2.3 Parameter selection in assembly planning

After the subassembly task assignment is completed, the planner needs to select the parameters for the parts in the subassembly for assembly planning. The parameters include the tool used to hold and deliver the part, and the assembly operation type for each part during the assembly process. Besides the above parameters, the planner needs also to select the fuzzy weight parameters, including the rough weight, the maximum fluctuation of the weight, and the fluctuation probability of the weight, for the objectives of orientation change, gripper change and operation change, respectively. These fuzzy weight parameters are used to build the fitness functions for assembly planning using the proposed multi-objective genetic algorithm, as discussed in chapter

5.

The selection of the above parameters should be determined by the assembly planner according to the geometry, dimension, and the assembly conditions of the parts in the subassembly, and the working condition and facilities to carry out the assembly process for the selected subassembly.

7.2.4 Assembly planning for the subassembly using the multi-objective genetic algorithm

After the parameters are selected for the selected subassembly, the assembly planning application using the multi-objective genetic algorithm will be activated. The detailed assembly planning approach using the multi-objective genetic algorithm has been discussed in chapter 5. Through the assembly planning, the feasible non-dominated assembly planning solutions will be displayed at the web client for the planner to evaluate the assembly process, as well as the assembly design through identifying the potential design problems, as discussed in chapter 6.

7.3 Case study

In this section, a motor table assembly (shown in Figure 7.2) is used to illustrate the collaborative assembly planning process.

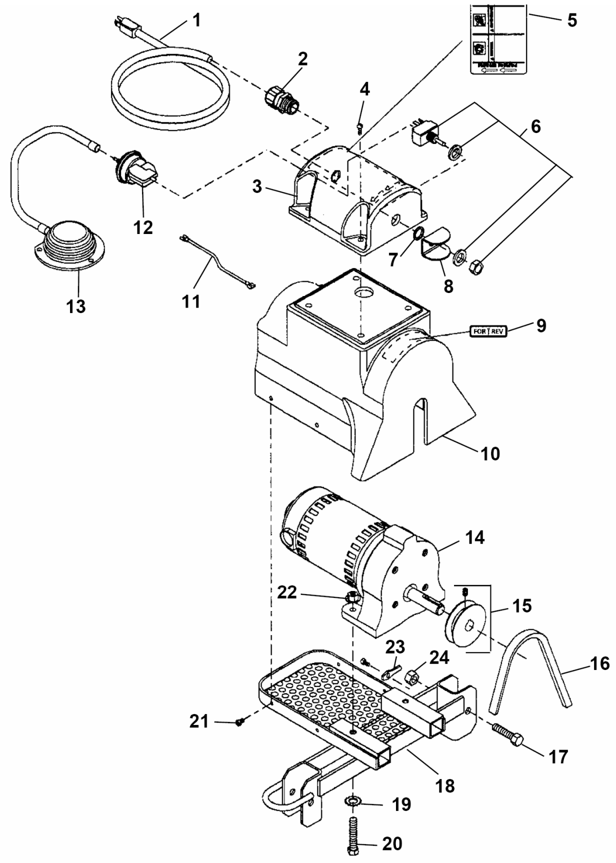
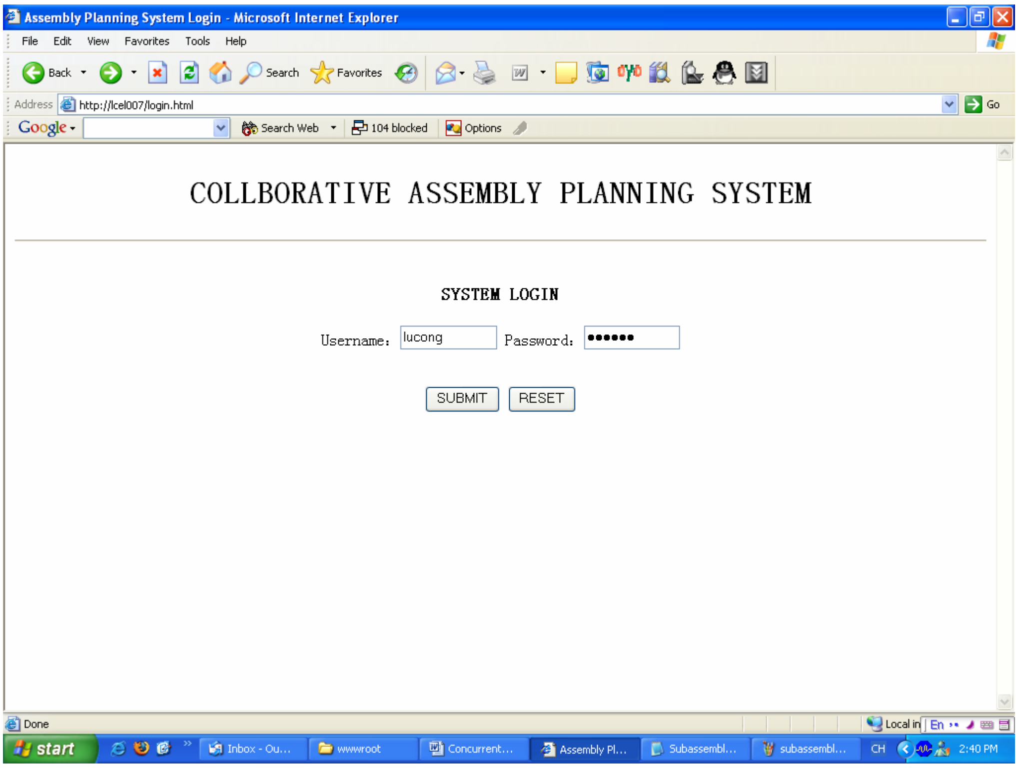


Figure 7.2 A motor table assembly [http://www.ridgid.com/catalogdocs/k7500.pdf]

In this case, we assume two geographically dispersed planners - planner *A* and planner *B*, who are working collaboratively to carry out the assembly planning for the product. As shown in Figure 7.3, when the planner inputs the username and password, he can log onto the web-based collaborative assembly planning system, and enter the subassembly task assignment web page, as shown in Figure 7.4.



Address: http://lcel007/login.html

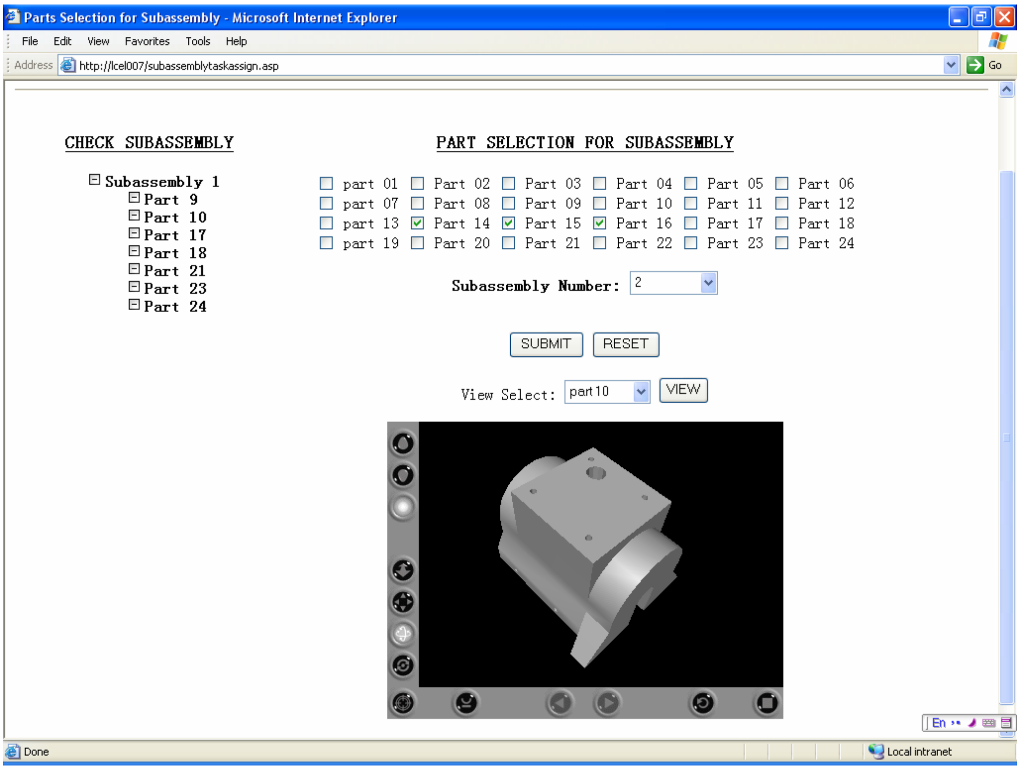
COLLBORATIVE ASSEMBLY PLANNING SYSTEM

SYSTEM LOGIN

Username: lucong Password: ••••••

SUBMIT RESET

Figure 7.3 User login



Address: http://lcel007/subassemblytaskassign.asp

CHECK SUBASSEMBLY

⊟ Subassembly 1
  ⊟ Part 9
  ⊟ Part 10
  ⊟ Part 17
  ⊟ Part 18
  ⊟ Part 21
  ⊟ Part 23
  ⊟ Part 24

PART SELECTION FOR SUBASSEMBLY

☐ part 01	☐ Part 02	☐ Part 03	☐ Part 04	☐ Part 05	☐ Part 06
☐ part 07	☐ Part 08	☐ Part 09	☐ Part 10	☐ Part 11	☐ Part 12
☐ part 13	☑ Part 14	☑ Part 15	☑ Part 16	☐ Part 17	☐ Part 18
☐ part 19	☐ Part 20	☐ Part 21	☐ Part 22	☐ Part 23	☐ Part 24

Subassembly Number: 2

SUBMIT RESET

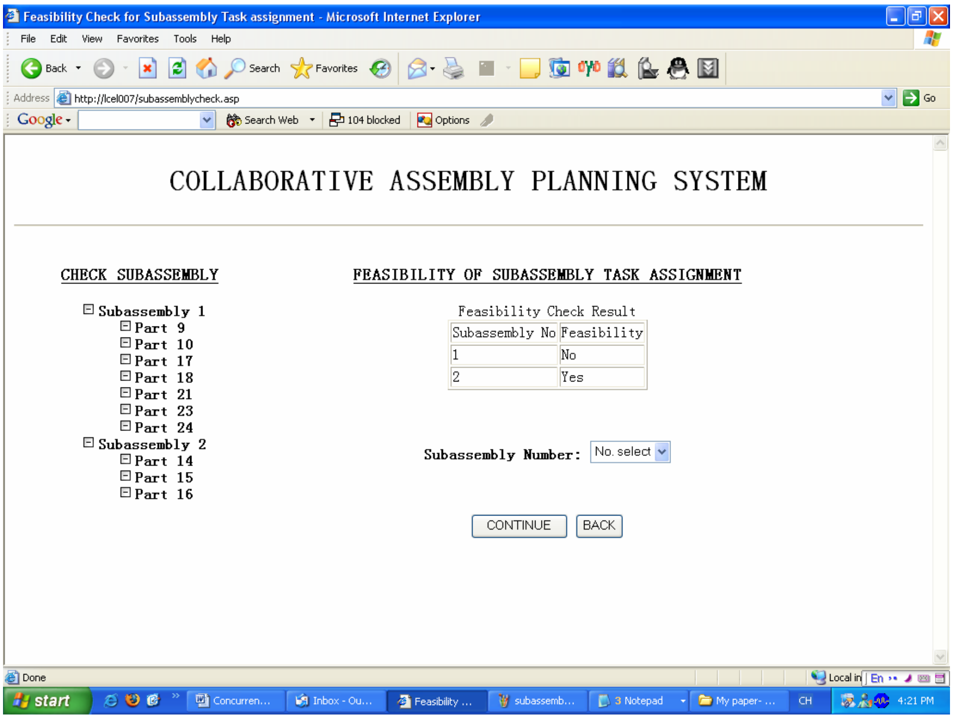
View Select: part 10 VIEW

Figure 7.4 Part selection for subassembly 2

During the subassembly task assignment, as shown in Figure 7.4, through the VRML plug-in in the web page, the planner can check and view the 3D solid model of the assembly and each part that has been stored in the web server. When he inputs the assembly or the part number, he can view the 3D solid model of the assembly or the selected part. He can also rotate, zoom in or zoom out of the model to view the different aspect of the model from different angle. As shown in Figure 7.4, Part 10 is displayed in the VRML plug-in for the planner to view. Through viewing the assembly or the part online, the planner in the web client can check the assembly relationship among different parts and the geometric shape of each part; thus it can help the planner to carry out the subassembly task assignment.

At this stage, as shown in Figure 7.4, planner *A* has finished the task assignment for subassembly 1, and planner *B* can then check the assembly status in subassembly 1 by checking the tree list of subassembly 1, and view each part in subassembly 1. Thereafter planner *B* can view and select the parts from the remaining unassembled parts into subassembly 2. At this time, planner *B* only selects parts 14, 15 and 16 into subassembly 2, and then submits the part selection information for the feasibility check on subassembly task assignment using the algorithm proposed in section 7.2.2.

The feasibility check results on the subassembly task assignment are displayed on the web page, as shown in Figure 7.5. The results show that the task assignment for subassembly 1 is infeasible, and the parts in subassembly 1 need to be reselected. Similarly, the feasibility on subassembly task assignment can also be checked by planner *A* when the initial task assignment on subassemblies 1 and 2 have been completed. In this case, when planner *A* finds that the task assignment for subassembly 1 obtained is infeasible, he enters the subassembly task assignment web page again to reselect the parts for subassembly 1. And planner *B* will also have to reselect the parts for subassembly 2 according to the new parts selected for subassembly 1, as shown in Figure 7.6.



COLLABORATIVE ASSEMBLY PLANNING SYSTEM

CHECK SUBASSEMBLY

- ⊟ Subassembly 1
  - ⊟ Part 9
  - ⊟ Part 10
  - ⊟ Part 17
  - ⊟ Part 18
  - ⊟ Part 21
  - ⊟ Part 23
  - ⊟ Part 24
- ⊟ Subassembly 2
  - ⊟ Part 14
  - ⊟ Part 15
  - ⊟ Part 16

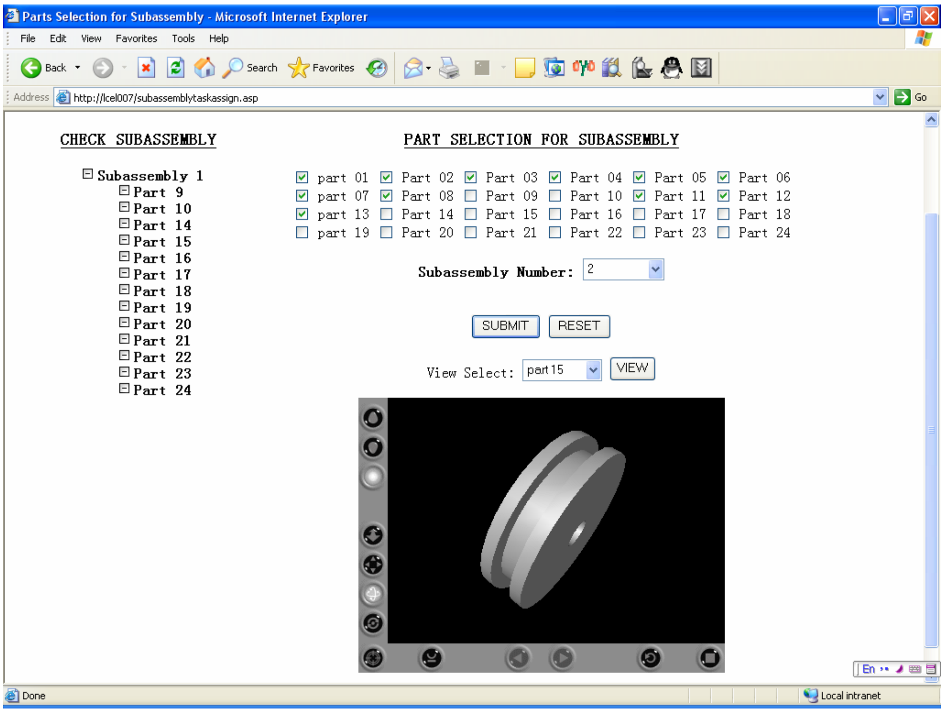
FEASIBILITY OF SUBASSEMBLY TASK ASSIGNMENT

Feasibility Check Result

Subassembly No	Feasibility
1	No
2	Yes

Subassembly Number: No. select

Figure 7.5 Feasibility check on subassembly task assignment



CHECK SUBASSEMBLY

- Subassembly 1
  - Part 9
  - Part 10
  - Part 14
  - Part 15
  - Part 16
  - Part 17
  - Part 18
  - Part 19
  - Part 20
  - Part 21
  - Part 22
  - Part 23
  - Part 24

PART SELECTION FOR SUBASSEMBLY

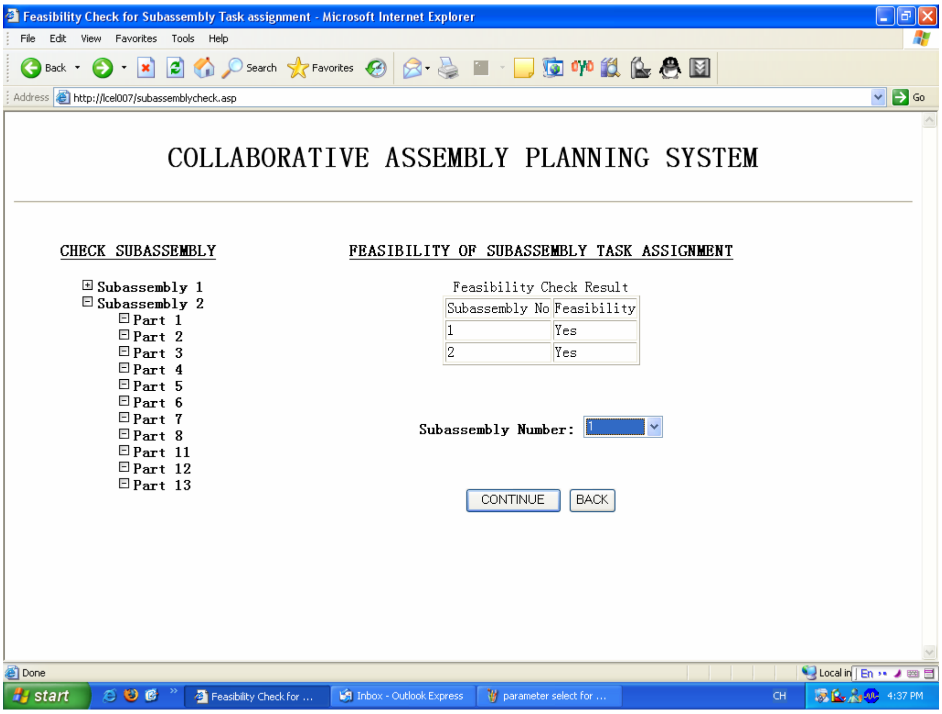
☑ part 01 ☑ Part 02 ☑ Part 03 ☑ Part 04 ☑ Part 05 ☑ Part 06
☑ part 07 ☑ Part 08 ☐ Part 09 ☐ Part 10 ☑ Part 11 ☑ Part 12
☑ part 13 ☐ Part 14 ☐ Part 15 ☐ Part 16 ☐ Part 17 ☐ Part 18
☐ part 19 ☐ Part 20 ☐ Part 21 ☐ Part 22 ☐ Part 23 ☐ Part 24

Subassembly Number: 2

View Select: part15

Figure 7.6 Part reselection for subassembly 2

The feasibility check results on the new subassembly task assignment are displayed on the web page, which shows that the task assignment for subassembly 1 and subassembly 2 are now both feasible, as shown in Figure 7.7.



COLLABORATIVE ASSEMBLY PLANNING SYSTEM

CHECK SUBASSEMBLY

- Subassembly 1
- Subassembly 2
  - Part 1
  - Part 2
  - Part 3
  - Part 4
  - Part 5
  - Part 6
  - Part 7
  - Part 8
  - Part 11
  - Part 12
  - Part 13

FEASIBILITY OF SUBASSEMBLY TASK ASSIGNMENT

Feasibility Check Result

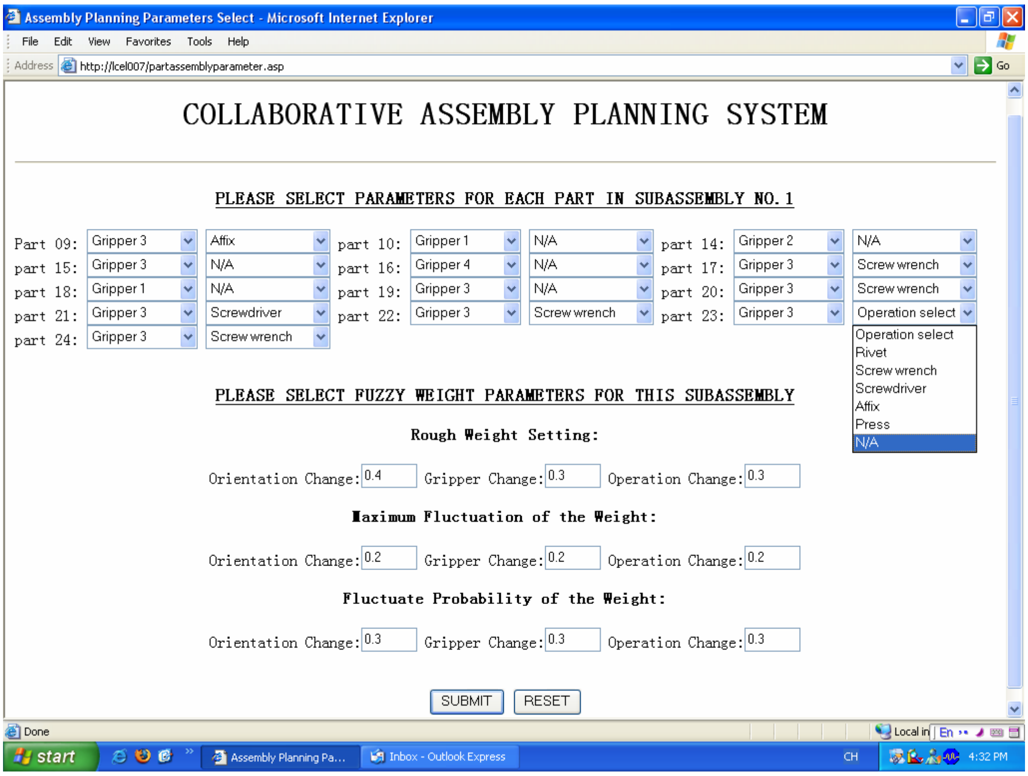
Subassembly No	Feasibility
1	Yes
2	Yes

Subassembly Number: 1

Figure 7.7 Feasibility check on subassembly task reassignment

At this stage, when the planner *A* enters the subassembly task assignment web page to check the subassembly task assignment condition again, he will find that all the parts have been selected and assigned into the subassembly 1 or subassembly 2. Further, planner *A* can also check and find the new task assignment for both subassembly 1 and subassembly 2 are feasible (Figure 7.7), and thus the subassembly task assignments are complete.

In the next step, planner *A* can enter the subassembly number – 1 (Figure 7.7), for which he would make the further assembly planning, and enter the parameter selection web page for subassembly 1, as shown in Figure 7.8. In this web page, the planner *A* can input the tool (gripper) type and assembly operation type for each part in subassembly 1. Besides the above parameters, the planner also needs to input the fuzzy weight parameters, including the rough weight, the maximum fluctuation of the weight, and the fluctuation probability of the weight, for each objective of orientation change, gripper change, and operation change, respectively, for assembly planning, as shown in Figure 7.8.



COLLABORATIVE ASSEMBLY PLANNING SYSTEM

PLEASE SELECT PARAMETERS FOR EACH PART IN SUBASSEMBLY NO. 1

Part 09:	Gripper 3	Affix	part 10:	Gripper 1	N/A	part 14:	Gripper 2	N/A
part 15:	Gripper 3	N/A	part 16:	Gripper 4	N/A	part 17:	Gripper 3	Screw wrench
part 18:	Gripper 1	N/A	part 19:	Gripper 3	N/A	part 20:	Gripper 3	Screw wrench
part 21:	Gripper 3	Screwdriver	part 22:	Gripper 3	Screw wrench	part 23:	Gripper 3	Operation select
part 24:	Gripper 3	Screw wrench						

PLEASE SELECT FUZZY WEIGHT PARAMETERS FOR THIS SUBASSEMBLY

Rough Weight Setting:

Orientation Change: 0.4 Gripper Change: 0.3 Operation Change: 0.3

Maximum Fluctuation of the Weight:

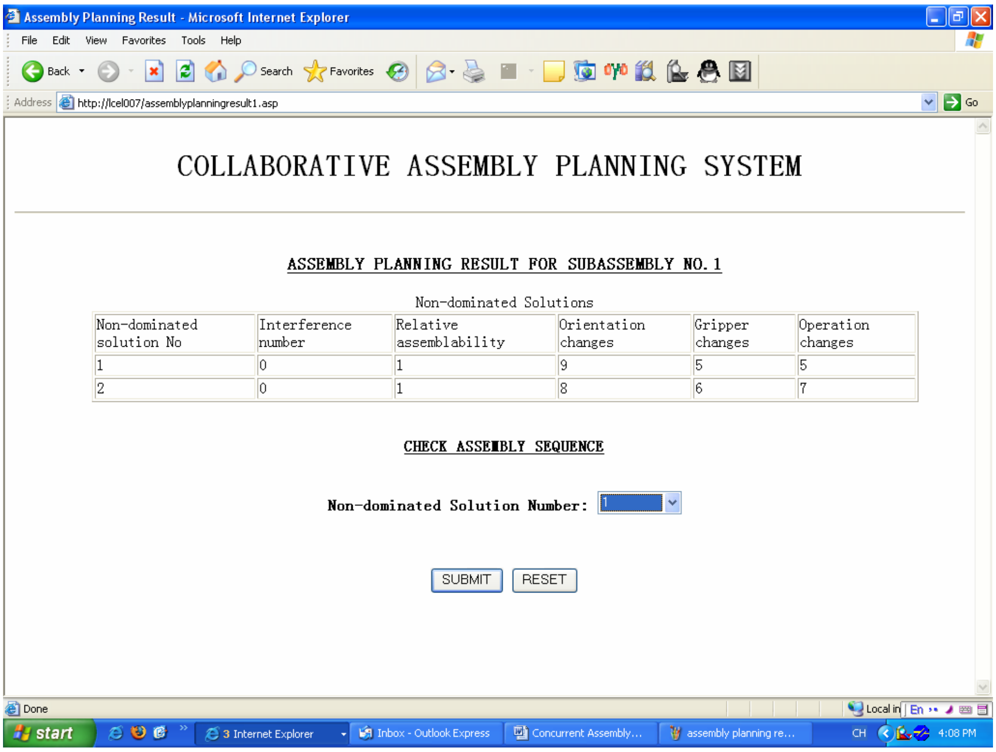
Orientation Change: 0.2 Gripper Change: 0.2 Operation Change: 0.2

Fluctuate Probability of the Weight:

Orientation Change: 0.3 Gripper Change: 0.3 Operation Change: 0.3

Figure 7.8 Parameter selection for subassembly No. 1

After all the parameters are input, planner *A* can submit them to the web server for assembly planning. The assembly planning results can be displayed on the web client, including the different non-dominated solutions for the selected subassembly 1, and the value of different objectives of each non-dominated solutions, as shown in Figure 7.9. The planner can further check the detailed assembly sequence of the non-dominated solution on which he shows interest by inputting the number of that solution, which is No. 1 in this case (Figure 7.9). Then the web page showing the assembly sequence of the non-dominated solution No. 1 will be displayed, as shown in Figure 7.10.



COLLABORATIVE ASSEMBLY PLANNING SYSTEM

ASSEMBLY PLANNING RESULT FOR SUBASSEMBLY NO. 1

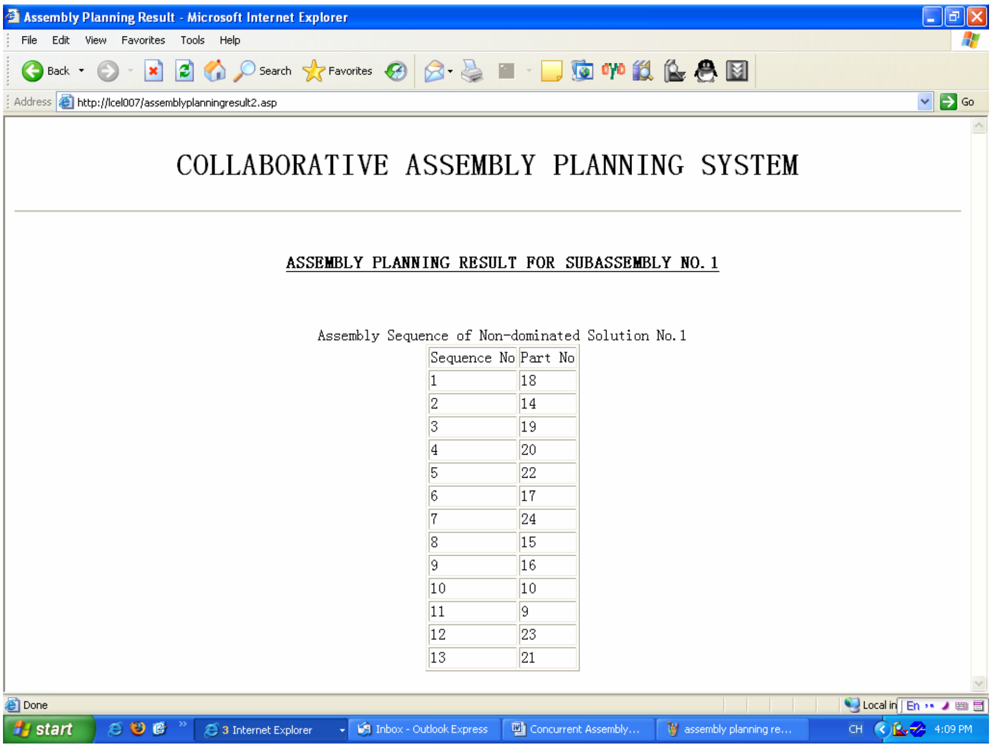
Non-dominated Solutions

Non-dominated solution No	Interference number	Relative assemblability	Orientation changes	Gripper changes	Operation changes
1	0	1	9	5	5
2	0	1	8	6	7

CHECK ASSEMBLY SEQUENCE

Non-dominated Solution Number: 1

Figure 7.9 Assembly planning results showing evolved non-dominated solutions



COLLABORATIVE ASSEMBLY PLANNING SYSTEM

ASSEMBLY PLANNING RESULT FOR SUBASSEMBLY NO. 1

Assembly Sequence of Non-dominated Solution No. 1

Sequence No	Part No
1	18
2	14
3	19
4	20
5	22
6	17
7	24
8	15
9	16
10	10
11	9
12	23
13	21

Figure 7.10 Assembly sequence of a non-dominated solution

Similarly, planner *B* can complete the assembly planning for subassembly 2 as in the above steps. And the final assembly plan is composed of the assembly plan of subassembly 1 and subassembly 2, which can be carried out concurrently in the different assembly facilities.

7.4 Summary

This chapter discusses a collaborative assembly planning approach which enables geographically dispersed planners to carry out the assembly planning task collaboratively, based on the GA-based assembly planning approach proposed in chapter 5. Through subassembly task assignment, feasibility check of the subassembly task assignment and parameter selection, the assembly can be decomposed into several subassemblies, and for each subassembly, the non-dominated solutions can be derived considering the detailed assembly facilities and condition, and the experience of the planners.

Chapter 8 Conclusions and Recommendations

8.1 Conclusions

The research in this thesis concerns a collaborative assembly design modification and assembly planning approach, to enhance the efficiency of product assembly design and assembly planning. The focus is mainly on the following areas:

(1) Investigate an approach to realize design modification in collaborative assembly design

(2) Investigate an approach to evaluate the product assemblability in different assembly sequences

(3) Propose and develop an enhanced assembly planning approach using multi-objective genetic algorithm

(4) Propose a set of redesign guidelines based on potential design problems that can be identified from the assembly planning results

(5) Propose and develop a collaborative assembly planning approach

The detailed contributions are concluded as follows:

- Investigate an approach to realize design modification in collaborative assembly design. A feature-based hierarchical co-assembly representation model is proposed, through which the assembly relationship between different parts and the network-based working relationship between geographically dispersed designers can be built up. Based on the proposed co-assembly representation model, an XML schema has been developed to transfer the design modification information. In order to realize the design modification, a control mechanism is proposed for design modification propagation control and transfer of the design modification information to different designers in a collaborative design environment. A three-tier client-server system framework has been developed, and this framework can realize the above design modification issue effectively in a collaborative assembly design environment.

- Investigate an approach to evaluate the product assemblability in different assembly sequences. A concept called Sensitive Tolerance in Assembly is proposed and its influence on assembly has been investigated. By analyzing the influence of tolerance and assembly clearance on assembly, an approach is proposed using transformation matrices to derive the geometric deviations of mating features and their propagations and accumulations in different assembly sequences. Through comparison between the geometric deviations and the assembly clearance at the final assembly process, the relative assemblability of the product in different assembly sequences can be concluded.

- Propose an enhanced assembly planning approach using a multi-objective genetic algorithm. The relative assemblability of different assembly sequences is used as a constraint in assembly planning to make the assembly planning results more feasible. In addition, a fuzzy weight distribution algorithm is proposed. Based on this algorithm, a genetic search algorithm with multiple search directions has been proposed which builds different fitness functions using the fuzzy weight distribution algorithm. This can overcome the problem of the uncertainty of weights set by the decision maker, and can find more non-dominated solutions with the decision maker's experience.

- Propose a set of redesign guidelines. The identification of potential design problems through the evaluation of the assembly planning results is first discussed, including the different non-dominated assembly sequences, the corresponding assembly interference number, relative assemblability, and the different objective values. According to the design problems, a set of redesign guidelines is proposed. These guidelines focus on the following two areas: firstly, to improve the assemblability of the product by improving the relative assemblability (*RA*), and to reduce the assembly interference number; secondly, to reduce the assembly cost of the product, by reducing the assembly orientation change number, the assembly tool change number and the operation change number during the assembly process. These redesign guidelines can effectively help the designer improve the product design by considering the detailed assembly process in the design stage. Therefore, the design modification or redesign should be more practical and feasible.

- Propose a collaborative assembly planning approach. A web-based collaborative assembly planning approach is proposed, which enables geographically dispersed planners to carry out the assembly planning task collaboratively, based on the GA-based assembly planning approach proposed earlier. A Browser/Server system framework has been developed, and an algorithm to check the feasibility of the subassembly task assignment is proposed. Through the subassembly task assignment, the feasibility check of the subassembly task assignment and parameter selection, the assembly can be decomposed into several subassemblies, and for each subassembly, the non-dominated solutions can be derived by considering the detailed assembly facilities and condition, and the experience of the planners.

This research work presents a collaborative assembly design modification and assembly planning approach, which can be used in the design and manufacturing process of complex assembly products that include many parts and need to be outsourced to different manufacturers. This approach can help the manufacturers to shorten the product development cycle, and satisfy the rapidly changing market requirements.

8.2 Recommendations for future works

Based on the research presented in this thesis, there are some issues to be further addressed in future work, as discussed in the following:

- In collaborative assembly design modification, we have mainly considered the influence of design modification on the geometric mating constraints between the mating features. Other aspects such as degree of freedom and motion limits have not been considered. The degree of freedom and motion limits can usually affect the product assemblability, and these factors can also be affected by the design modification; so the influence of design modification on these factors needs to be further investigated.

- In assembly design, some design information, such as tolerance design, cannot be retrieved automatically for assembly planning because the tolerance modeling function has not been developed. They need to be manually edited and input to the assembly planning system. Further study is needed to realize the tolerance modeling and further realize the seamless integration of the assembly design system and the assembly planning system.

- The co-assembly design modification approach proposed in the thesis is realized in a client-server architecture, where the modeling function is carried out in the modeling server, and the different design clients carry out the design or design modification in different locations with the same modeling server. The heterogeneous co-assembly problems which can be caused by using different CAD systems with different modeling kernels will be considered and investigated in the future works.

- In the assembly planning stage, the mechanical stability in an assembly sequence has not been considered. Sometimes in an assembly sequence, we need some holding device to complete the assembly process due to the mechanical instability. Therefore further study is needed on evaluation of mechanical stability in assembly planning.

- In this research, a set of redesign guidelines has been proposed through the evaluation of the assembly planning results. However, these guidelines cannot help the designer to realize the design modification or redesign automatically. For future work, further study is needed to develop some operators based on these redesign guidelines to carry out the design modification or redesign automatically. This can facilitate the further integration of the assembly design system and the assembly planning system.

- In collaborative assembly planning, some communication functions among different planners need be further developed to help the planners complete the collaborative assembly planning more conveniently.

References:

Alibre, 2004, Alibre Design, Alibre Inc., Richardson, TX. (http://www.alibre.com).

ANSI (American National Standard Institute), 1982, Dimensioning and Tolerancing, ANSI Y14.5M-1982, American Society of Mechanical Engineers, New York.

Ashiagbor, A., Liu, H.C. and Nnaji, B.O. Tolerance Control and Propagation for the Product Assembly Modeller, International Journal of Production Research, 36 (1), pp.75-93. 1998.

Baldwin, D. F., Abell, T. E., Lui, M.-C. M., De Fazio, T. L., and Whitney, D. E. An Integrated Computer Aid for Generating and Evaluating Assembly Sequences for Mechanical Products, IEEE Transactions on Robotics and Automation, 7(1), pp.78-94. 1991.

Bidarra, R., Kranendonk, N., Noort, A. and Bronsvoort, W. F. A Collaborative Framework for Integrated Part and Assembly Modeling, Transactions of the ASME, Journal of Computing and Information Science in Engineering, 2 (4), pp.256-264. 2002.

Boothroyd, G. and Dewhurst P. Product Design for Assembly Handbook, Boothroyd and Dewhurst Inc., 1989.

Bourjault A. Contribution a une approche methodologique de I'assemblage automatise: Elaboration automatique des sequences operatoires. PhD thesis, Universite de Franche-Comte, 1984.

Chase, K.W., Gao, J., Magleby, S.P. and Sorensen, C.D. Including Geometric Feature variations in tolerance analysis of mechanical assemblies, IIE Transactions, 28, pp. 795-807. 1996.

Chen, L., Song, Z. and Feng, L. Internet-enabled Real-time Collaborative Assembly Modeling via an E-assembly System: Status and Promise, Computer Aided Design, 36(9), pp. 835-847. 2004.

Chen, S. F. and Liu, Y. J. An Adaptive Genetic Assembly-sequence Planner, International Journal of Computer Integrated Manufacturing, 14(5), pp.489-500. 2001.

CoCreate, 2004, OneSpace.net, CoCreate Software, Inc., Fort Collins, CO. (http://www.cocreate.com).

De Fazio, T. L. A Prototype of Feature-based Design for Assembly, ASME Advances in Design Automation, Chicago, IL, USA, pp. 9-16. 1990.

De Fazio, T. L. and Whitney, D. E. Simplified Generation of All Mechanical Assembly Sequences, IEEE Journal of Robotics and Automation, 3(6), pp.640-658. 1987.

Delchambre, A. A Pragmatic Approach to Computer-aided Assembly Planning. In Proceedings of the IEEE International Conference on Robotics and Automation, 1990, Cincinnati, OH, pp.1600-1605.

Delchambre, A. Computer-aided Assembly Planning. London: New York: Chapman & Hall. 1992.

Desrochers, A. and Riviere, A. A Matrix Approach to the Representation of Tolerance Zones and Clearances, International Journal of Advanced Manufacturing Technology, 13, pp. 630-636. 1997.

Dini, G.., Failli, F., Lazzerini, B. and Marcelloni, F. Generation of Optimized Assembly Sequence Using Genetic Algorithms, Annals of the CIRP, 48(1), pp.17-20. 1999.

Dini, G.. and Santochi, M. Automated Sequencing and Subassembly Detection in Assembly Planning, Annals of the CIRP, 41(1), pp.1-4. 1992.

Dong, T., Tong, R., Zhang, L. and Dong J.X. A Collaborative Approach to Assembly Sequence Planning, Advanced Engineering Informatics, 19, pp.155-168. 2005.

Extensible Markup Language (XML). (http://www.w3.org/xml)

Foster, L.W. GEO-METRICS Ⅱ, Addison-Wesley Publishing Company. 1992.

Goldberg, D. E. Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison-Wesley. 1989.

Guan, Q., Liu, J. H. and Zhong, Y. F. A Concurrent Hierarchical Evolution Approach to Assembly Process Planning, International Journal of Production Research, 40 (14), pp.3357-3374. 2002.

Gui, J. K. and Mantyla, M. Functional Understanding of Assembly Modeling, Computer Aided Design, 26(6), pp.435-451. 1994.

Henrioud, J. M. and Bourjault, A. LEGA: A Computer-aided Generator of Assembly Plans. In L. Homem de Mello and S. Lee (eds), Computer-Aided Mechanical Assembly Planning, Chapter 8, pp. 191-215. Norwell, MA: Kluwer Academic. 1991.

Hirai, J. I. and Nagata, T. Agent-oriented and Distributed Assembly Task Planning for Multiple Manipulators. In IEEE International Conference on Intelligent Robots and Systems, 1994, vol.1, pp. 113-118.

Holland, W. V. and Bronsvoort, W. F. Assembly Features in Modeling and Planning, Robotics and Computer Integrated Manufacturing, 16, pp.277-294. 2000.

Homen de Mello, L. S., and Sanderson, A.C. Task sequence planning for assembly. Proc. 12th world Congress on Scientific Computation - IMACS, Paris, France, July 1988, vol.3, pp.390-392.

Homem De Mello, L. S. and Sanderson, A. C. A Correct and Complete Algorithm for the Generation of Mechanical Assembly Sequence, IEEE Transactions on Robotics and Automation, 7(2), pp. 228-240. 1991.

Homem De Mello, L. S. and Sanderson, A. C. AND/OR Graph Representation of Assembly Plans, IEEE Transactions on Robotics and Automation, 6(2), pp.188-199. 1990.

Hong, D. S. and Cho, H. S. A Genetic-algorithm Based Approach to the Generation of Robotic Assembly Sequence, Control Engineering Practice, 7, pp.151-159. 1999.

Ishibuchi, H. and Murata, T. A Multi-objective Genetic Local Search Algorithm and Its Application to Flowshop Scheduling, IEEE Transactions on Systems, Man and Cybernetics- Part C: Applications and Reviews, 28(3), pp.392-403. 1998.

ISO 8879: Information processing- Text and office systems- Standard Generalized Markup Language (SGML), Geneva Switzerland, 1986.

Kim, K. Y., Wang, Y., Muogboh, O. S. and Nnaji, B. O. Design Formalism for

Collaborative Assembly Design, Computer Aided Design, 36(9), pp. 849-871. 2004.

Laperriere, L. and ELMaraghy, H. A. Planning of Products Assembly and Disassembly, Annals of the CIRP, 41(1), pp.5-9. 1992.

Lazzerini, B. and Marcelloni, F. A Genetic Algorithm for Generating Optimal Assembly Plans, Artificial Intelligence in Engineering, 14, pp.319-329. 2000.

Lee, K. and Andrews, G.. Inference of Positions of Components in an Assembly: Part 2, Computer Aided Design, 17(1), pp.20-24. 1985.

Lee, S. and Yi, C. Evaluation of Assemblability Based on Statistical Analysis of Tolerance Propagation. In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 1995a, pp 256-261.

Lee, S. and Yi, C. Statistical Measure of Assemblability under the Propagation of Tolerance and Clearance. In Proceedings of the IEEE International Symposium on Assembly and Task Planning, 1995b, pp. 94-99.

Lee, S. and Yi, C. Tolerance Analysis for Multi-chain Assemblies with Sequence and Function Constraints. In Proceedings of the 1997 IEEE International Conference on Robotics and Automation, 1997, pp. 927-932.

Leung, Y. W. and Wang, Y. P. Multiobjective Programming Using Uniform Design and Genetic Algorithm, IEEE Transactions on Systems, Man and Cybernetics- Part C: Applications and Reviews, 30(3), pp.293-304. 2000.

Li, X. W., Liu, J. H. and Wang, J. F. Web-based Collaborative Assembly Planning, Journal of Computer-aided Design & Computer Graphics, 15 (3), pp.348-354. 2003.

Lin, C.Y., Huang, W.H., Jeng, M.C. and Doong, J.L. Study of an Assembly Tolerance Allocation Model Based on Monte Carlo Simulation, Journal of Materials Processing Technology, 70, pp 9-16. 1997.

Lit, P. D., Latinne, P., Rekiek, B. and Delchambre, A. Assembly Planning With an Ordering Genetic Algorithm, International Journal of Production Research, 39(16), pp.3623-3640. 2001.

Mervyn, F., Kumar, A. S., Bok, S. H. and Nee, A.Y.C. Developing Distributed Applications for Integrated Product and Process Design, Computer Aided Design, 36(8), pp.679-689. 2004.

Michalewicz, Z. Genetic Algorithms+ Data Structures=Evolution Programs. Berlin, Heidelberg: Springer-Verlag. 1996.

Nevis, J. L. and Whitney, D. E. Assembly Research. Automatica, 16, pp. 595-613.

1980.

Ngoi, B.K.A. and Ong, J.M. A Complete Tolerance Charting System in Assembly, International Journal of Production Research, 37(11), pp. 2477-2498. 1999a.

Ngoi, B.K.A. and Ong, J.M. Optimum Tolerance Allocation in Assembly, International Journal of Advance Manufacturing Tehnology, 15, pp. 660-665. 1999b.

Noort, A., Hoek, G. F. M. and Bronsvoort, W. F. Integrating Part and Assembly Modeling, Computer Aided Design, 34(12), pp.899-912. 2002.

Open CASCADE, 2004, Open CASCADE Technology, Open CASCADE S.A. (http://www.opencascade.com).

Pahng, G. F., Park, S. and Ha, S. Ontology-based Design Knowledge Management Using XML. In Proceedings of Geometrical Modeling and Computer Graphics in the World Wide Web Era, Koera-Israel Bi-National Conference, 1999, korea, pp.285-289.

Park, S. and Lee, K. Verification of Assemblability Between Toleranced Parts, Computer Aided Design, 30(2), pp. 95-104. 1998.

Paul, P.R. Robot Manipulators: Mathematics, Programming, and Control. The MIT Press. 1981.

PTC, 2004, Pro/ENGINEER Wildfire, Parametric Technology Corporation. (http://www.ptc.com).

Rabemanantssoa, M. and Pierre, S. An Artificial Intelligence Approach for Generating Assembly Sequence in CAD/CAM, Artificial Intelligence in Engineering, 10, pp.97-107. 1996.

Rabemanantosa, M. and Pierre, S. A Knowledge-based Approach for Robot Assembly Planner. In Proc. Canadian Conference on Electrical and Computer Engineering, IEEE, 1993a, Vancouver, vol. 2. pp.829-832.

Rabemanantosa, M. and Pierre, S. A Knowledge-based System for Assembly Process Planning. In Proc. IEEE SESS'93, Int. Conf. on Artificial Intelligence, 1993b, Brighton, pp.262-72.

Rembold, U., Blume, C. and Dillmann, R. Computer-integrated Manufacturing Technology and Systems. New York, NY: Marcel Dekker. 1985.

Rezayat, M. The Enterprise-web Portal for Life-cycle Support, Computer Aided Design, 32(1), pp. 85-96. 2000.

Ridge Tool Company/Elyria, Ohio, U.S.A. http://www.ridgid.com/catalogdocs/

k7500.pdf

Senin, N., Groppetti, R. and Wallace, D. R. Concurrent Assembly Planning with Genetic Algorithms, Robotics and Computer Integrated Manufacturing, 16, pp.65-72. 2000.

Shah, J. J. and Rogers, M. T. Assembly Modeling as an Extension of Feature-based Design, Research in Engineering Design, 5, pp.218-237. 1993.

Shyamsundar, N. and Gadh, R. Internet-enabled Collaborative Product Design with Assembly Features and Virtual Design Space, Computer Aided Design, 33(9), pp.637-651. 2001.

Smith, G. C. and Smith, S. S. F. An Enhanced Genetic Algorithm for Automated Assembly Planning, Robotics and Computer Integrated Manufacturing, 18, pp.355-364. 2002.

Sodhi, R. and Turner, J.U. Relative Positioning of Variational Part Models for Design Analysis, Computer Aided Design, 26(5), pp. 366-378. 1994.

Sodhi, R. and Turner, J. U. Representing Tolerance and Assembly Information in a Feature-based Design Environment. In Proceedings of the ASME Design Automation

Conference, DE-Vol. 32-1, 1991, Miami, Florida, USA, pp.101-108.

Solimanpur, M., Vart, P. and Shankar, R. A Multi-objective Genetic Algorithm Approach to the Design of Cellular Manufacturing Systems, International Journal of Production Research, 42(7), pp.1419-1441. 2004.

Srikanth K., Liou F.W. and Balakrishnan S.N. Integrated Approach for Assembly Tolerance Analysis, International Journal of Production Research, 39(7), pp. 1517-1535. 2001.

Toshiki, M. and Cutkosky, R. M. Agent-based Collaborative Design of Parts in Assembly. In Proceedings of the ASME Design Engineering Technical Conference, September 13-16, 1998, Atlanta, Georgia, USA.

Treacy, P., Ochs, J.B., Ozsoy, T.M. and Wang N.X. Automated Tolerance Analysis for Mechanicl Assemblies Modeled with Geometric Features and Relational Data Structure, Computer Aided Design, 23(6), pp. 444-453. 1991.

Tseng, H. E. and Kweili, R. A Novel Means of Generating Assembly Sequences, Journal of Intelligent Manufacturing, 10(5), pp. 423-435. 1999.

Wang, H., Pramanik, N., Roy, U., Sudarsan, R., Sriram, R.D. and Lyons K.W. A

Scheme for Transformation of Tolerance Specifications to Generalized Deviation Space for Use in Tolerance Synthesis and Analysis. In Proceedings of ASME Design Engineering Technical Conference, 2002.

Wang, J.F., Liu, J.H. and Zhong, Y.F. Collaborative Assembly Planning System Under Web Environment, Computer Integrated Manufacturing System, 10(1), pp. 83-87. 2004.

Whitney, D. E. and Gilbert, O. L. Representation of Geometric Variations Using Matrix Transforms for Statistical Tolerance Analysis in Assemblies. In Proceedings of IEEE International Conference on Robotics and Automation, 1993, pp 314-321.

Yang, C.C. and Naikan, V.N.A. Optimum Design of Component Tolerance of Assemblies Using Constraint Networks, International Journal of Production Economics, 84, pp.149-163. 2003.

Ye, X. G., Fuh, J. Y. H. and Lee, K. S. Automated Assembly Modeling for Plastic Injection Moulds, International Journal of Advanced Manufacturing Technology, 16, pp.739-747. 2000.

Yin, Z. P., Ding, H., Li, H. X. and Xiong, Y. L. A Connector-based Hierarchical Approach to Assembly Sequence Planning for Mechanical Assemblies, Computer Aided Design, 35(1), pp.37-56. 2003.

Zha, X. F., Lim, S. Y. E. and Fok, S. C. Development of Expert System for Concurrent Product Design and Planning for Assembly, International Journal of Advanced Manufacturing Technology, 15, pp.153-162. 1999.

Publications Produced From the Thesis

Journal Paper

1. Lu C., Fuh J.Y.H. and Wong Y.S., Evaluation of Product Assemblability in Different Assembly Sequences Using the Tolerancing Approach, *International Journal of Production Research*, vol. 44, no.23, pp. 5037-5063, 2006.

2. Lu C., Wong Y.S. and Fuh J.Y.H., An Enhanced Assembly Planning Approach Using a Multi-objective Genetic Algorithm, *Proceedings of the Institution of Mechanical Engineers, PartB, Journal of Engineering Manufacture*, vol.220, no.2, pp.255-272, 2006.

3. Lu C., Fuh J.Y.H., Wong Y.S., Qiu Z.M., Li W.D. and Lu Y.Q., Design Modification in a Collaborative Assembly Design Environment, *Transactions of the ASME, Journal of Computing and Information Science in Engineering*, vol.6, no.2, pp.200-208, 2006.

4. Lu C., Fuh J.Y.H., Wong Y.S., Li W.D. and Lu Y.Q., Feature-based Design Modification in Co-assembly Design, *Computer-Aided Design and Applications*, vol.1, no.1-4, pp.411-420, 2004.

Conference paper

1. Lu C., Fuh J.Y.H. and Wong Y.S., "A multi-objective genetic algorithm with fuzzy weights distribution", *Proceedings of the International Symposium on Flexible Automation*, Osaka, Japan, July 10-12, 2006, pp. 225-229.

2. Lu C., Fuh J.Y.H. and Wong Y.S., Evaluation of Product Assemblability in Different Assembly Sequences, *Proceedings of the 6th IEEE International Symposium on Assembly and Task Planning*, Montreal, Canada, July 19-21, 2005, pp.168-173.