Disassembly Sequence Planning for Product Maintenance

By

Yongtao Luo

A Thesis Submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Mechanical Engineering
University of Manitoba
Winnipeg, Manitoba

Abstract

Environmental and sustainable issues have brought more and more attention to industries in product design and manufacturing. It is important for a product to meet its lifecycle requirements in repairing, replacing and recycling. Disassembly is required in product maintenance and recycling. An efficient disassembly plan can reduce the cost of product maintenance and minimize the product repair time. This thesis introduces an efficient method for selective disassembly planning for the need of product maintenance and recycling to reduce the product operation time and cost. The method is based on an efficient product representation and effective sequence searching. It considers the product structure, removing direction of components, operation constraints and complex in the product representation and sequencing planning. Examples are used to verify the proposed method. Challenges and further work are also discussed.

Acknowledgments

I would like to take this opportunity to thank my advisor, Dr. Qingjin Peng. Without his constant support and great patience, I would not be able to complete this thesis. His encouragement is very important for me to complete my degree.

I would also like to give my thanks to my colleagues in the Virtual Manufacturing Lab and friends who helped me with my research and thesis.

I would like to thank Canadian NSERC Discovery Grant for the funding support.

Finally, I would like to thank my parents. They are the most important people in my life. Without their support and understanding, it is impossible for me to obtain this degree.

Contents

List of Tables

List of Figures

List of Abbreviations

DSP	Disassembly Sequence Planning
EOL	End of Life
EEE	Electrical and Electronic Equipment
CAPP	Computer-aid Path Planning
DF	Disassembly Matrix for Fasteners
DC	Disassembly Matrix for Components
MF	Motion Constraints for Fasteners
MC	Motion Constraints for Components
GA	Genetic Algorithm
EA	Evolutionary Algorithms
SA	Simulated annealing
ACOA	Ant Colony Optimization Algorithm
BOM	Bill of Materials
DFIG	Disassembly Feasibility Information Graph
GCS	Global Coordinate System
LCS	Local Coordinate System

Chapter 1

Introduction

1.1 Background

Huge pressure has been brought on environment due to the rapid economic growth since the 1960s. Manufacturing industries have the most negative impact on environment [1]. For the sake of more profit, manufacturing companies produce more products and reduce the lifetime of products. As a consequence, lots of resources are wasted. For example, about 5 millions of computers are discarded annually in China, and million tons of harmful chemical elements such as lead and mercury are transported to waste-treatment plants [1, 2]. Manufacturing industries should make adjustments to reduce the environmental stress and pursue a sustainable development.

In order to solve the environmental and resources problems essentially, the definition of green manufacturing, as shown in Fig. 1-1, was proposed by Lu et al. [2]. Green manufacturing considers the lifecycle activities of products such as design, manufacturing and maintenance to guarantee a minimum influence on environment and a maximum utilization of resources without sacrifices of product function, quality and production cost [3]. Recycling and reusing of scrapped products is the major measure of green manufacturing, which can decrease the environmental pollution problems greatly [4].

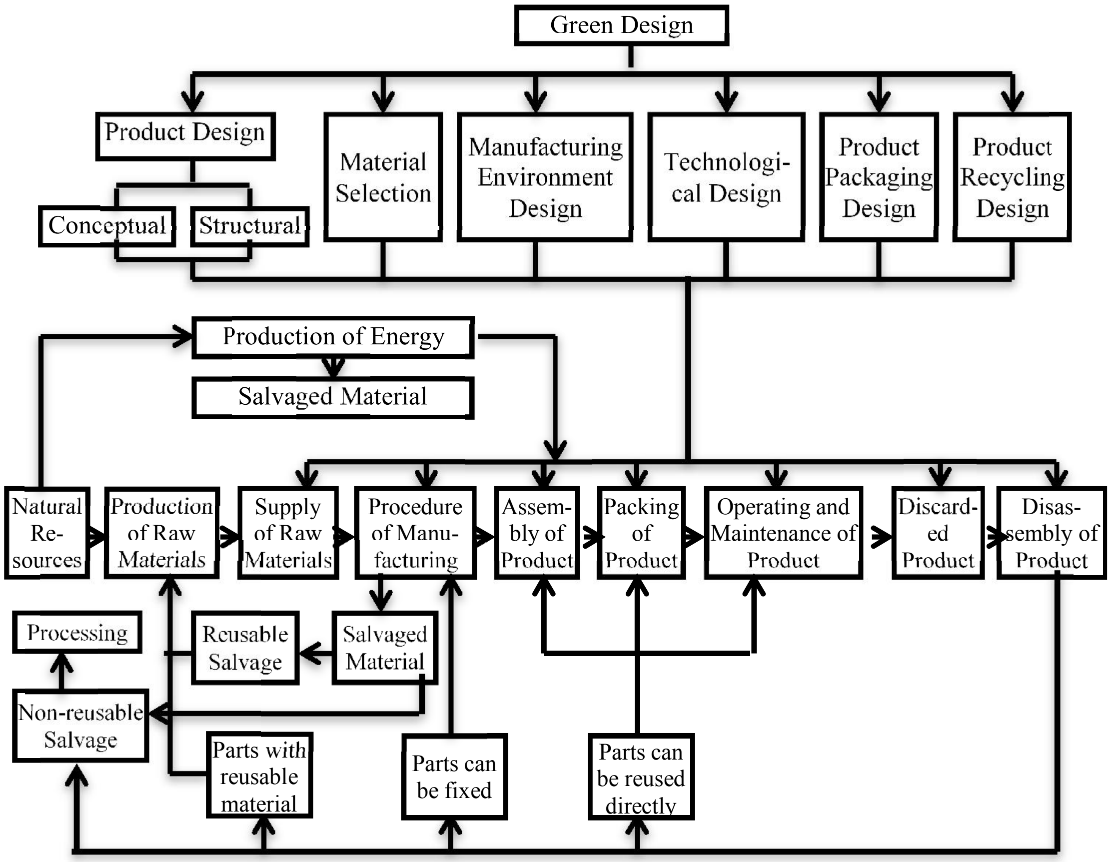


Fig. 1-1 Green Manufacturing System [2]

Disassembly is a basic and important operation in product maintenance and recycling. The purpose of disassembly operation is to collect and recycle valuable materials and reusable parts of products by removing parts and subassemblies systematically. Harmful chemical elements are also collected and separated for further treatments. Studies on disassembly sequence planning (DSP) are the foundation of product disassembly analysis.

The DSP is to find and express the optimal or near optimal sequences of the product depending on the principles.

Besides environmental issues, importance of DSP in product design stage has also been identified by many researchers for product maintenance and recycling [5]. Manufacturing industries have increased their responsibilities for the end-of-life products recently. They not only manufacture products but also consider the service to repair, replace or recycle the products. A product has to be taken into apart in the repairing or recycling process. Usually, a product can be disassembled in various sequences. Disassembly sequences have strong influences on the difficulty level of disassembly process and disassembly cost [6]. Consequently, finding an optimal or near optimal disassembly sequence is one of the most effective and important operations for maintenance of most industrial products. Pnueli and Zussman [7] indicated that the majority of product recycling cost is determined at the product design stage, while only 10% to 20% of the recycling cost depends on the product recycling process. They also expressed that disassembly is the last and most important step in product value-added recovery operations. In conclusion, determining an efficient disassembly sequence is a main task of disassembly planning. Since time is a crucial factor related to the disassembly cost, a good disassembly planning should be able to reduce time required in the product disassembly process [8].

Based on different disassembly tasks in a product lifecycle, disassembly can be classified into three categories: complete disassembly, incomplete disassembly, and selective disassembly. The complete disassembly separates the entire product into components or subassemblies, which is similar to the reverse processing of an assembly. However, incomplete disassembly only separates some components of the product, which is usually used to recover the original function of the end-of-life products. In contrast, the selective disassembly is mainly employed for the purpose of product maintenance or component upgrading in a product. Selective disassembly removes particular components in a product, the components may be damaged or need to be upgraded. Compared to the incomplete disassembly, the target component of the selective disassembly is already known. Owing to the importance of the selective disassembly in the product maintenance and repairing, we will mainly consider the operation planning for the selective disassembly in this thesis.

1.2 Research Objectives

Disassembly sequence planning is a popular research topic in recent years. Lots of researchers are studying related areas and each one of them pursues a specific aspect. Some methods are only effective to a certain kind of models or products. Some of them may have the problems of computation-intensive and time consuming. Whereas, there are also some systems developed based on the existing 2D or 3D CAD software such as ProE, UG, SolidWorks et al. However, there is not an effective method for the selective disassembly planning.

In this thesis, a new method based on an efficient model representation method and an optimized searching method will be proposed in order to generate a feasible disassembly sequence in a short time, especially for selective disassembly sequence planning. Factors affecting the difficulty level to disassemble a product will be considered, which guarantees a reliable result and provides a realistic meaning for industries. Ultimately, simulation of the disassembly procedure will be provided using a professional simulation software called Eon Studio.

1.3 Thesis Outline

Chapter two reviews three commonly used model representation methods as well as two types of sequences searching methods. By stressing the pros and cons, suitable methods are selected for solving the DSP problem.

Chapter three introduces the proposed representation method, a multi-layer constraint matrix representation method. An example of a gearbox is used to explain the efficiency of the method. Traversing searching method is also used in this chapter.

Traversing searching method is not an optimized method. In Chapter four, the ant colony searching method is used instead of the traversing searching method. Evaluation criteria and evaluation methods are also introduced in this chapter, which helps to find an near optimum disassembly sequence. A searching program is developed based on the proposed method, which improves the automatic search level of the method.

Simulation is used as a presentation tool in this research. Chapter five explains the related knowledge on generating simulation for product models. Local coordinate system and global coordinate system are also introduced in this chapter.

Three cases are used in Chapter six to further explain the proposed method and to prove the efficiency of the method.

Chapter seven concludes the thesis and identifies the contributions of the research. Future work is also discussed.

2 Chapter 2

Literature Review

Disassembly planning decides the sequence of a product disassembly process as shown in Fig. 2-1. A disassembly sequence is generated based on the product's geometric structure, components' relationships, the selective part removing constraints from other parts, types of fasteners, tools required in the operation, space accessibility for the tool application, the complexity of parts, and fasteners to be removed in order to disassemble the selected part et al. All of these factors will affect the feasibility of a disassembly plan. As planning starts based on design information, the method of a product representation will directly affect the way of disassembly planning. Product representation and disassembly sequencing have a close relation in disassembly planning [9]. In this chapter, three commonly used representations of product models will be introduced, namely, graph based representation, constraint matrix and adjacency table representation. Two types of sequence planning methods are also discussed based on the graph representation and heuristic algorithms.

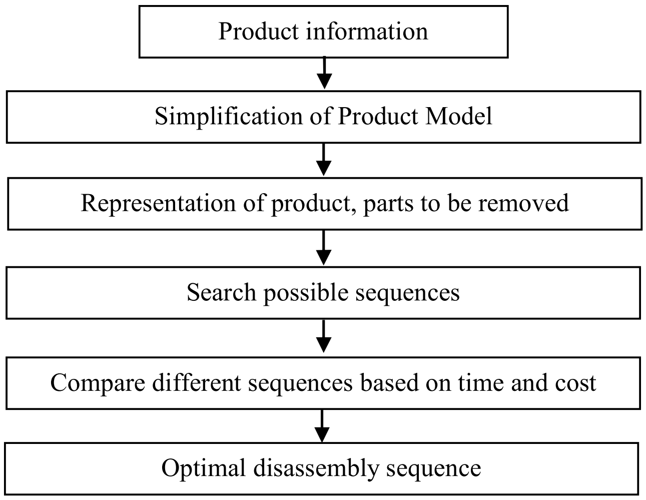


Fig. 2-1 Flowchart of Disassembly Planning

2.1 Representation of Product Model

2.1.1 Graph Based Representation

AND/OR graph is a widely used graph-based modeling method, and it is a basic approach of product representation for the product assembly or disassembly analysis [10]. AND/OR graph consists of two parts: nodes and hyperarcs. The nodes stand for components or subassemblies of a product, while disassembly tasks are represented by the hyperarcs. Two or more components or subassemblies can join together to form a more complicated subassembly [11]. The nodes of the graph are labeled as either "AND" or "OR" branches. The vertical links are "AND" relations. The "AND" successors are subassemblies that are split from the upper node, while nodes' links in the same level are "OR" relations. "OR" branches indicate alternative subassemblies, any one of which can generate a separate sequence. An AND/OR graph can explain all possible sequences for disassembling a product completely, and a model will be used to explain this representation below.

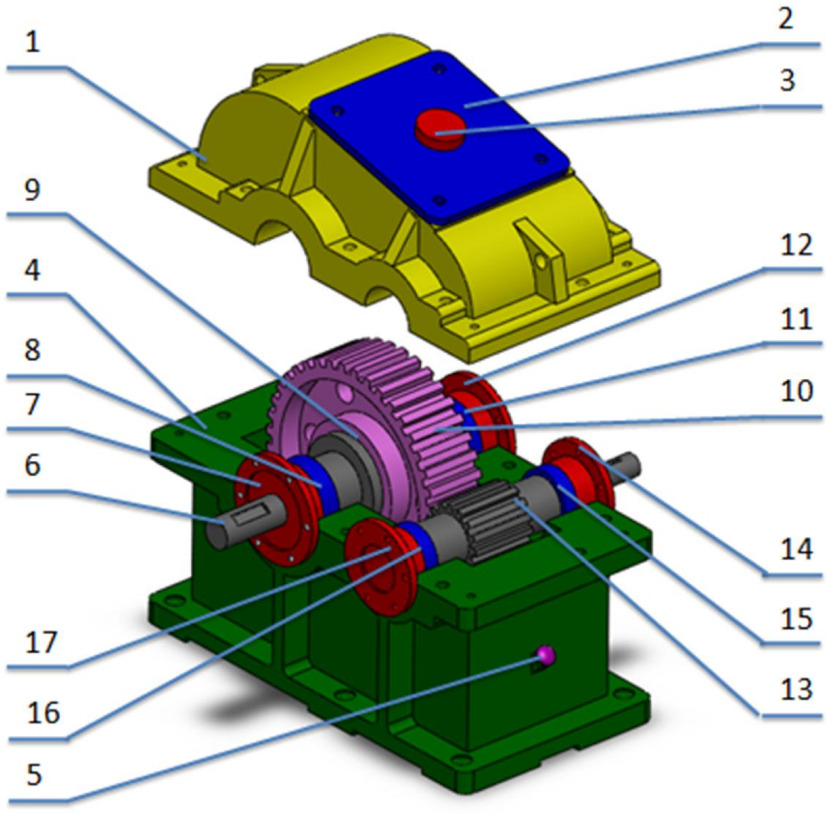


Fig. 2-2 Single-speed Reduction Gearbox

Fig. 2-2 is the model of a simplified single-speed reduction gearbox. Usually, a gearbox consists of a lot of parts, for example, lock washer, seal gasket, water drain valve et al. However, in order to simplify the explanation, all small parts of the gearbox are ignored in this simplified model. 17 parts of the gearbox are considered as components of the 4 subassemblies: the upper body (LS1), the reducer box (LS2), the first transmission axis (LS3) and the second transmission axis (LS4). Table 2-1 lists parts of the model, and this model will be used for the comparison of different product representations. Fig. 2-3 is the AND/OR graph representation for this model.

Table 2-1 Parts of the Gearbox Model

Part No.	Subassembly No.	Part Name
1	LS1 (A)	Upper Body
2	LS1 (A)	Upper Cover
3	LS1 (A)	Vent Hood
4	LS2 (B)	Reducer Box
5	LS2 (B)	Oil-level Pointer
6	LS3 (C)	Transmission Axis
7	LS3 (C)	Cover 1
8	LS3 (C)	Bearing 1
9	LS3 (C)	Key
10	LS3 (C)	Gear 1
11	LS3 (C)	Bearing 2
12	LS3 (C)	Cover 2
13	LS4 (D)	Gear Axis
14	LS4 (D)	Cover 3
15	LS4 (D)	Bearing 3
16	LS4 (D)	Bearing 4
17	LS4 (D)	Cover 4

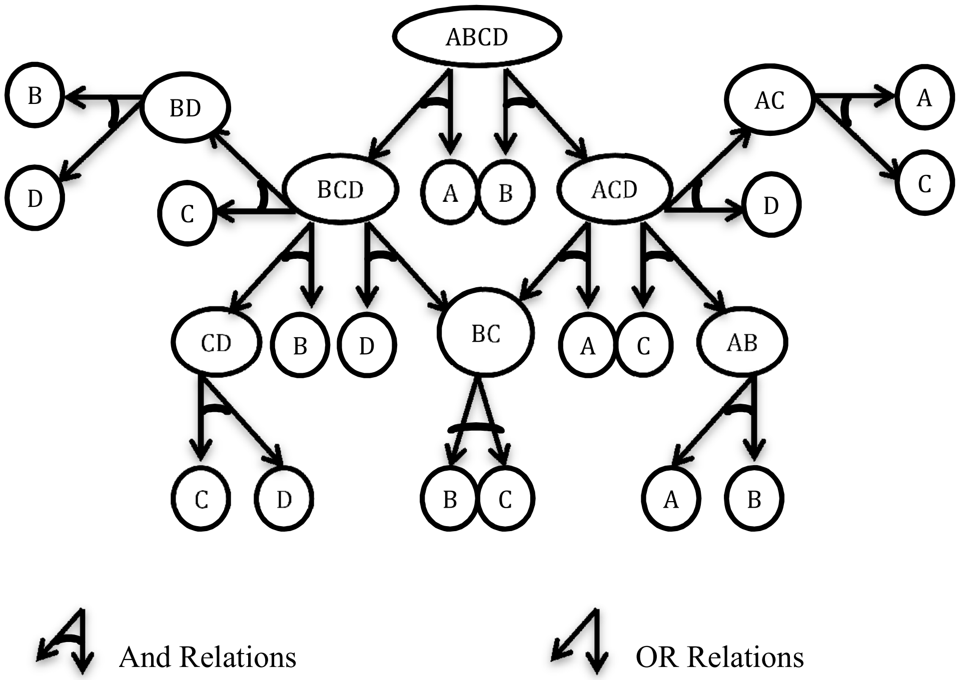


Fig. 2-3 AND/OR Graph of the Gearbox

AND/OR graph was used by Mello and Sanderson [12] for the representation of a ten-part space-based satellite equipment. The objective of this representation was to solve maintenance and repair problems of robots in hazardous environments such as space, undersea and nuclear power plants. This representation has been proved to be efficient in increasing the assembly flexibility by allowing an intelligent robot to pick more convenient course of actions according to instantaneous conditions.

AND/OR graphs are still being used in recent years to solve various problems. For example, by making a combination of weight value and the AND/OR graph, a weighted AND/OR graph was proposed by Min et al. [13]. This graph was successfully used to deal with a disassembly problem of an 11-part product, which expounded and proved that weighted AND/OR graph is efficient for representation product structure and element constraints. AND/OR graphs made a good foundation for the mechanical product automatic disassembly decision.

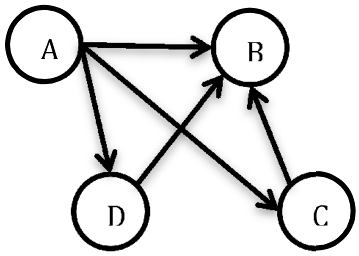


Fig. 2-4 Adjacency Graph of the Gearbox

Adjacency Graph is a visualized representation of relationships between parts. For example, Fig. 2-4 shows an adjacency graph of the gearbox model in Fig. 2-2. Nodes of the graph represent parts of the product. Relationships between connected parts are represented by directed lines or undirected lines. A directed line or undirected line in adjacency graph usually represents two relations of components (relationship of part A to part B and relationship of part B to part A), and this figure seems much neater than the AND/OR Graph representations.

The adjacency graph was used by Song et al. [19] to represent the door of a dishwasher, their visualized representation led to the success of sequence planning. Zhang et al. [20] made a combination of adjacency graph and matrix. They also used the door of a dishwasher as an example to prove the feasibility of their method.

Compared to other graph-based representations, the adjacency graph focuses more on constraints between parts. While AND/OR graphs and some other graphs concentrate on the product structure. Sharing the same disadvantage as other graph-based representations, the adjacency graphs result in a poor automatic level when searching for sequences. Rather than using directly for sequence planning, adjacency graphs are usually transformed into adjacency tables or adjacency matrix for computers to calculate and search for possible sequences.

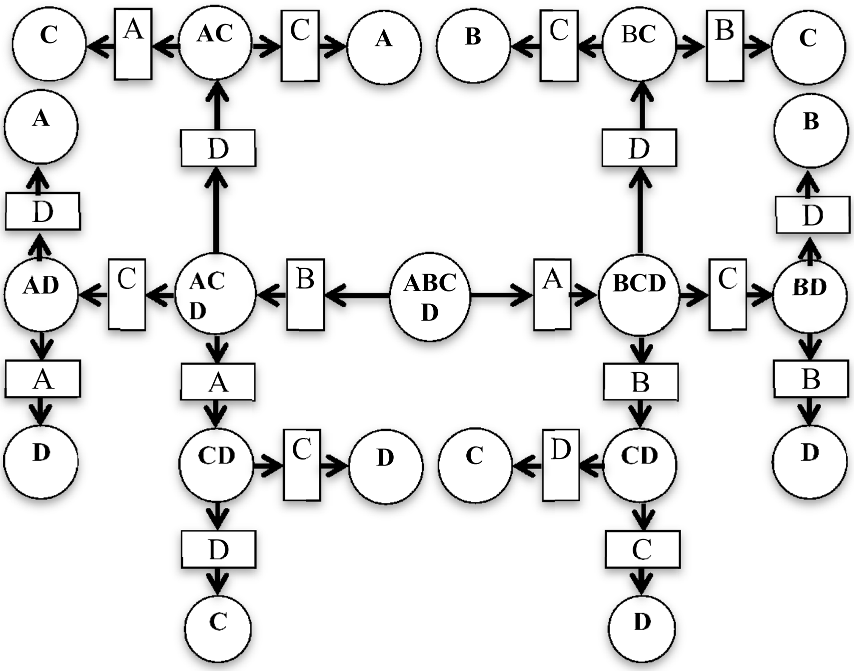


Fig. 2-5 Petri Net Graph of the Gearbox

There are also some other kinds of representations that are the variants of AND/OR graphs, for example, Petri net graph. A Petri net mainly consists of tokens, places, transitions, and arcs. Tokens stand for real-world products, places represent an assembly and subassemblies, while transitions and arcs equal to actions and hyperarcs in AND/OR graphs. Arcs run from a place to a transition or vice versa. They never run between places or between transitions. The places from which an arc runs to a transition are called input places of the transitions, while places to which arcs run from a transition are called output places of the transition [14]. Fig. 2-4 is the Petri net graph of the gearbox. In 2010, Petri net graph was used to cope with the disassembly and recycling problems in end-of-life (EOL) electrical and electronic equipment (EEE) by Tsai and Wang [15]. An ADSL router was taken as an example to prove the feasibility of the representation.

The AND/OR graph and its variants are able to represent relationships among different parts or subassemblies of a product, and AND/OR graphs manage to provide a super easy way for finding all possible sequences. But as it is discussed in literatures [10, 11, 12, 13, 14, 15], the problem of this method is that the graph is difficult to generate automatically. It can only be used in dealing with models with simple structure and few parts. Otherwise, the representation may become very complicated. However, real-life manufacturing products usually consist of lots of parts, therefore, AND/OR graphs and other graph-based representations are impractical for dealing with the product with a large number of components.

2.1.2 Constraint Matrix

Constraint matrices are also very popular representation methods of product. They can be used to represent product structure information, parts constraints and precedence. Usually, graph-based representations are converted into matrix-based representations. There are several different kinds of constraint matrices. Some of them are used to represent relationships between every two parts, while others only consider the contact parts. Some methods consider only one direction, while others consider two or more directions. However, no matter which matrix is to be implemented, there is always a rule to follow. Constraint matrix representation provides a possibility for dealing with DSP problems with numerous parts, and it will also lead to a higher automatic level. Besides the generation of the matrix, it is also convenient for computers to deal with matrix formats, instead of graphs, when searching possible sequences. Fig. 2-6 is a constraint matrix representation for the model in Fig. 2-2. The number “1” in the matrix stands that the corresponding part cannot be moved in the positive direction because of another part. On the contrary, the “-1” indicates the negative direction restriction. “0” means there are no restriction relationships between the corresponding two parts.

Chung and Peng developed a disassembly system based on the constraint matrix representation in 2006, which improved the automatic level of sequence planning problem [16]. Wu and Peng also solved the product maintainability evaluation problem based on an improved constraint matrix in 2009, which contributed to the efficiency of product life-cycle operations and the reduction of operation cost [17].

Although constraint matrix representations have many advantages, they also have disadvantages. Usually, the matrix records relationships between every two parts even though there is no constraint between the two parts. In other words, some recorded data in the matrix do not have to be considered or recorded, for instance the number “0” in the matrix shown in Fig. 2-4. It brings the problem of storage space waste and it increases the searching space.

	A	B	C	D
A	0	1	1	1
B	-1	0	-1	-1
C	-1	1	0	0
D	-1	1	0	0

Fig. 2-6 Constraint Matrix of the Gearbox

2.1.3 Adjacency Table

A typical adjacency table for the relationships of parts in Fig. 2-2 is shown in the Fig. 2-7 . The data in the first column of the table represent parts of the model, while the data in the following column represent parts that have constraint relationships with the part in the first column. It is obvious that this representation is more concise than the constraint matrix representation. Adjacency tables are usually translated from a graph. Hopcroft [18] illustrated the advantages of adjacency tables when dealing with traversal problems in his book.

Comparing the constraint matrix representation and adjacency table, the former shows much clearer relationships between elements, while the latter saves more memory for computers. Considering clearer constraint relationships are extremely important when dealing with sequence planning problems, the constraint matrix is believed to be a better representation than the adjacency table in this situation.

A	BCD
B	N/A
C	B
D	B

Fig. 2-7 Adjacency Table of the Gearbox

Evaluating all aspects of the DSP problem, the constraint matrix is considered to be easy for generation, implementation and calculation. Thus, it is chosen as the ultimate representation method for the product model in this research.

2.2 Sequence Planning

2.2.1 DSP Based on Graph Representation

As it has been mentioned in the previous sections, the AND/OR graph can manage to generate all possible operations for a product disassembly [17], and each route from an upper node to the lower node is a unique disassembly sequence.

Gao et al. [21] took a partial disassembly of PC as an example to illustrate the disassembly planning for recycling. They also used the AND/OR graph method in concurrent design to evaluate the disassembly-recycling property of the PC product.

Comparing to AND/OR graphs, Petri nets representation has extensively applications in different engineering areas. Fuzzy Petri nets were used by Cao et al. [22] to solve the problem of sequence planning in a robotic system. The Petri nets proved to be an efficient tool for modeling and representing feasible and complete process schemes. Based on the Petri nets, Moore et al. [23] developed a new approach to search the possible disassembly sequence automatically. They used a cost function to select the near optimum sequence. This approach can be used not only in simple products but also in complicated products with a large amount of elements. Zha [24] developed a novel methodology for the integration of design and assembly planning on the basis of Petri nets. Hsieh [25] used an optimized Petri nets to tackle the assembly or disassembly processes with unreliable resources.

Although a lot of research activities have been done based on AND/OR graphs, Petri net graphs and other graph-based representations, these methods have some disadvantages.

The concentration of the entangled hyperarcs is high [23], and they require enormous search space for achieving a near optimum scheme. In conclusion, graph-based methods are computationally complex, which may cause combinatorial explosion problems, and they are nearly impossible to handle disassembly path scheming or maintenance for real-life products with numerous elements [21, 23].

2.2.2 Heuristic Algorithms for Disassembly Planning

Heuristic algorithms usually use functionality, performance, manufacturability, reliability, eco-compatibility etc. as criteria to select the near optimum disassembly sequence [26]. Comparing with the graph-based searching method, heuristic algorithms are much more efficient, which is practical for solving Computer-aided path planning problems. Normally, matrices are introduced to represent constraints, precedence relationships, interferences and structure information of a product [27]. A lot of different matrices have been defined to express product information such as part constraints, fasteners and interferences [28]. These matrices can then be used to search feasible schemes.

In order to tackle the problem of gripper selection problems in assembly cells, Pedrazzoli et al. [29] developed a set of rules to search a near optimal assembly sequence. Once the sequence is decided, required grippers or robots can be chosen. A set of related problems can be addressed later on. With the help of the rule-based method, Zhang et al. [30] carried out a research on assembly modeling and sequence planning. They succeeded in the assembly of a ball valve. Smith et al. [31] developed a rule-based recursive selective disassembly method for finding a near-optimal disassembly sequence for products. They introduced four matrices and five disassembly rules. The four matrices are: disassembly matrix for fasteners (DF), disassembly matrix for components (DC), motion constraints for fasteners (MF) and motion constraints for components (MC). The five rules define the disassembly order or priorities among components. They then introduced a cost function in the method to find the near optimal disassembly sequence with the lowest cost. By using this method they solved selective disassembly sequence planning problems with both single-target component and multiple-target components.

The Heuristic algorithms concern the geometric and topological information of the components. Compared to other existing disassembly methods, it is easier to be implemented for general products with less time to find a better conclusion. Genetic algorithms and ant colony searching methods belong to the large class of Heuristic algorithms [32].

Genetic algorithm (GA) has many applications in solving optimal search problems, and it mimics the process of natural evolution processes [11]. Using the genetic algorithm, a number of strings are generated. The fittest one survives to be used for further analysis. Genetic algorithms can provide the near optimum disassembly sequence in a short time. This method can consider precedence relationships and additional constraints in the product structure [10].

Genetic Algorithm was used for the study of the disassemble planning of a torch by Li et al. [11]. The result was proved to be feasible and is in accordance with the manual disassembly of the torch. Mohanta et al. [33] combined the Genetic algorithm with the Petri nets method to solve the path planning problems. The result shows that the proposed method could handle the robot collision avoidance more effectively than other methods. Xie et al. [34] developed a new disassembly sequence planning method named simulated annealing and genetic algorithm (SAGA) by making a combination of the simulated annealing (SA) and genetic algorithm (GA). As has been mentioned, the graph theory has the problem of combination explosion, while the genetic algorithm has the problem of premature phenomena in disassembly sequence planning. However, by making the combination, both problems in the existing DSP can be solved.

Genetic algorithms are simple to be implemented, but their behaviour is difficult to understand. In particular, it is hard to know why these algorithms frequently succeed in generating solutions of the high fitness when applied to practical problems. What's more, this method also requires very expensive fitness function evaluations.

Ant colony optimization algorithm (ACOA) was initially proposed by Marco Dorigo [35] in his thesis, and he then explained the contributions of this method in his book Ant Colony Optimization. This method is designed based on the behaviour of ants seeking a path between their colony and a source of food, and the method is efficient in solving computational problems to find an optimal or near optimal path through a graph.

Based on the ant colony algorithm, Wang et al. [36] solved the disassembly problem of an automotive engine, which consists of 50 parts after the simplification. Disassembly Feasibility Information Graph (DFIG) was defined in the paper. Based on the DFIG, the problem of disassembly path planning was transformed into the problem of searching near optimal sequence in this directed, weighted graph. In conclusion, ant colony algorithm is an efficient search method and it can be used to cope with practical problems and products with lots of elements.

All methods listed above have their pros and cons. Although they were used in dealing with different disassembly planning problems, there is a lack of real-world considerations for applications of the planning solutions. One is environmental factor in the operation, such as the operation accessibility when applying tools in the disassembly, and collision detections not only between components, but also between parts and surrounding devices. Others are operators' factors and product complexity levels, which affect the feasibility of the disassembly plan. Therefore, this research improves exiting methods with an efficient product representation method for the near optimal search process considering necessary factors in product operations.

2.3 Summary

This chapter reviews three commonly used model representation methods, as well as two types of optimized searching methods. The pros and cons are also discussed for each method. Considering the real-life situation for product disassembly sequence planning problems, a proposed method will be introduced in following chapters.

3 Chapter 3

Proposed Method

This chapter introduces a method for disassembly sequence planning (DSP). The proposed method is developed based on an efficient representation method of product modeling.

3.1 Introduction

General procedures of the proposed method are similar to what is shown in Fig. 2-1. The method consists of three main parts: representation of product models, searching, and evaluating of disassembly sequences. Simulation of the near optimal result is built afterwards.

Reducing the search size of models is an important issue in this research. In order to achieve this goal, matrices with multi-layer constraints are used to represent the product structure and relationships between components based on the bill of materials.

Reducing computational complexity is as equally important as reducing the search size of models, and this will lead to a high efficiency of the method. An optimized searching method is developed based on the ant colony algorithm, and this will be introduced in the next chapter.

Criteria related to the product disassembly complexity are considered in evaluating all possible sequences, which guarantee the reliability of the final result. At last, a near optimal disassembly sequence is generated with the least time required and fewer parts to be removed in the disassembly operation of the selected part. This sequence will be used for following simulation model. Detailed procedures of the proposed method will be explained in the following parts.

3.2 Bill of Materials

Products usually consist of lots of components. For example, the component number of General Motors automobiles can be millions [37]. In the manufacturing process, depending on the product complexity, individual parts may be firstly combined into small subassemblies. These small subassemblies are then combined to form larger subassemblies. At last the larger subassemblies are assembled into the final product. Tian et al. [6] pointed out that product parts of the same subassembly are very likely to be assembled in the same assembly line. Although disassembly is not the exactly reverse processing of the assembly operation, it is still very likely for a subassembly to be disassembled as a whole in the disassembly process. In this research, based on the bill of materials, a multi-layer representation method is used to represent the product model.

A bill of materials (BOM) is a list of components, from the end item all the way down to the raw materials [38]. It explains the structure of a product. In this research, the number of layers of a product decides the number of layers of the constraint matrices when representing this product. For example in Fig. 3-1, an end product consists of two first-level subassemblies. While the first-level subassembly 1 consists of two second-level subassemblies, and each of the second-level subassembly contains two pars. Thus, there are four layer components and three layers composition relations in this bill of materials. Therefore, the end product can be represented by three layers of constraint matrices.

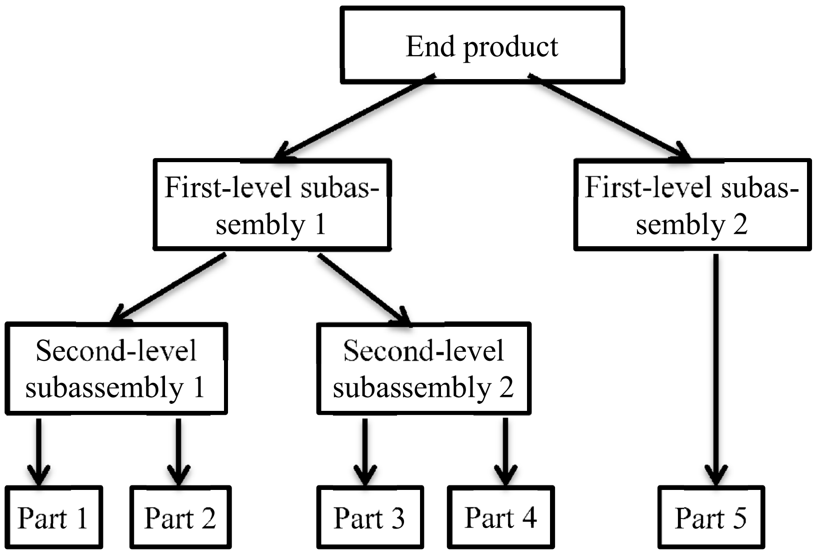


Fig. 3-1 Example for Bill of Materials

3.3 Product Representation Based on Multi-layer Constraint Matrices

The detailed procedures for generating constraint matrices are shown in Fig. 3-2, which will also be explained with an example.

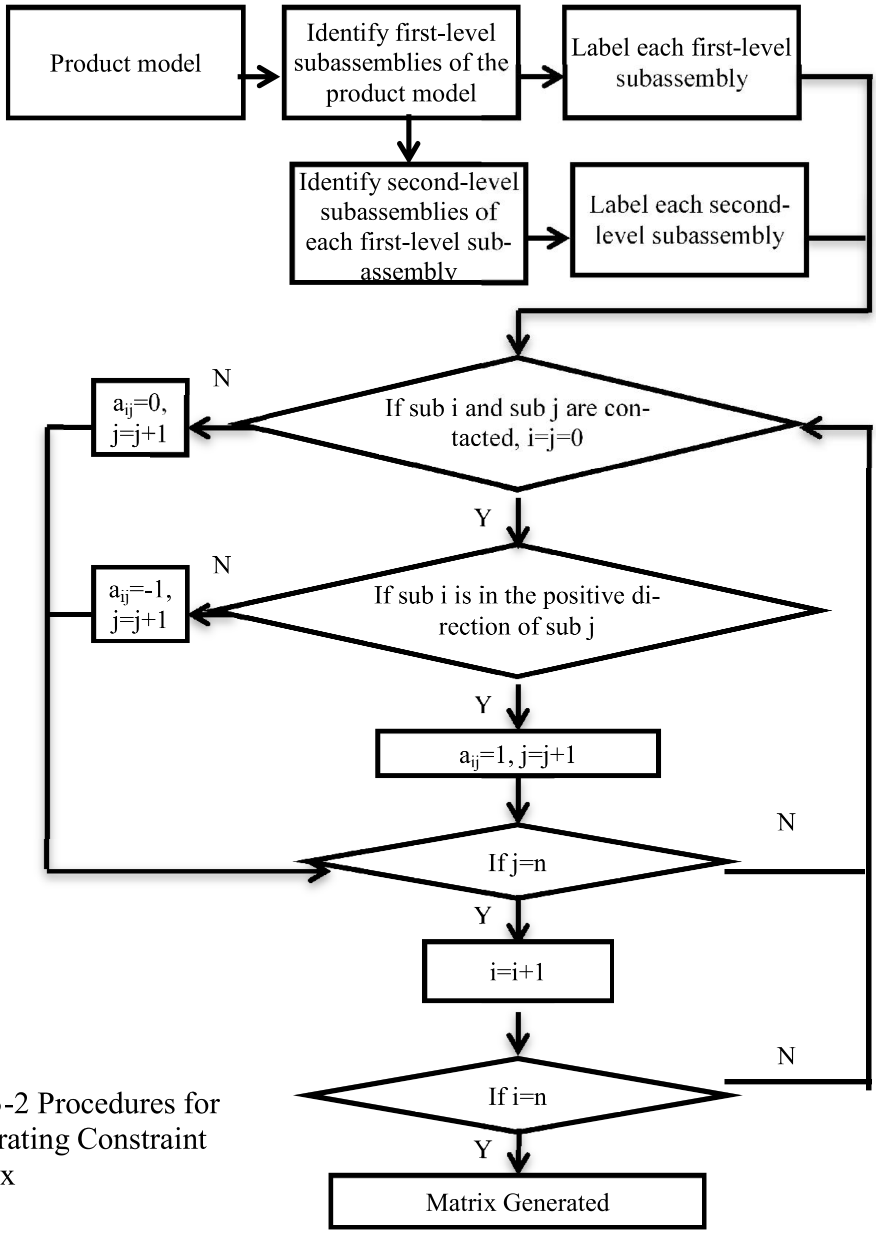


Fig. 3-2 Procedures for Generating Constraint Matrix

Design information is obtained from the product model, and each layer of subassemblies of the product model can be identified and labeled with a number. In a disassembly operation, a product is firstly separated into several first-level subassemblies, and the product structure is represented in a constraint matrix including the information about relations of the first-level subassemblies and moving constraints. This is the first-level adjacent matrix. Since most components can only be removed from one direction, only one-degree motion along axial direction of its Local Coordinate System (LCS) is considered (Details about Local Coordinate System will be introduced in Chapter 5). A general formation of the constraint matrix is represented as shown in Fig. 3-3.

$$\begin{vmatrix} a_{11} & a_{12} & \cdots\cdots & a_{1n} \\ a_{21} & a_{22} & \cdots\cdots & a_{2n} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ a_{n1} & a_{n2} & \cdots\cdots & a_{nn} \end{vmatrix}$$

Fig. 3-3 Constraint Matrix for Large Subassemblies

The matrix has an initial size of n x n, where n stands for the number of first-level subassemblies in the product. Elements of the matrix a_{ij} (i=1, 2, …, n; j=1, 2, …, n; $i \neq j$) is either 0, 1 or -1. They stand for the restriction relationships between subassemblies. The “1” indicates that one element will stop another one from moving in the positive direction. In contrast, the “-1” illustrates that the element will stop another one from moving in the negative direction. In this research, only relationships between contacted subassemblies will be concerned, otherwise a_{ij} state will be set as “0”, where there are no direct restrictions between these subassemblies. The principal diagonal elements of the matrix stand for the time required to disassemble a subassembly. For example, a_{ij}=50 (i=j=k) represents that 50 seconds is required to disassemble the subassembly k. Exact time required for disassembling a subassembly can be calculated using criteria, and the criteria will be introduced in Section 4.2.1.

First-level subassemblies contain one or several second-level subassemblies, and each of the first-level subassemblies is represented by one adjacent matrix as well. These matrices are the second level matrices. The generating method and information of these second level constraint matrices are almost the same as the first level constraint matrix. The difference is that the matrices represent relationships between small subassemblies instead of large subassemblies.

The multi-layer matrices representation of a product is efficient in solving disassembly sequence planning problems. Firstly, smaller memory is required to save multi-layer matrices than one level matrix when dealing with complexity problems. For example in Fig. 3-4, the single-speed reduction gearbox consists of 17 components totally, and these 17 components form four large subassemblies: the upper body (LS1), the reducer box (LS2), the first transmission axis (LS3) and the second transmission axis (LS4). The upper body contains three components, while the reducer box is made up of two components. The first transmission axis consists of seven components, and the second transmission axis composes of five components. Using one level matrix, *17* x *17 = 289* elements need to be recorded. However, by using the multi-layer matrices representation method, this gearbox can be represented by a set of matrices (one *4* x *4* matrix, one 3 x 3 matrix, one 2 x 2 matrix, one *7* x *7* matrix and one *5* x *5* matrix). Then, *4* x *4* + *3* x *3* + *2* x *2* + *7* x *7* + *5* x *5* = *103* elements will be recorded. The one-level matrix and multi-layer matrices representation of the gearbox are listed in Figs. 3-5 and 3-6, respectively.

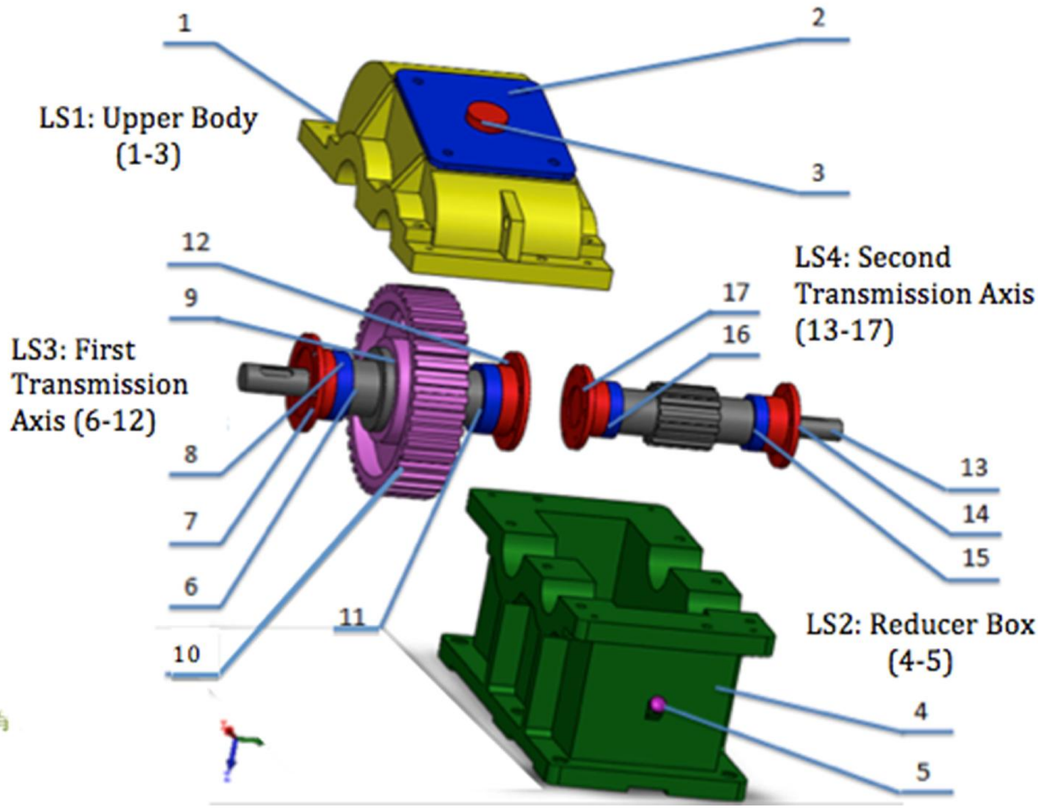


Fig. 3-4 Subassemblies of the Gearbox

50	-1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	15	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-1	0	0	50	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
0	0	0	1	10	0	0	0	0	0	0	0	0	0	0	0	0
-1	0	0	1	0	30	0	1	-1	-1	-1	0	0	0	0	0	0
-1	0	0	1	0	0	20	-1	0	0	0	0	0	0	0	0	0
-1	0	0	1	0	-1	1	15	0	0	0	0	0	0	0	0	0

-1	0	0	1	0	1	0	0	20	0	0	0	0	0	0	0	0
-1	0	0	1	0	1	0	0	1	25	-1	0	0	0	0	0	0
-1	0	0	1	0	1	0	0	0	1	15	-1	0	0	0	0	0
-1	0	0	1	0	0	0	0	0	0	1	20	0	0	0	0	0
-1	0	0	1	0	0	0	0	0	0	0	0	40	0	-1	1	0
-1	0	0	1	0	0	0	0	0	0	0	0	0	20	1	0	0
-1	0	0	1	0	0	0	0	0	0	0	0	1	-1	15	0	0
-1	0	0	1	0	0	0	0	0	0	0	0	-1	0	0	15	1
-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	-1	20

Fig. 3-5 One-Level Constraint Matrix Representation of the Gearbox

CM1 =

50	1	1	1
-1	50	-1	-1
-1	1	80	0
-1	1	0	80

CM 1

Fig. 3-6 (a) Multi-Layer Matrices Representation of the Gearbox – First Level Matrix

50	-1	0
1	15	-1
0	1	10

CM 2.1

50	-1
1	10

CM 2.2

30	0	1	-1	-1	-1	0
0	20	-1	0	0	0	0
-1	1	15	0	0	0	0
1	0	0	20	0	0	0
1	0	0	1	25	-1	0
1	0	0	0	1	15	-1
0	0	0	0	0	1	20

CM 2.3

40	0	-1	1	0
0	20	1	0	0
1	-1	15	0	0
-1	0	0	15	1
0	0	0	-1	20

CM 2.4

Fig. 3-6 (b) Multi-Layer Matrices Representation of the Gearbox – Second Level Matrix

It is obvious that multi-layer matrices require smaller sized memory to store the models. The searching time for the near optimal sequence is also reduced because of the smaller search size. An example will be used to explain the efficiency improved comparing to other existing methods in section 3.4. Besides the reduction of searching time, the disassembly time of a large subassembly is usually less than the total disassembly time of each element in this large subassembly. For example in the gearbox, the disassembly time for LS1 is 50 seconds, while the total disassembly time for the three components of LS1 is 70 seconds. That means that this representation may even lead to a reduced total disassembly time.

3.4 Searching for Possible Sequences Using Traversing Search Algorithm

A product model is represented by multi-layer matrices. The first level matrix represents the model, and it is marked as Matrix 1. The second level matrices represent the first-level subassemblies of the model, and the matrices are marked as Matrix 2.p, where “p” stands for the numbers of the first-level subassemblies. The other level matrices follow the same procedure. If a target part is confirmed, the positions of the target part in each level matrices are firstly identified. Theses matrices will be searched in following processes. The framework of the traversing searching method is shown in Fig. 3-7.

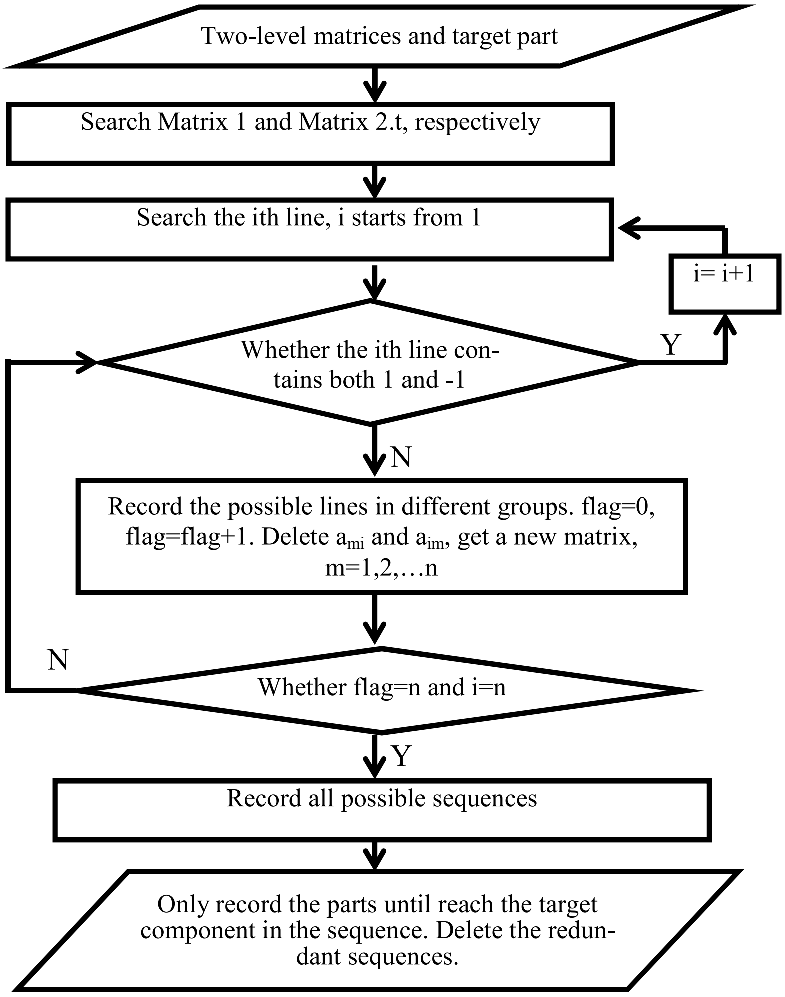


Fig. 3-7 Framework of the Traversing Searching Method

Take Fig. 3-3 as an example. If a row in the matrix has both “1” and “-1”, this subassembly is fixed temporally in current state of the product. Otherwise, this subassembly will be able to move from at least one direction of the LCS. After a subassembly of the product is removed, the corresponding row and column of the matrix are deleted, and the matrix is simplified into a smaller size. The next row will then be searched likewise. Following this process, the matrix can be simplified into a smaller and smaller size. A disassembly sequence of first-level subassemblies can be generated following the search process.

Following the similar procedures, the matrices contain the target part will be searched, while the other matrices will be skipped. This procedure helps to reduce the search size greatly. After the searches of each level matrices are finished, sequences of disassembling each level subassemblies will be combined together. Then all possible disassembly sequences are finally generated. By evaluating all possible sequences, a near optimum sequence can be selected finally.

Taking the gearbox in Fig. 3-4 as an example, part 5 is chosen as the target part. There are $17! = 3.55 \times 10^{14}$ sequences to be searched if using one level matrix representation method to search the matrix in Fig. 3-5. However, using multi-layer representation method, the product model can be represented by two-level constraint matrices. And only matrix 1 and matrix 2.3 are searched. The total search number will be $4! + 7! = 5064$, which reduces the computational complexity greatly.

3.5 Summary

The search size and computational complexity of product models are greatly reduced by using the multi-layer constraint matrices representation method as this method requires small memory of computers to run when dealing with complexity problems.

However, the traversing searching method is not efficient enough. A lot of unrealistic sequences may be searched and generated. In Chapter 4, the ant colony method will be introduced to increase the solution efficiency. A computer program is developed to increase the automatic search level of this method.

4 Chapter 4

Optimized Searching Method for DSP

An efficient searching method for the near optimal disassembly sequence is important to reduce the search time of product models. Two types of commonly used searching methods, which are discussed in the literature review, have their own advantages as well as disadvantages. Since the ant colony optimization searching method is efficient in solving practical problems as discussed in literature review, it is chosen as the searching method for the optimal disassembly sequence in this research.

4.1 Ant Colony Optimization Algorithm

In the natural world, ants travel randomly while looking for food. However, once foods are found, ants will lay down pheromone trail in their way back to colony. Other ants will stop travelling randomly upon finding the path, and there is a very high possibility that they follow this path [36]. The pheromone trail starts evaporating over time, and the attractive strength of the trail reduces simultaneously. Thus, it is very possible that some other paths to be found. Whenever a good path from the colony to the food source is found, other ants are very likely to follow this path. The positive feedbacks eventually lead to all the ants following this path [39].

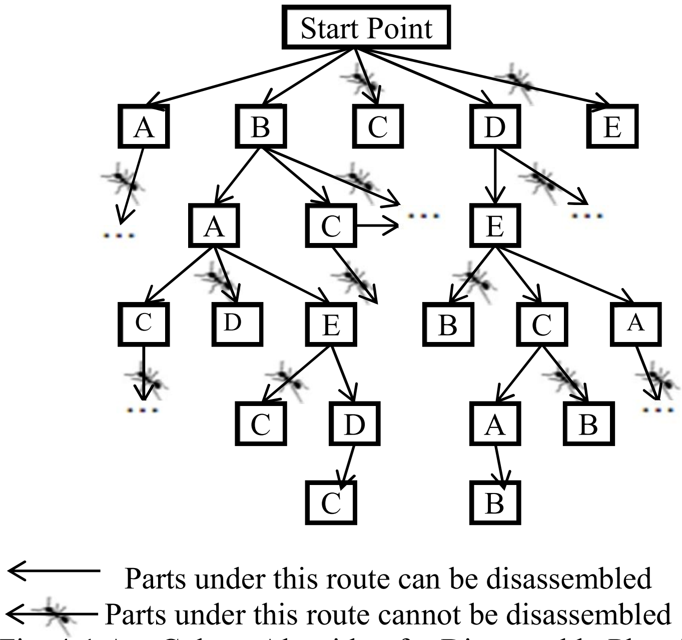


Fig. 4-1 Ant Colony Algorithm for Disassembly Planning [40]

The idea of the ant colony operation algorithm is to mimic ants' behaviour, and the evaporation of pheromone can stop the solution from convergence to a locally optimal solution [41]. Considering the real-life problem of disassembly sequence planning, if an element of a path cannot be disassembled, then this path is an impossible one, and there is no necessary to consider the components afterwards. Fig. 4-1 shows operation principles of the ant colony algorithm [41]. A tour of ants can be used to represent a possible product disassembly solution. If an ant has passed N-1 possible operation nodes of its route, it moves to the next node where the operation possibility is false. That means the route is unreasonable, and the following nodes will not be calculated. By applying this principle for disassembly sequence planning, the computational complexity can be reduced a lot comparing with traversing searching methods. The efficiency of the searching process is also improved. Moreover, if the target part is already disassembled in the previous operation, the searching stops and no other components need to be considered in future operations. Once a possible path is found, criteria will be used to evaluate the quality of the path for a near optimum sequence.

If a product contains "n" elements, there are totally "n!" sequences to be searched using the traversing search algorithm. However, applying the ant colony method in searching the constraint matrices can reduce the searching time and computational complexity in following two aspects for selective disassembly sequence planning:

1). This method can recognize unrealistic sequences to be skipped from the beginning of the search. For example, if there are both "1" and "-1" in the first row of Matrix 1 shown in Fig. 3-4 in searching, the large subassembly (LS1) cannot be the first element to be removed. Therefore, sequences starts with "LS1" are unrealistic, which will not be searched. A total of (n-1)! - 1 searches will be saved, which is approximately 1/n of all searches compared to using the traversing search. For example in Fig. 4-1, components C and E cannot be disassembled as the first components, then all sequences starts with C and E are not possible, and they will not be searched. In the same way, if "LS1" can be removed as the first element, but "LS2" cannot be disassembled as the second element, sequences starts with "LS1-LS2" will be skipped, and (n-2)! - 1 searches will be saved. The same process happens when searching small subassemblies. A lot of unrealistic searches can be avoided.

2). The method keeps detecting the target part whenever a component is disassembled. This helps to recognize the time when the searching is completed, which avoids repetitive searching. For selective disassembly sequence planning, if a large subassembly with the target part is disassembled, the searching process moves to the next level (small subassembly level). Or if the target part itself is disassembled, the searching process stops. For example in Fig. 4-1, taking component A as the target part, after finding the sequence B-A, this sequence will be recorded. Other sequences starting with B-A will not be searched any more, even though component C and component E can still be disassembled. The disassemble possibility of component C as the first element to be removed will be checked subsequently.

Taking the gearbox in Fig. 3-3 as an example, if the small subassembly (SS5) is chosen as the target part, according to the above description, matrix CM 1 and matrix CM 2.3 in Fig. 3-5 should be searched. When searching the matrix 1, there are totally $4! = 24$ searches using the traversing search algorithm. However, only 14 sequences ($24-(3!-1)-(3!-1)-(1!-1)-(1!-1) = 14$) are searched based on the ant colony method. Among the 14 sequences, 10 can finally lead to the disassembly of the target part (possible sequences). When searching matrix 2.3, there are $7! = 5040$ searches if using traversing search algorithm. Only 77 sequences are searched using the ant colony method, and there are 22 possible sequences among the 77 searches. Overall, searching time and computational complexity can be greatly reduced by using this method.

4.2 Criteria for Evaluating Sequences

The searching efficiency is improved by using the proposed method. However, there may be multi-sequences generated when searching the solution for DSP problems, especially when there are lots of parts in a product. Sometimes there might be hundreds of different valid sequences. Even though each of these sequences can finally lead to the removal of the target part, it is essential to decide which one is better (based on different requirements) to reach the target part efficiently. Usually, there are different standards for evaluating sequences. In this research, total disassembly time for a sequence and the total number of components removed are considered to be the evaluation standards. To calculate the total disassembly time of a sequence, criteria for product disassembly complexity are built.

4.2.1 Criteria for Optimal Product Disassembly Sequence

The number of parts removed to reach the target part is the most obvious factor when evaluating possible sequences. However, time spent is accompanied with operations of product disassembly. Therefore, time is also one of the most important and effective factors in the evaluation. In this research, sequences with the least disassembly time required and the minimum number of parts to be removed is selected as the optimum one.

The time required to disassemble a part is different from one to another depending on the part location and complexity. Some parts take longer time than others for processing. The longer moving time may result from the difficulty of a disassembly operation. Data related to the product disassembly complexity in geometry, additional time required, and processes are collected from literature as listed in Tab. 4-1 [42].

Tab. 4-1 Product Disassembly Complexity [42]

Criterion	No.	Complex level	Additional time required
Geometry complexity	1	Part size: (1) Regular size (2) Small size (3) Large/heavy (4) Super large/heavy subassembly (need additional assist)	 0s 23s 25s 86s
	2	Handling difficulty: (1) Tool or fixture required (2) Difficulty	 20s 18s
	3	Features may cause jam and tangle: (1) Yes (2) No	 12s 0s
Process complexity	4	Fastening type: (1) Multi-pieces (nuts and bolts) (2) Screws and nails (3) Rivets, staples and adhesive	 24s 24s 20s
	5	Enough space for disassembly operations (1) Yes (2) No	 0s 20s
	6	Interference with other parts (1) Yes (2) No	 22s 0s

4.2.2 Evaluation Method of Disassembly Sequences

Based on the above criteria of the optimal or near optimal product disassembly planning, the total disassembly time to remove the target part can be calculated for possible sequences to select the best one which needs the least time.

Firstly, the total disassembly time of a component is formulated as following:

$$T_{Pi} = T_0 + \sum_{j=1}^{6} T_{cj} \qquad (1)$$

Where T_0 stands for the basic time required for disassembling a regular part. A regular part in this research is defined as a part with regular size (larger than 64 cm^3 and smaller than $6.4x10^4$ cm^3), regular weight (less than 20kg) and normal shape, which will not bring difficulty in a disassembly operation. In this research, T_0=15s. While j is the criteria number listed in Tab. 4-1, and i is the number of the removed part. T_{cj} represents the additional time required for different criteria, which is also listed in Table 1. If part i satisfy the criteria j, then the corresponding additional time T_{cj} will be added to the total disassembly time of this component i. Otherwise, the additional time will not be calculated. According to the formula 1, time required for removing different parts could all be calculated.

If a subassembly contains several components, the total removing time of these components can be calculated by adding all removing time spent for different components together.

$$T_{sub} = \sum T_{Pi} = \sum (T_0 + \sum_{j=1}^{6} T_{cj}) \qquad (2)$$

Results obtained from the formula (2) are then put into the principal diagonal positions of constraint matrices. They are used for the calculation of total disassembly time of different sequences.

If a disassembly sequence given, the total operation time can be calculated by adding all removing time spent for different subassemblies together. Formula (3) is used to calculate the final result of a sequence.

$$T_{all} = \sum T_{sub} = \sum\sum(T_0 + \sum_{j=1}^{6} T_{cj}) \qquad (3)$$

The total number of components disassembled for a sequence is N_{com}. Thus, the objective function of this research is as follows.

$$min\ f(x) = [T_{all}, N_{com}] \qquad (4)$$

When both the least disassembly time and removing the minimum number of components cannot be met, the total disassembly time of a sequence is considered as more important than the total number of components disassembled, and a near optimum sequence will be applied.

4.3 Implementation of Ant Colony Algorithm

The basic principles of the ant colony method and its advantages have been introduced in the previous parts. The implementation of this method in this research will be explained below. Fig. 4-2 is the framework of the ant colony operation algorithm.

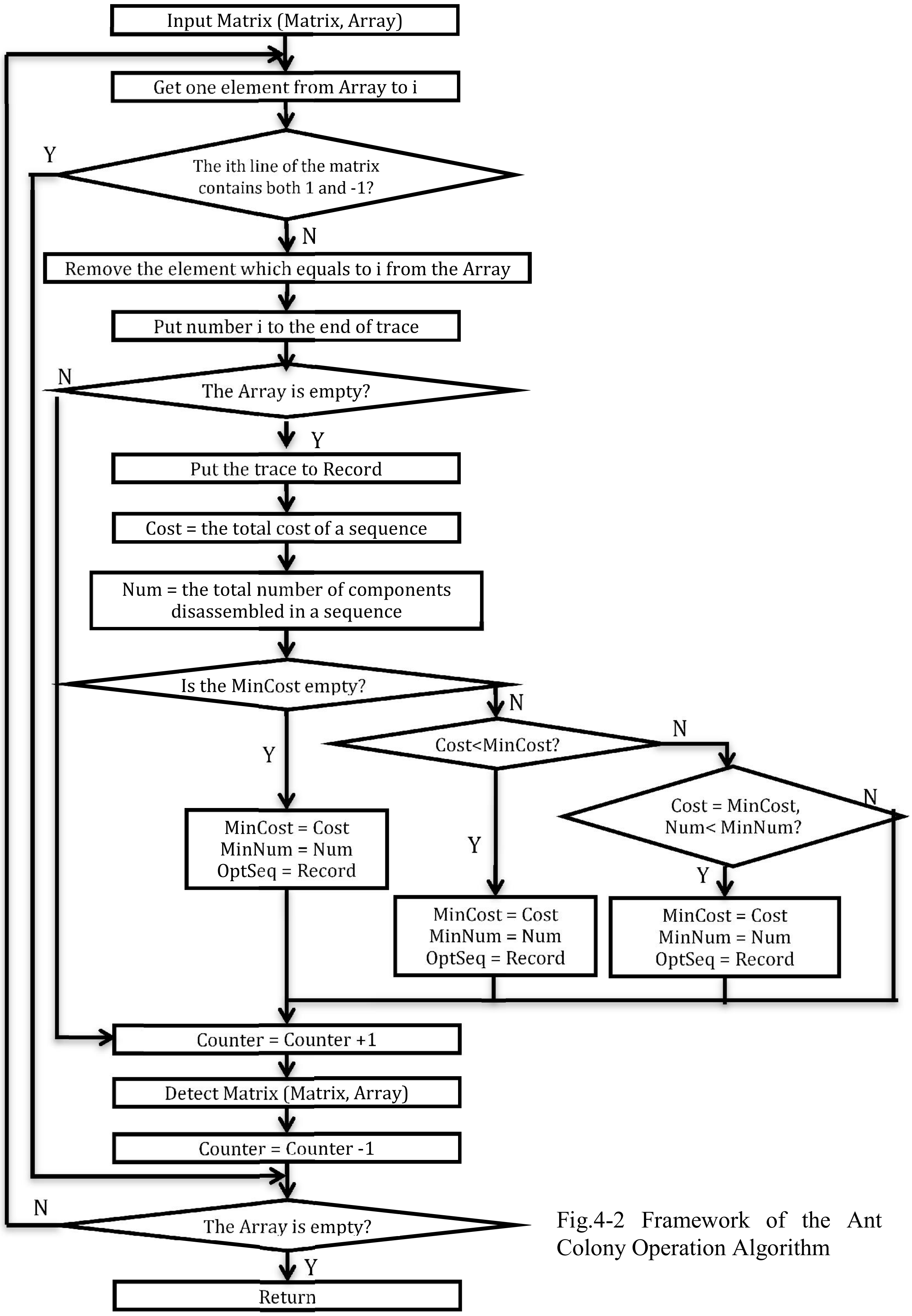


Fig.4-2 Framework of the Ant Colony Operation Algorithm

Once a matrix is generated, the disassembly ability of each component is checked. If the component cannot be disassembled, sequences start with this component will not be searched any more, and other components will be checked continuously. Otherwise, the component disassembled will be recorded and compared to the target part. If the component is not the target part, the matrix will be simplified and the next level searching starts. If the component is the target part, the disassembly sequence will be recorded and its cost will be calculated. This disassembly cost is used to compare to the minimum cost. If the cost is smaller than the minimum cost, the disassembly sequence and its corresponding cost will be used to replace the optimized sequence and minimum cost. If the disassembly cost equals to the minimum cost, the number of components removed will be considered. Sequence and cost with fewer components movements will be saved as the optimized sequence and minimum cost. Otherwise, if the disassembly cost is larger than the minimum cost, the optimized sequence and minimum cost keep the same, and the system continues searching other sequences. Following these processes, an optimized sequence with the least disassembly cost and fewer movements of components will be finally generated.

4.4 Program Coding

The proposed optimized searching method has improved the planning efficiency. However, when dealing problems with a large number of product components, there might be hundreds of possible sequences searched. Based on the above method, a program coding is developed for searching the near optimum sequence automatically.

The program coding is written in Matlab. Fig. 4-3 shows the user interface of the search program. Fig. 4-4 explains the working processes of this program.

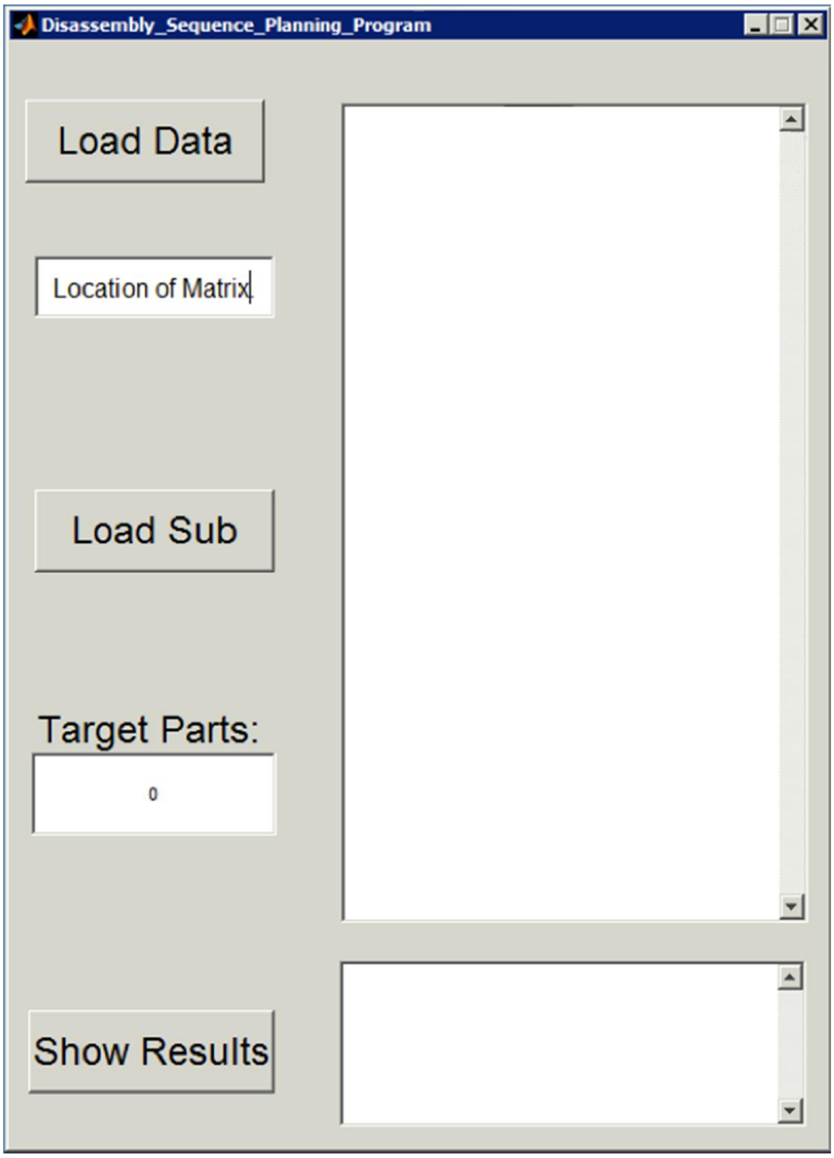


Fig. 4-3 Interface of the Developed Program

Inputs of this search program are the constraint matrices and the part number of the target part. Each of the matrices is saved in an Excel file separately. Firstly, users click on the

“Load Data” button, and a window will show up to let users choose/input the location of Matrix 1. Matrix 1 records relationships between first-level subassemblies. The search program can also extract the number of first-level subassemblies. Based on the number of the first-level subassemblies, the program sets the number of second-level matrices for input. It is not necessary for users to click on “Load Sub” button each time, the system can record/identify the operation automatically.

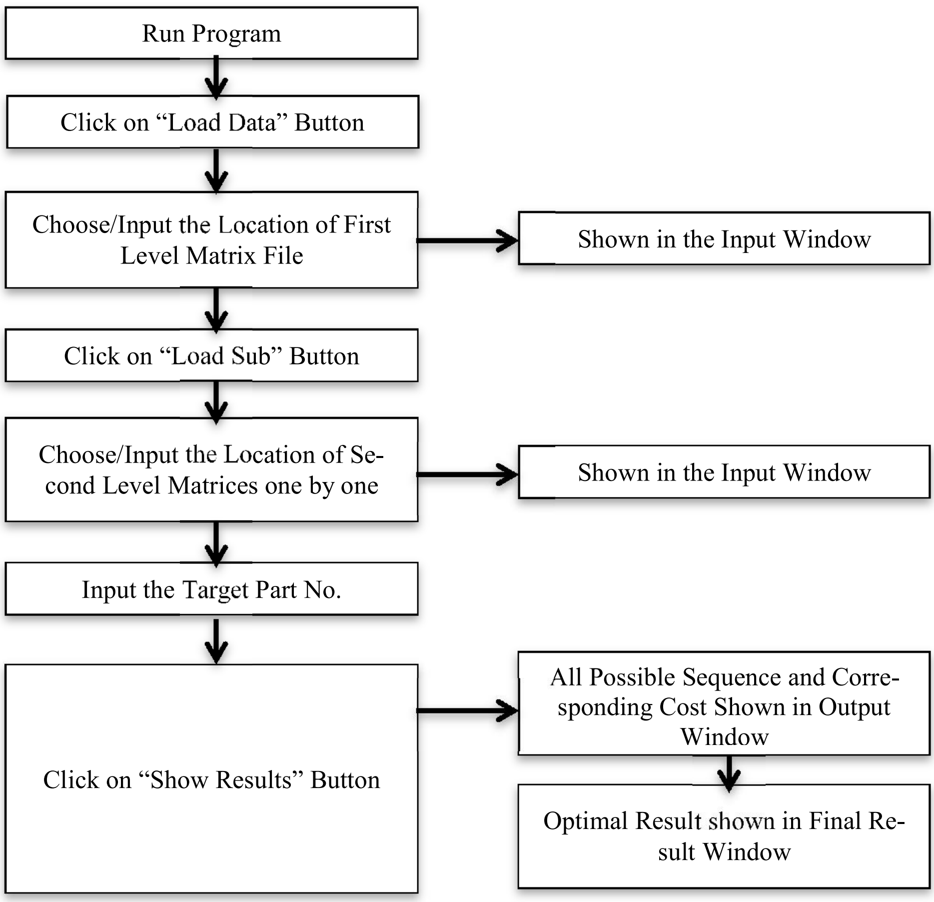


Fig. 4-4 Flowchart for Using the Program

As long as “Load Sub” button is clicked, a window will show up for users to choose the location of Matrix 2.1, and then Matrix 2.2, Matrix 2.3 … automatically. All input matrices, both the first-level matrix and second-level matrices, are shown in the right upper blank window. Once all second-level matrices are loaded, the number of the target part is required in the blank window as “Target Part”. At this point, all the inputs required are loaded in the program. After clicking on the “Show Results” button, the program will start to search and compare sequences automatically. At last, a near optimum sequence with the minimum time in the operation and fewer movements of all parts will be selected and recorded in the lower blank window of the user interface. This sequence is considered as the near optimum sequence, and it will be used for the simulation to verify the solution later.

4.5 Example

Fig. 4-5 shows an example used to explain the working procedures of the ant colony method and its efficiency. The model is a transmission system of a car. It has two forward gears and one reverse gear. There is a spline on the first transmission axis and another one on the second transmission axis. Fig. 4-6 (a) shows the structure of the first transmission axis, and Fig. 4-6 (b) shows the structure of the second transmission axis. When the spline 1 works on the green and blue gears while the spline 2 works on no gears, the transmission system works as a high-speed forward gear, shown in Fig. 4-7 (a). However, when the spline 1 works on the red and blue gears while the spline 2 works on no gears, the transmission system works as a low-speed forward gear, shown in Fig. 4-7 (b). If the spline 1 works only on the blue gear, while spline 2 works on the green and blue gear, the transmission system works as a backward gear, shown in Fig. 4-7 (c).

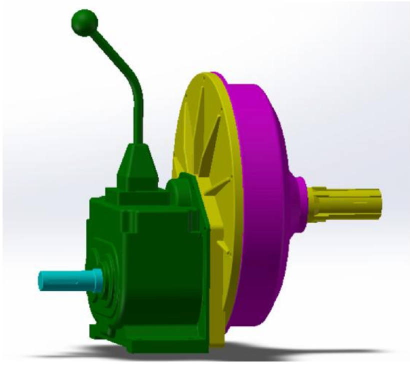
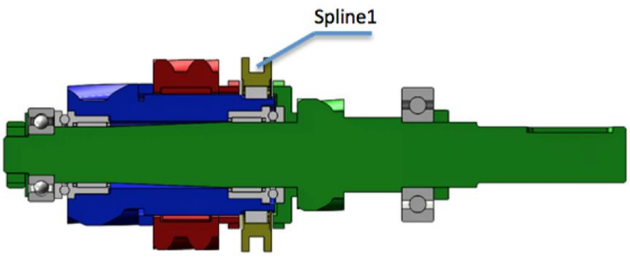
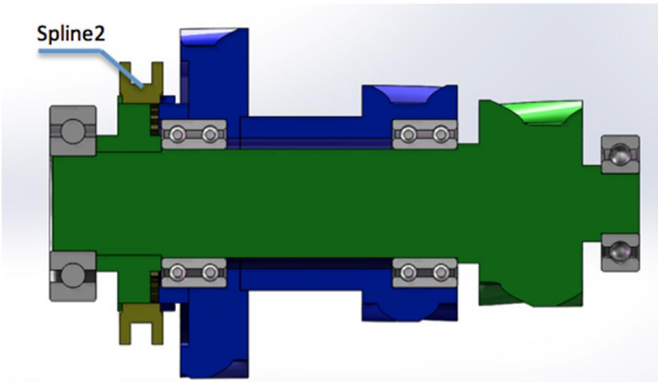


Fig. 4-5 Transmission System of a Car



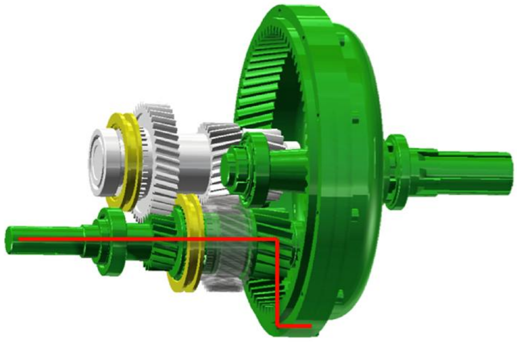
(a)



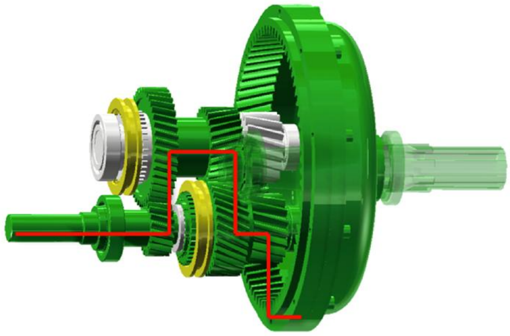
(b)

Fig. 4-6 (a) Structure of the First Transmission Axis

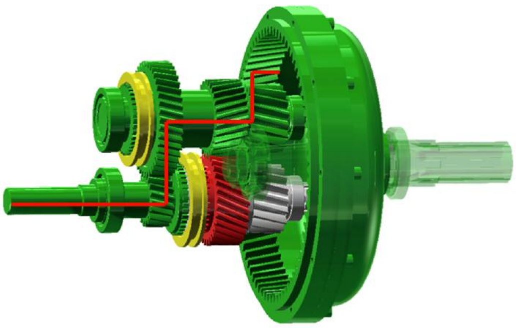
(b) Structure of the Second Transmission Axis



(a)



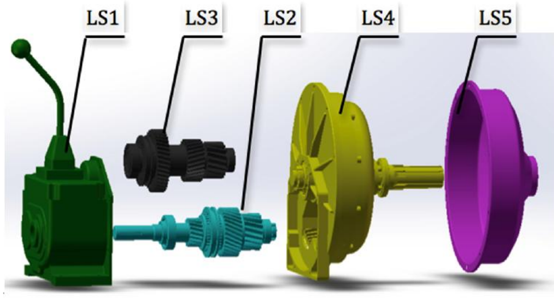
(b)



(c)

Fig. 4-7 (a) High-speed Forward Gear
(b) Low-speed Forward Gear
(c) Backward Gear

The transmission system contains 31 elements. Using one-level constraint matrix representation method, it can be represented by a 31x31 matrix. However, based on the function of each element, the transmission system can be separated into five first-level subassemblies: body, three transmission axes, and a cover. The relationships of the first-level subassemblies are represented by matrix 1. The relationships of the first-level subassemblies and its constraint matrix are shown in Fig. 4-8. Each first-level subassembly contains several parts, and a constraint matrix is used to represent each first-level subassembly. Fig. 4-9 shows each first-level subassembly and its corresponding constraint matrix.

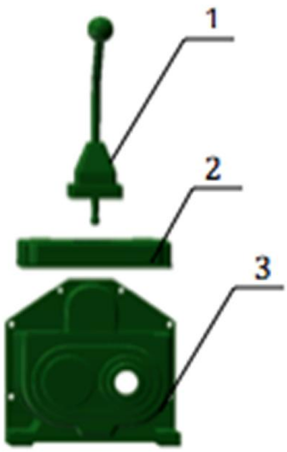


202	1	1	1	0
-1	130	0	1	0
-1	0	130	1	0
-1	-1	-1	123	1
0	0	0	-1	104

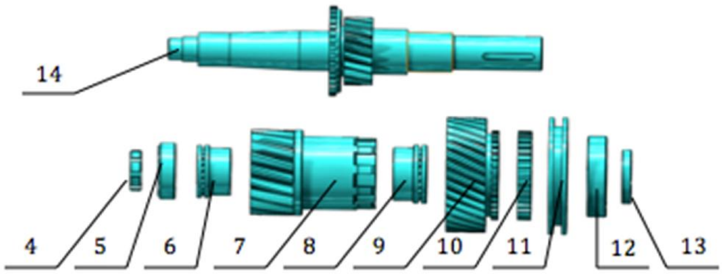
(a) (b)

Fig. 4-8 (a) Subassemblies of the Transmission System

(b) First-level Constraint Matrix of the Product

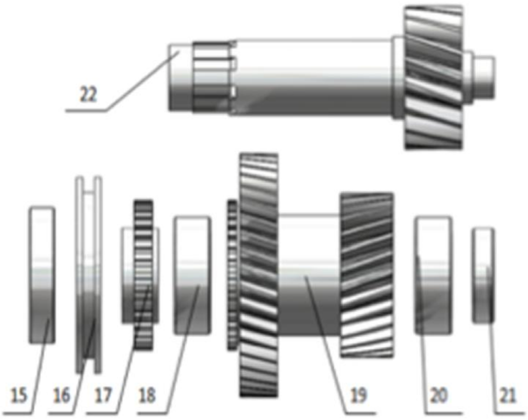


84	1	0
-1	84	1
0	1	86



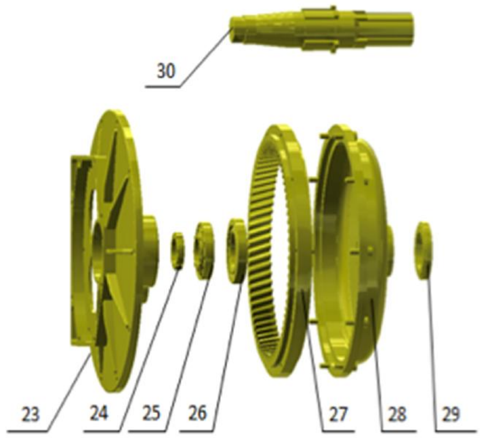
64	1	0	0	0	0	0	0	0	0	1
-1	64	1	0	0	0	0	0	0	0	1
0	-1	64	1	0	0	0	0	0	0	1
0	0	-1	94	1	1	1	0	0	0	0
0	0	0	-1	64	0	0	0	0	0	1
0	0	0	-1	0	69	1	1	0	0	0
0	0	0	-1	0	-1	84	0	0	0	1
0	0	0	0	0	-1	0	64	0	0	0
0	0	0	0	0	0	0	0	64	1	-1
0	0	0	0	0	0	0	0	-1	64	0
-1	-1	-1	0	-1	0	-1	0	1	0	123

(a) (b)



44	0	0	0	0	0	0	1
0	40	0	0	1	0	0	0
0	0	44	1	1	0	0	1
0	0	-1	40	1	0	0	0
0	-1	-1	-1	84	1	0	0
0	0	0	0	-1	44	0	1
0	0	0	0	0	0	40	-1
-1	0	-1	0	0	-1	1	69

(c)



69	0	0	0	1	0	0	0
0	58	1	0	0	0	0	1
0	-1	58	0	0	0	0	1
0	0	0	58	0	1	0	1
-1	0	0	0	84	1	0	0
0	0	0	-1	-1	84	0	1
0	0	0	0	0	0	44	-1
0	-1	-1	-1	0	-1	1	69

(d)

84

(e)

Fig. 4-9 (a) Body of the Transmission System
(b) Input Transmission Axis
(c) Second Transmission Axis
(d) Output Transmission Axis
(e) Cover of the Transmission System

Taking element 17 as the target part, Table 4-2 compares one-level matrix representation traversing searching method, two-level matrix representation traversing searching method, and two-level matrix representation ant colony searching method over the search size, number of searches, number of sequences searched, and number of possible sequences. It is obvious that the two-level constraint matrix representation method reduces the search size greatly, and the ant colony searching method help avoid a lot of unnecessary searching.

Table 4-2 Comparisons of Different Methods

Target Part: 17	Search size	Number of searches	Sequences searched	Possible sequences
One-level matrix representation Traversing searching Algorithm	961	$8.2\text{x}10^{33}$	N/A	N/A
Two-level matrix representation Traversing searching Algorithm	284	$4.0\text{x}10^{4}$	$4.8\text{x}10^{6}$	8918(26x343)
Two-level matrix representation Ant colony searching Algorithm	284	630(26+604)	15704	8918(26x343)

4.6 Summary

The multi-layer matrices representation has reduced the search size greatly. The removing of unnecessary data avoids unnecessary searches. The ant colony operation algorithm can further reduce the search size, which helps to reduce the searching complexity. The combination of the multi-layer constraint matrices representation method and ant colony searching method improved the efficiency of disassembly sequence planning, especially for the product with many parts. The evaluation criteria guaranteed a near optimal sequence selected, the program developed helps to improve the automatic level.

5 Chapter 5

Simulation of Disassembly Process

Simulation is the imitation of an operation in a real-world process or system over time [43]. One of the functions of simulation is to show details of a design or process using the computer generated model. Simulation is widely used for education and training purposes [43]. In this research, simulation is used to show the product disassembly process to verify the suggested solution for an easy way to learn the disassembly operations of a product. Eon Studio is an interactive 3D software tool based on the Virtual Reality technology [44]. It provides visualized solutions to improve communication and knowledge transfer using simulation. Eon Studio is used for the simulation of the generated disassembly plan in this research.

5.1 Building of 3D Models

Simulation is generated based on the product model built in the computer. In Eon Studio, 18 data formats are supported including 3DS, DWG, STL, SLDASM and so on. However in the application, it is found that these formats are sometimes difficult to be converted into suitable formats for the model importing. For example, when importing SLDASM (data format of Solidworks) format into Eon studio, product models sometimes cannot be presented properly in color, shape, size, and location. The STL format is found to have a good performance in most cases, and it is therefore selected as the data format to import product model into Eon.

STL is a data format native to the stereolithography CAD software, it is also supported by many other software packages. Considering the convenient level when building a product model, Solidworks is chosen as the 3D model building software instead of stereolithography CAD. Of course, the STL format can be transformed from Solidworks format directly and easily. Fig. 5-1 shows the process to generate a suitable format of the product model.

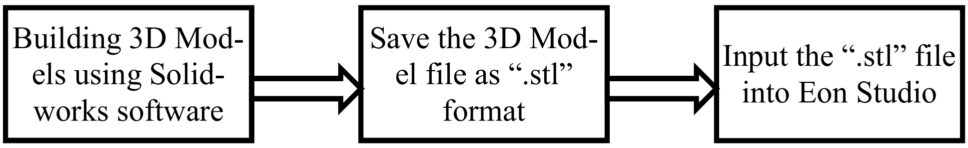


Fig. 5-1 Framework of Generating Suitable Model Format

SolidWorks is a Parasolid-based solid modeller, which uses a parametric feature-based approach to create models and assemblies [45]. However, STL files describe only a raw unstructured triangulated surface by the unit normal and vertices (ordered by the right-hand rule) of the triangles using a 3D Cartesian coordinate system. When transferring SLDASM models into STL models, common CAD model attributes, such as color and texture, are ignored. Even though these attributes cannot be imported into Eon Studio, they can be created in Eon Studio if necessary.

5.2 GCS and LCS

In order to simulate the product disassembly process, directions of the removing components should be indicated. The coordinate systems allow problems in geometry to be translated into problems of numbers, such as matrices or vectors, to make the process easy to deal with. In this research, a global coordinate system (GCS) as well as a local coordinate system (LCS) is used as shown in Fig. 5-2 [46].

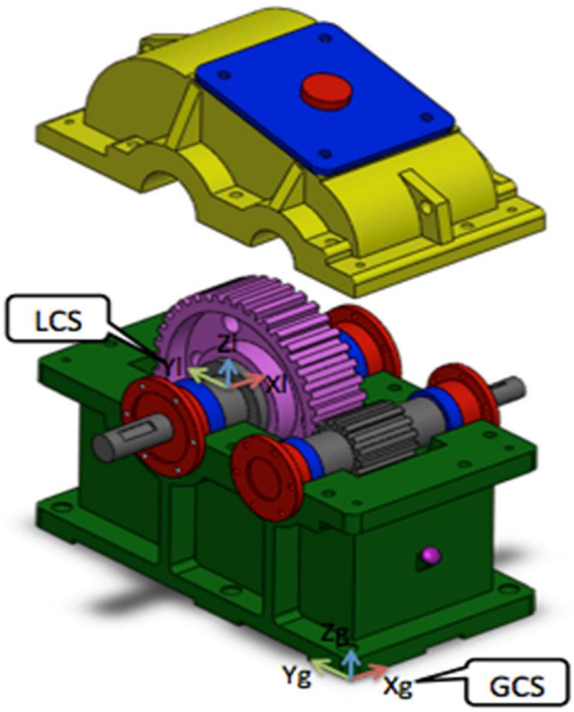


Fig. 5-2 GCS and LCS of the Gearbox

There is only one GCS in one product model, and the origin position of the GCS locates in the base of a product model, while each component in the disassembly model has its own LCS. The relationships between the GCS and the LCSs were set up when building the model using Solidworks, and they were transferred into Eon Studio with the importation of the model. Therefore, it is not necessary to consider the relationships between GCS and LCS in this research. A vector is used to represent the moving direction and distance of a component, and the vector is generated under its LCS. These data are used when simulating of the disassembly process. For example in Fig. 5-2, if the moving direction of the indicated gear is Xl (positive direction) and the moving distance is 10, then (10, 0, 0) will be used to represent the disassembly condition of the gear. The time cost for disassembling the gear is also indicated when doing simulation using Eon studio. Fig. 5-3 shows the information input into the Eon studio in order to do motion presentation. In this case, the removal time of the gear is 25 seconds, and the icons "RelX" "RelY" and "RelZ" are checked because the vector is under the LCS.

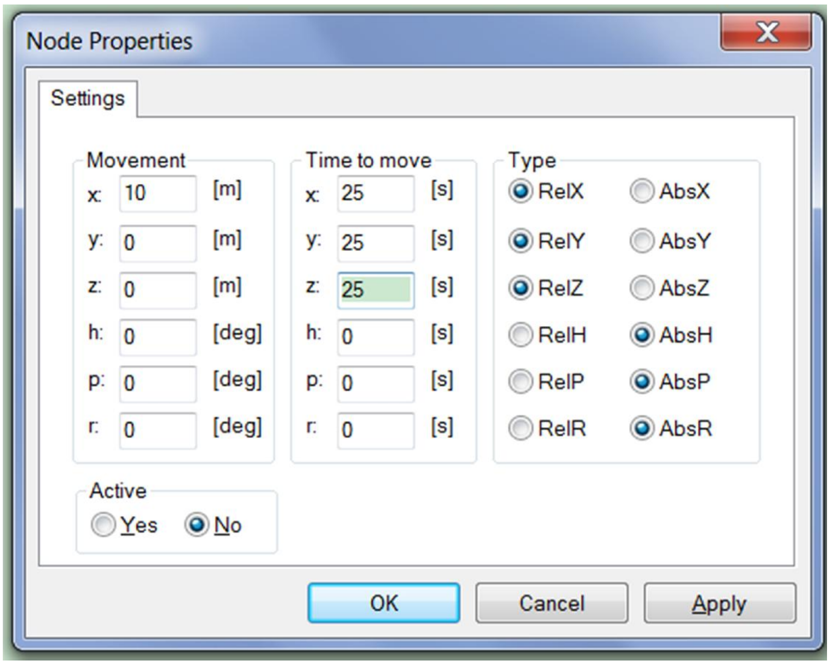


Fig. 5-3 Removing Information of the Indicated Gear

5.3 Simulation operations

After the product model is imported into Eon Studio, parts are listed in the simulation tree automatically. Nodes of the parts can be renamed, and attributes can be added to these nodes. Fig. 5-4 is the simulation tree of the gearbox shown in Fig. 5-2. When running the program, the product model will be shown on the computer screen with attributes added. Eon Studio provides lots of programming nodes with functions such as motion models, sensors, operations, collision, as well as prototypes with the encapsulated function. The visual interface of Eon is supported by the routes function, which allows developers to build a simulation quickly. However, sometimes these nodes and prototypes are not enough for real needs, it is always possible for users to create their own nodes through programming functions.

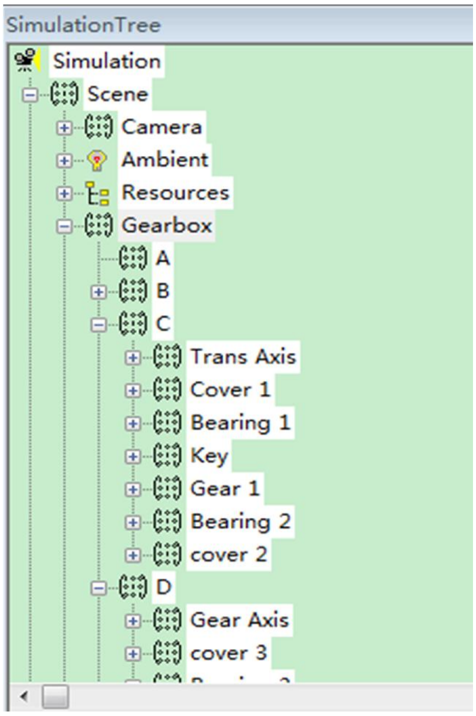


Fig. 5-4 Simulation Tree of the Gearbox

Fig. 5-5 shows the simulation interface, and it is used to simulate the disassemble procedures. Whenever a sequence is generated, users can click on the number buttons to choose the part for disassembly. This helps users to learn the disassemble operations. It can also be used to compare different disassemble processes visually to prove the near optimum solution.

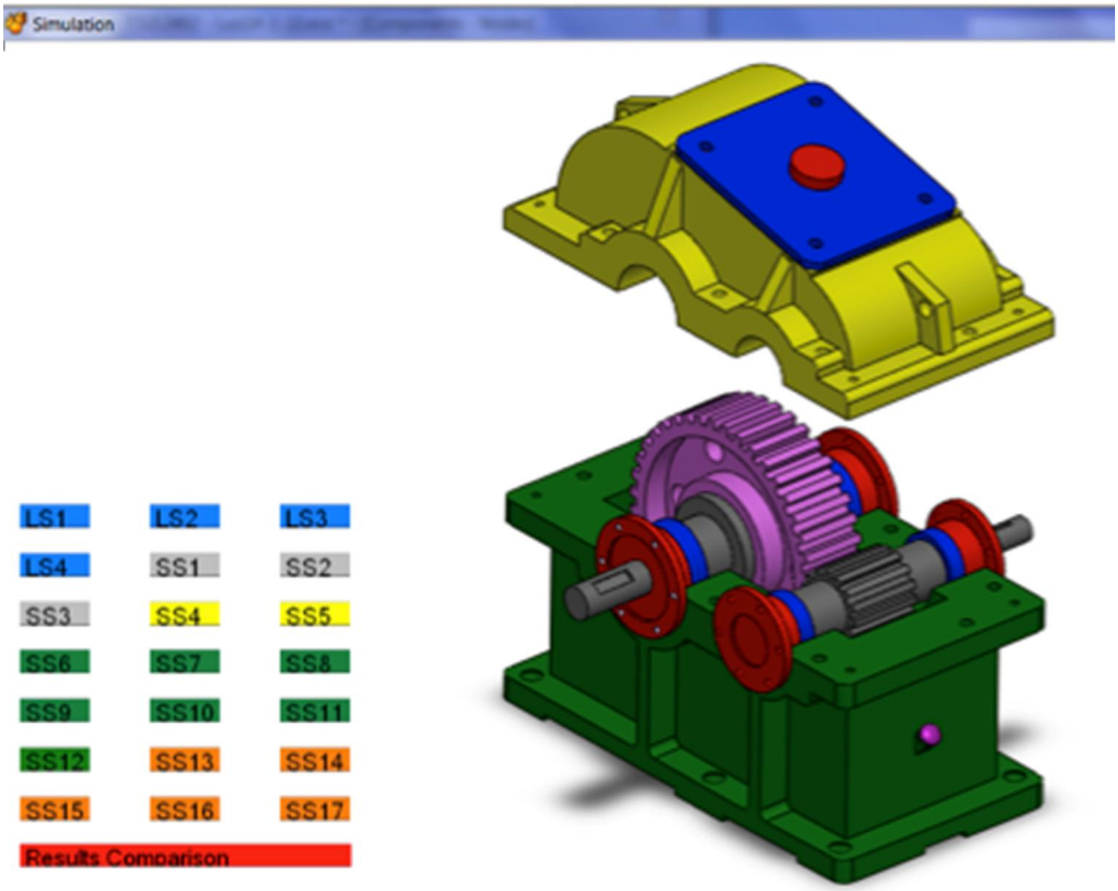


Fig. 5-5 Interface of the Simulation

5.4 Summary

Eon studio is used to simulate the disassemble operations in this research, which provides a visual evaluation for the disassembly planning. The simulation can also helps to verify the near optimum sequence.

6 Chapter 6

Case Study

6.1 Case one

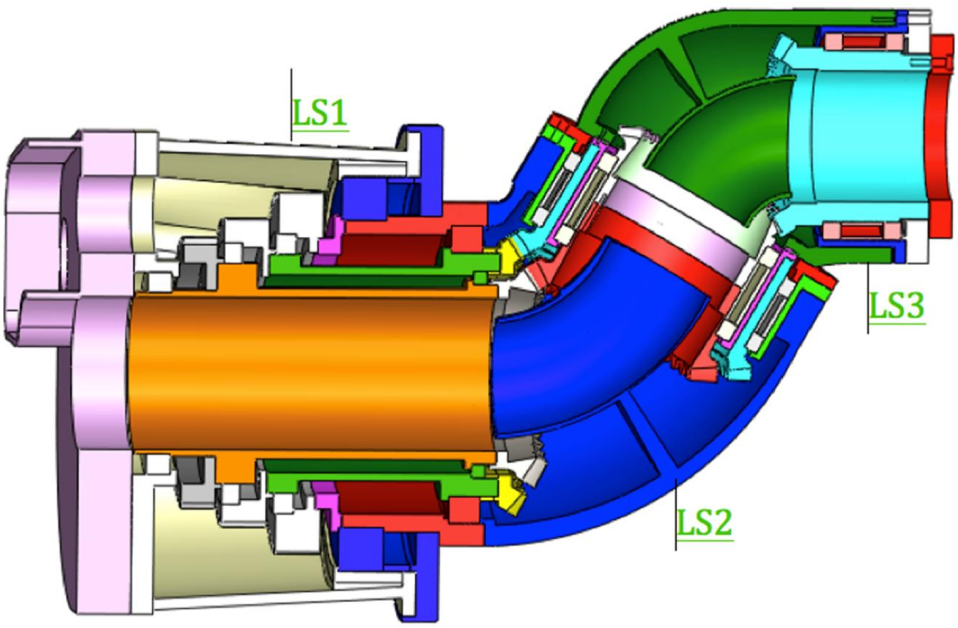


Fig. 6-1 Wrist of a Painting Robot

A product model as shown in Fig. 6-1 is used for the case study to explain and verify the proposed method. The model is an oblique axis non-spherical hollow wrist of a spraying painting robot. The wrist is designed with a compact structure, and it consists of three wrist joints: LS1, LS2, LS3 (shown in Fig. 6-1).

With the flexibility, the wrist can realize three continuous rotational motions to meet the requirements of working environments in spraying industry. Fig. 6-2 is the kinematic diagram of the wrist that shows how the three wrists are driven to rotate about their axes.

From Fig. 6-2, gear “Z4” is driven by gear “Z1”, and gear “Z4” drives “LS1”, “LS2” and “LS3” to rotate about the axis “a” together. Gear “Z2” drives gear “Z5”, while gear “Z5’” drives gear “Z8” to rotate about the axis “b”. “LS2” and “LS3” rotate with gear “Z8” together. Gear “Z6” is driven by gear “Z3”, and gear “Z7” is driven by gear “Z6’”. Gear “Z9” is driven by gear “Z7’”, and “LS3” rotates about the axis “c” with gear “Z9”.

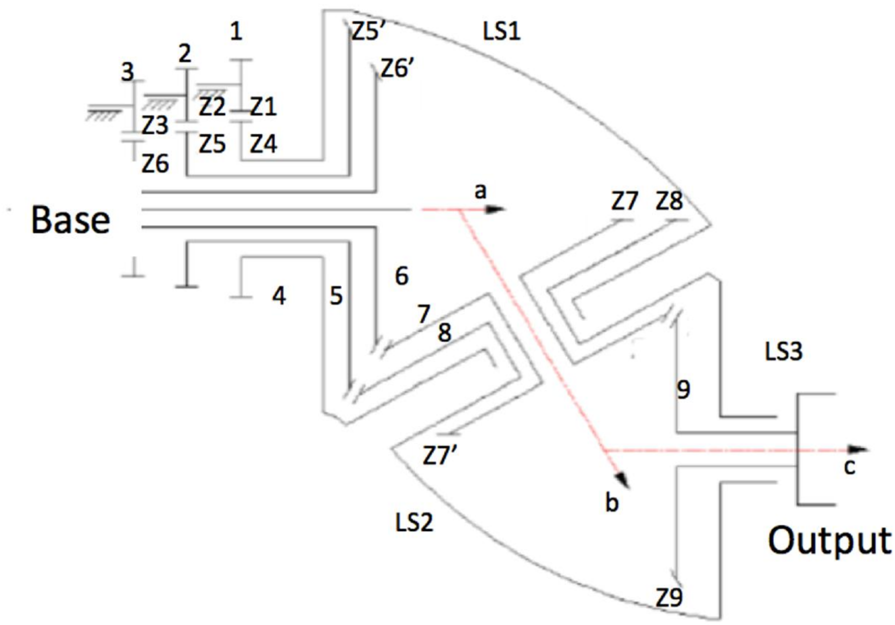


Fig. 6-2 Kinematic Diagram of the Wrist

In this research, the three joints are regarded as three subassemblies, while the three sub-assemblies contain 27 parts in total. Fig. 6-3 shows the structure and components of the product model, and each of the subassemblies and parts is indicated with a number. Table 6-1 records the detailed information of the product components.

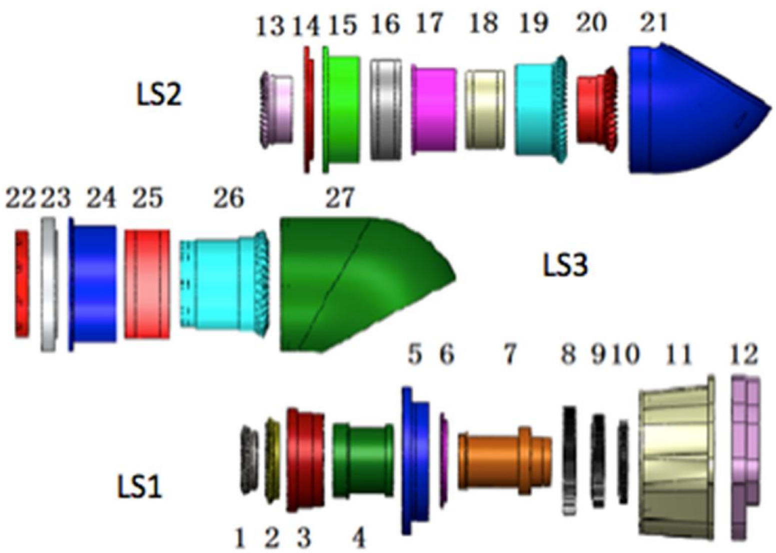


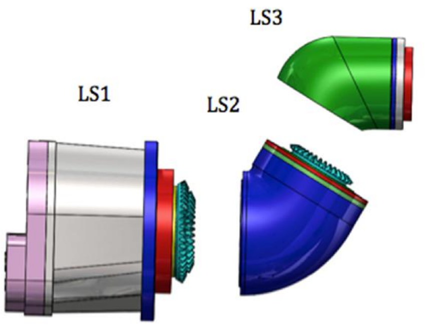
Fig. 6-3 Components of the Product Model

Table 6-1 Details of the Product Model

Part No.	Part Name	Moving Time T_p
LS1	First Joint	202
1	Bevel Gear 1-1	104
2	Bevel Gear 1-2	84
3	Housing 1-1	69
4	Housing 1-2	123
5	Cover 1-1	69
6	Housing 1-3	94
7	Housing 1-4	91
8	Spur Gear 1-1	86

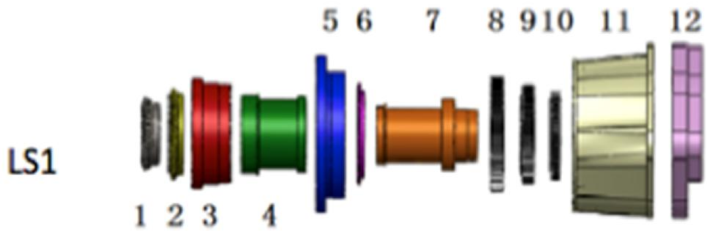
9	Spur Gear 1-2	64
10	Spur Gear 1-3	64
11	Shell 1-1	130
12	Base 1-1	69
LS2	Second wrist	202
13	Bevel Gear 2-1	56
14	Cover 2-1	44
15	Housing 2-1	66
16	Bearing 2-1	58
17	Housing 2-2	40
18	Bearing 2-2	80
19	Bevel Gear 2-2	119
20	Bevel Gear 2-3	94
21	Shell 2-1	130
LS3	Third wrist	180
22	Flange 3-1	44
23	End Cover 3-1	44
24	Housing 3-1	66
25	Bearing 3-1	64
26	Bevel Gear 3-1	91
27	Shell 3-1	130

Using the proposed representation method, the product model is represented by two-level constraint matrices, which are shown in Fig. 6-4 (A) (B) (C) (D). The matrices are input into the sequences searching program. Taking component 17 as a target part, a near optimal result is to be searched. Fig. 6-5 shows the search processing and the near optimum result. Base on the search result, sequence ”LS3-LS2-SS13-SS18-SS17” is chosen as the final result. Totally two subassemblies and three parts are disassembled in order to remove the target part 17, and the total disassembly time for this sequence is 558 seconds. This sequence will be verified using simulation later.



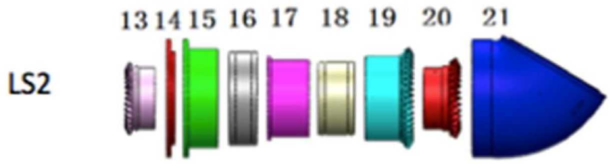
$$\begin{vmatrix} 202 & -1 & 0 \\ 1 & 202 & -1 \\ 0 & 1 & 180 \end{vmatrix}$$

Fig. 6-4 (a) The Product Model and Matrix 1



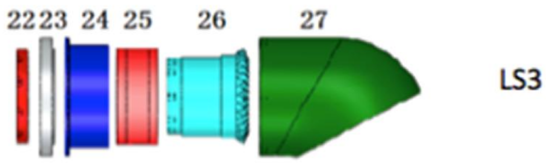
$$\begin{vmatrix}
104 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 84 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 69 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -1 & -1 & 123 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 69 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & -1 & 0 & -1 & 94 & 0 & 1 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & -1 & 0 & 0 & 91 & 0 & -1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 86 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 1 & -1 & 64 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 64 & 1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & -1 & 130 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 69
\end{vmatrix}$$

Fig. 6-4 (b) The First Joint and Matrix 2.1



56	0	0	0	0	1	0	1	0
0	44	1	1	0	0	0	0	0
0	-1	66	-1	0	0	1	0	1
0	-1	1	58	0	0	1	0	0
0	0	0	0	40	-1	1	1	0
-1	0	0	0	1	80	0	1	0
0	0	-1	-1	-1	0	119	1	1
-1	0	0	0	-1	-1	-1	94	1
0	0	-1	0	0	0	-1	-1	130

Fig. 6-4 (c) The Second Joint and Matrix 2.2



44	1	0	0	1	0
-1	44	1	1	0	0
0	-1	66	-1	1	1
0	-1	1	64	1	0
-1	0	-1	-1	91	1
0	0	-1	0	-1	130

Fig. 6-4 (d) The Third Joint and Matrix 2.3

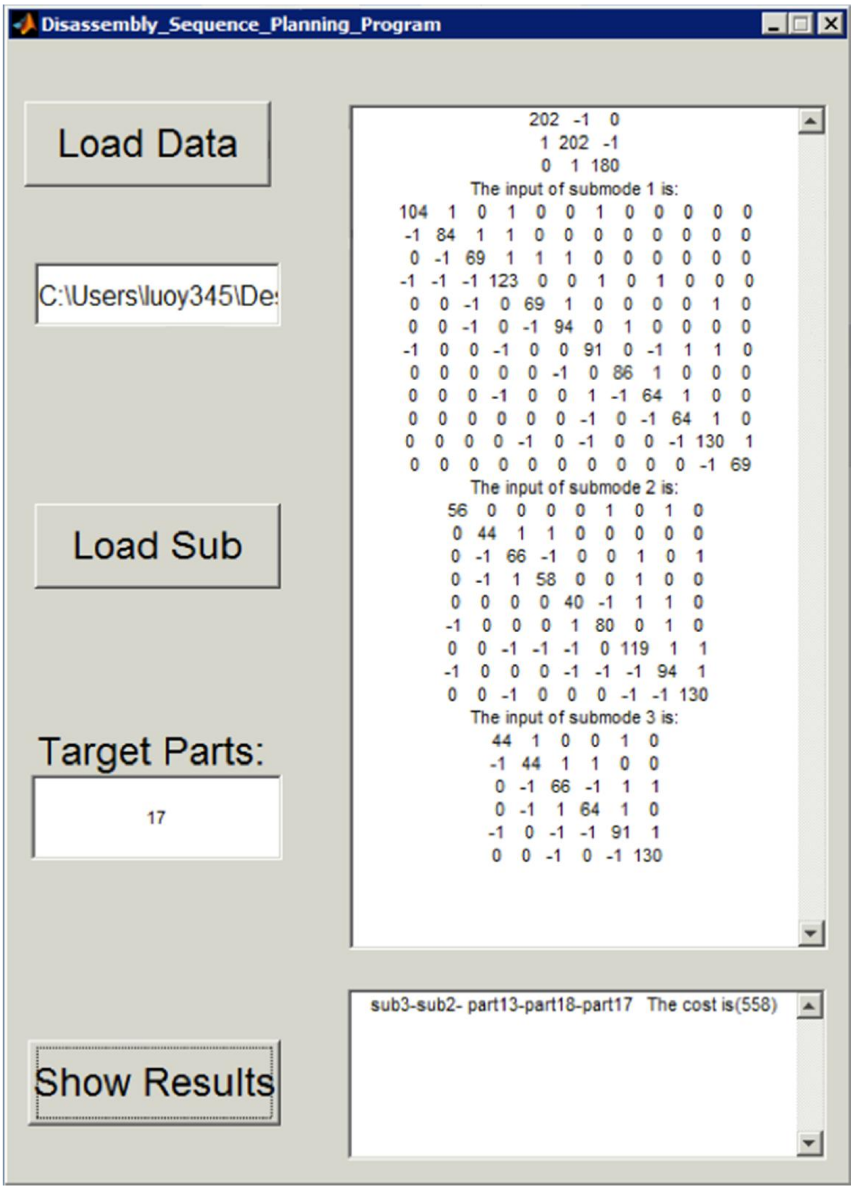


Fig. 6-5 Searching Case 1 for near Optimum Sequence

The product model is built using SolidWorks, and it is saved as ".stl" format before imputing into the simulation software Eon Studio. The moving direction is also defined and input into the simulation as discussed above. Fig. 6-6 shows the simulation tree and route window of the program. After a near optimum disassembly sequence is found, clicking on each component number shown on the operation window in order, the corresponding components will be disassembled.

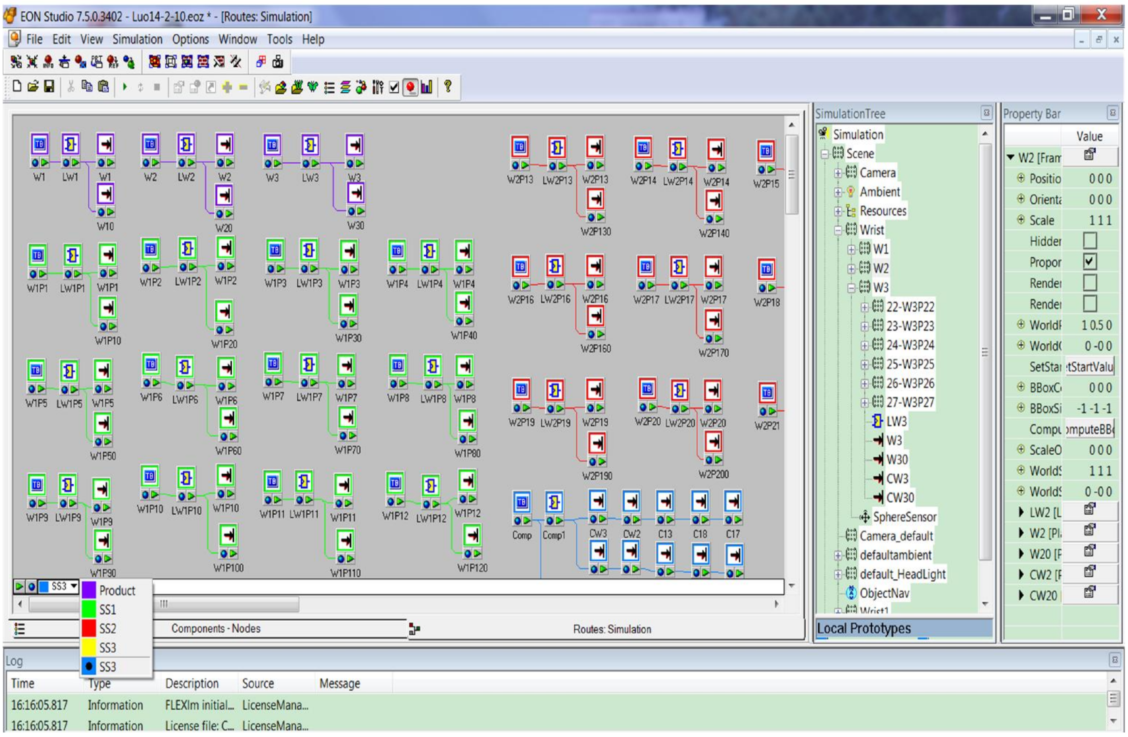


Fig. 6-6 Simulation Tree and Route Window of the Program

The simulation is also used to compare two different disassembly sequences, which can prove the efficiency of the near optimum sequence. Fig. 6-7 shows the comparison of two different disassembly sequences. Both sequences aim at disassembling the component SS17. The upper disassembly process follows the near optimum sequence “LS3-LS2-SS13-SS18-SS17”, and five components are disassembled with the total disassembly time 558 seconds. While the lower process follows the sequence “LS1-LS2-SS14-SS16-SS15-SS13-SS18-SS17” with eight components disassembled and 748 seconds cost for the disassembly process. It can also be observed easily from the simulation that the lower process takes more time than the near optimum process, which proves the efficiency of the near optimum sequence.

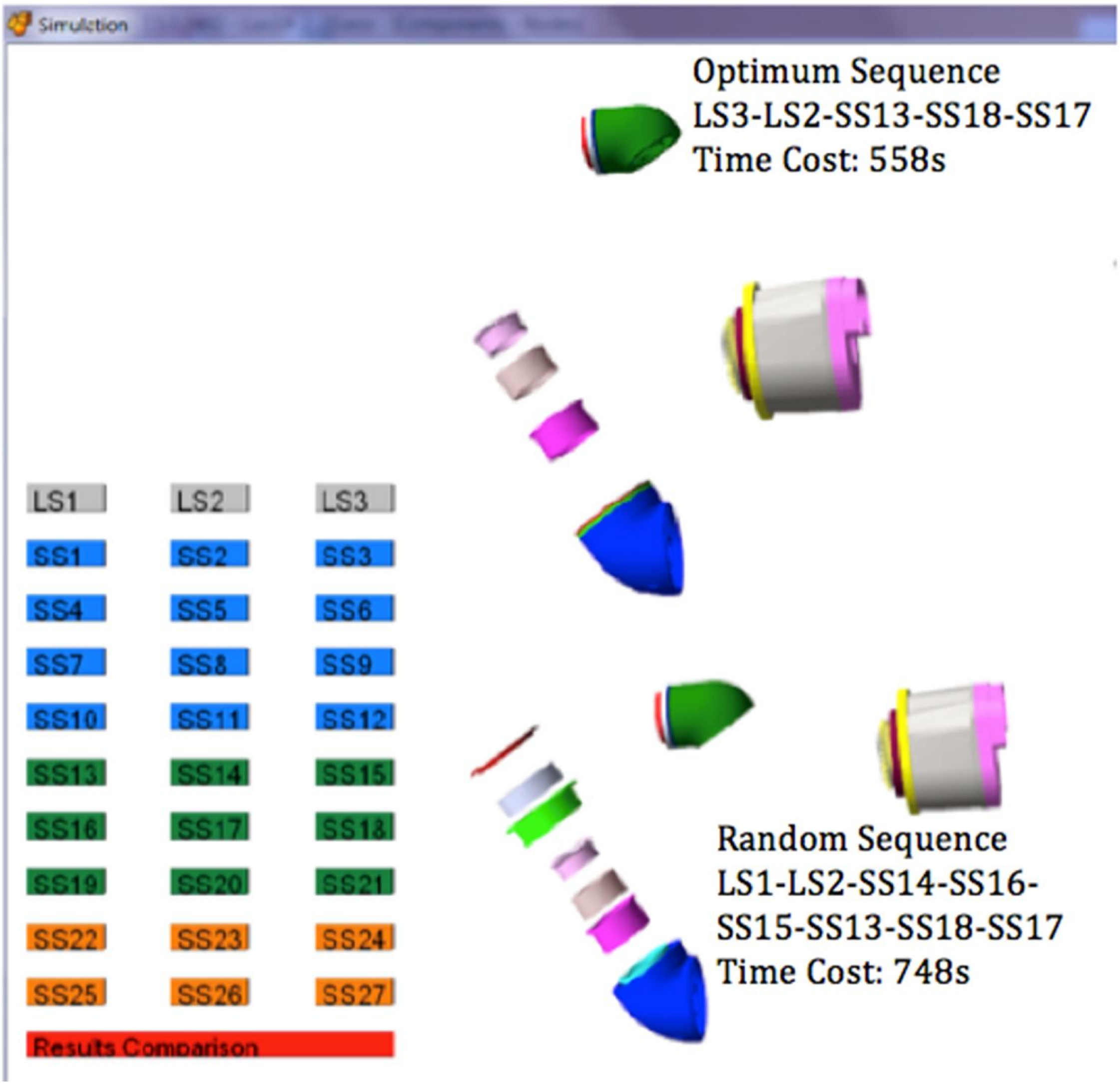


Fig. 6-7 Comparing the Two Different Disassembly Sequences of Model 1

6.2 Case two

Taking the gearbox in Fig. 3-3 as the second case, the model representation is shown in Fig. 3-5. A near optimum sequence is generated after loading the constraint matrices into the searching program and selecting the target part 8, it is "LS1-LS3-SS7-SS8". The total disassembly time for this sequence is 165s. Total components removed are 4. Searching process and searching result are shown in Fig. 6-8.

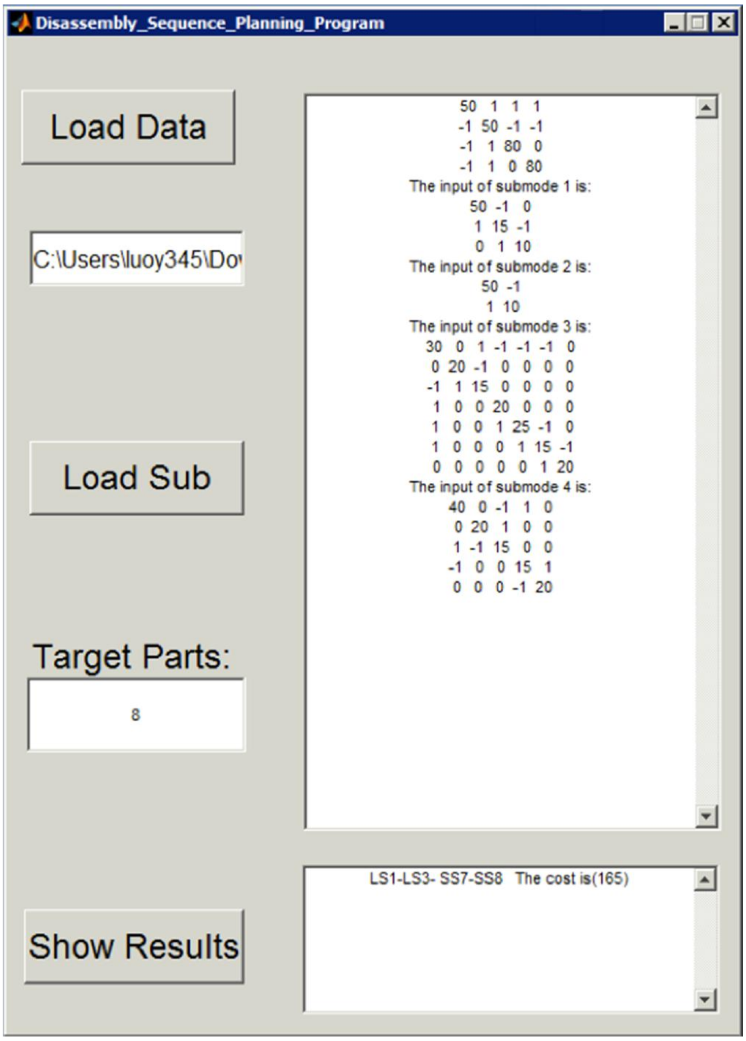


Fig. 6-8 Searching Case 2 for near Optimum Sequence

Fig. 6-9 shows the disassembly simulation and comparison of the near optimum sequence with another sequence. The comparison sequence is "LS1-LS3-SS12-SS11-SS10-SS7-SS8". Total components removed are 7, and the total time required is 225s.

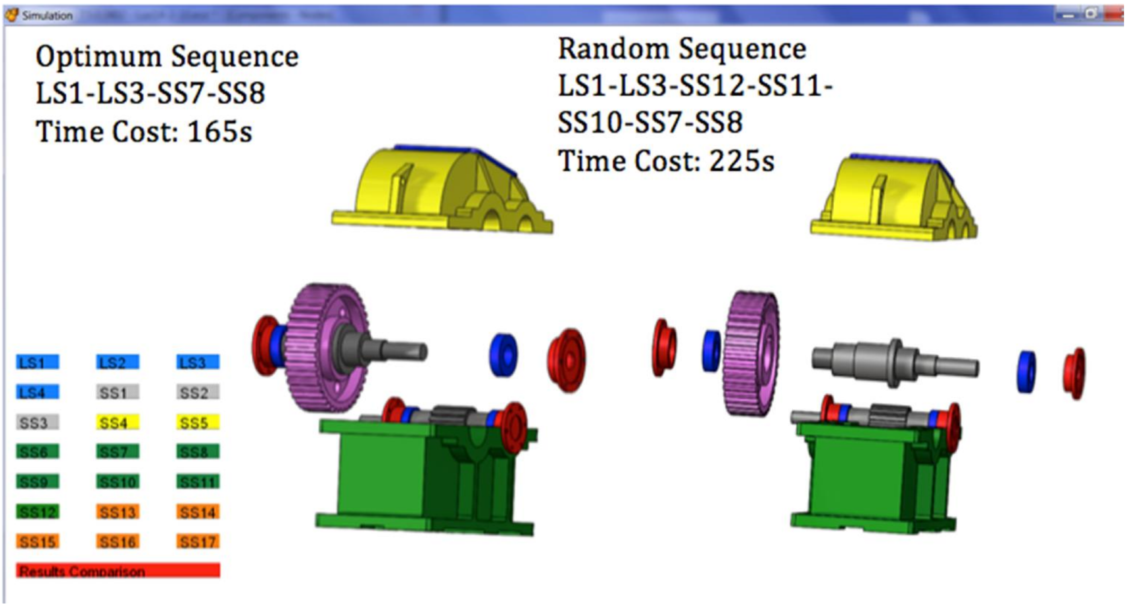


Fig. 6-9 Comparing Two Different Disassembly Sequences of Model 2

6.3 Case three

The transmission system model of a car shown in Fig. 4-5 is also used as a case to explain the proposed method. Fig. 4-8 and Fig. 4-9 show the constraint matrices of the model. Taking part No. 6 as the target part, the near optimum sequence is generated as "LS1-LS2-SS4-SS5-SS6", which is shown in Fig. 6-10. Two first-level subassemblies and three parts are removed, and the total disassembly time for this sequence is 524s. The simulation of the near optimum sequence and the comparison with another sequence is shown in Fig. 6-11. The total disassembly time for the comparison sequence is 549s, and three first-level subassemblies and three parts are removed. The number of parts removed of the near optimum sequence is fewer than the comparison sequence and the total disassembly time is also fewer.

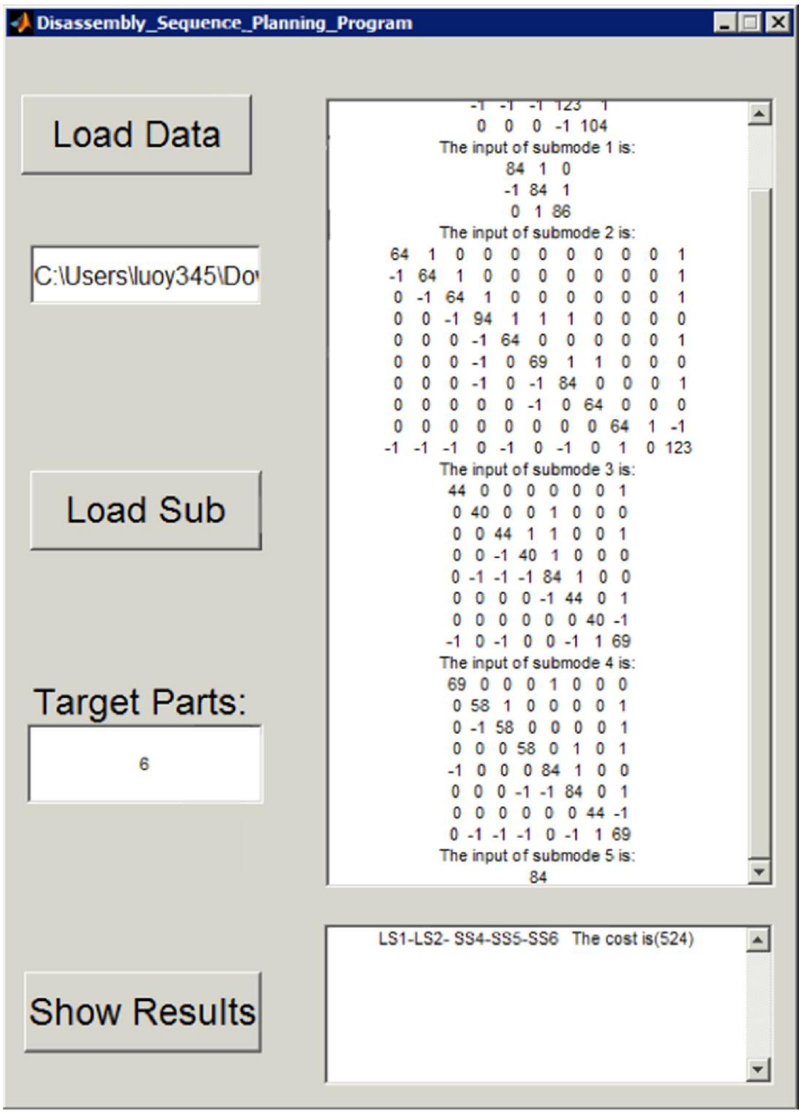


Fig. 6-10 Searching Case 2 for near Optimum Sequence

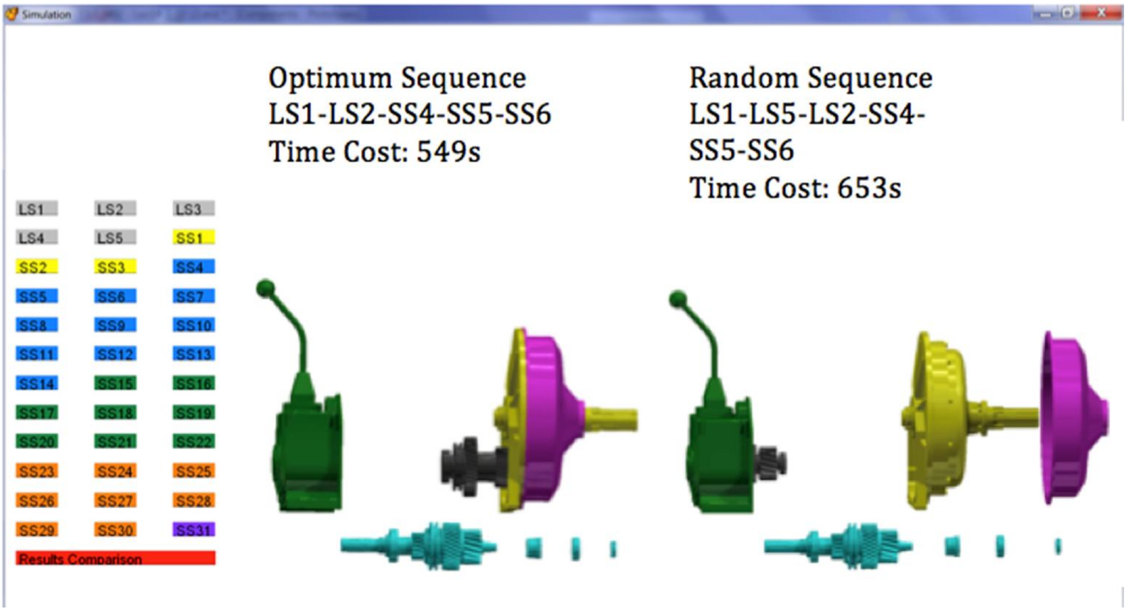


Fig. 6-11 Comparing Two Different Disassembly Sequences of Model 3

6.4 Summary

By applying to the three cases, the proposed representation and searching methods are explained in this chapter. The simulations show processes of disassembly operations. The comparisons of the near optimum sequences with random sequences provide verifications for the proposed method and research solutions.

7 Chapter 7

Conclusion and Future Work

7.1 Research contributions

It is demanded to recycle useful components or materials when products come to the end of their life. Before materials and components can be reused or recycled, a disassembly process is required. An efficient product disassembly sequence helps to save time, cost, and energy. In this research, an optimized method is proposed to deal with selective product disassembly sequence planning.

Based on the reviewed methods and the bill of materials, a multi-layer constraint matrices representation method is developed. The representation method considers the product structure and assembly procedures, thus, the optimum or near optimum sequence selected out has practical meaning. Compared to the one-level constraint matrix representation method, multi-layer matrices representation method can reduce the search size greatly in disassembly planning, which reduces the computational complexity of the search problem.

The ant colony searching method brings benefits when searching for the near optimal disassembly sequences. Whenever an unrealistic sequence is detected, a bunch of corresponding invalid sequences can be avoided using the ant colony searching method. If the target part is approached, the searching process stops without repeat searching. The ant colony searching method, together with the proposed multi-layer constraint matrices representation method, improves the efficiency of disassembly planning.

Factors including geometry complexity and process complexity are used for calculating the total removing time to evaluate different sequences. Based on the proposed method, the evaluation standards of minimum time cost and minimum number of components removed, a near optimum sequence can be decided. The evaluation criteria and the evaluation standards guarantee the reliability of the final solution.

The automatic search level of the proposed method is also improved by developing the searching program. The simulation of the sequences shows details of a product's disassembly procedures. The near optimum result can also be confirmed by comparing different disassembly procedures with the near optimum sequence.

7.2 Future Work

Even though the proposed method has improved the selective disassembly sequence planning, there are still some aspects to be further studied as follows.

The current searching process relies on the manual input of the product matrices. An automatic process should be developed to generate the constraint matrices based on the product design.

In this research, different procedures such as product modeling, product representation, searching for near optimum sequences, and simulation are conducted using different software tools and programs. These separated procedures should be built in one program in the future work.

The criteria for the solution evaluation are important for an accurate result. More factors in the product life cycle management should be included in the evaluation for different products and applications.

Other optimization algorithms such as genetic algorithms may also be studied to compare the final results.

Paper published related to this research

Qingjin Peng, Yongtao Luo, Peihua Gu, Cost-based Evaluation for Product Selective Disassemblability, A. Azevedo (ed.), Advances in Sustainable and Competitive Manufacturing Systems, Lecture Notes in Mechanical Engineering, Springer International Publishing Switzerland, P.45 -57, 2013.

Yongtao Luo, Qingjin Peng, Disassembly Sequence Planning For Product Maintenance, Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, IDETC/CIE 2012, August 12-15, Chicago, IL, USA, DETC2012-70430.

Reference

1. Jovane, F., Alting, L., Armillotta, A., Eversheim, W., Feldmann, K., Seliger, G. & Roth, N. (1993). A key issue in product life cycle: disassembly. *CIRP Annals-Manufacturing Technology*, *42*(2), 651-658.
2. Zhang, H., Kuo, T., & Lu H. (1997). Environmentally conscious design and manufacturing: a state-of-art survey. *Journal of Manufacturing System*, 16(5), 352-371.
3. Wong, J., Duan, G., & Lee, F. (1999). Current and future research situations on green manufacturing technology based on product lifecycle. *Computer Integrated Manufacturing System*, 5(4), 1-8.
4. Xie, Y., Huang, M., Zhong Y. & Kuang B. (2007). Disassembly sequence planning based on the simulated annealing and genetic algorithm. *Mechanical Engineer*, 1, 44.
5. Luo, Y., & Peng, Q. (2012). Disassembly sequence planning for product maintenance. *ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 601-609.

6. Tian, Y., Wang, T., He, G. & Ding B. (2009). Disassembly sequence planning based on generalized axis theory. *Journal of Mechanical Engineering*, 5(45), 160-170.

7. Pnueli, Y., & Zussman, E. (1997). Evaluating the end-of-life value of a product and improving it by redesign. *International Journal of Production Research*, 35(4), 921-942.

8. Kim, H. J., Lee, D. H., & Xirouchakis, P. (2007). Disassembly scheduling: literature review and future research directions. *International Journal of Production Research*, *45*(18-19), 4465-4484.

9. Behdad, S., & Thurston, D. (2010). Disassembly process planning trade offs for product maintenance. *ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 427-434.

10. Lambert, A. JD (2003). Disassembly sequencing: a survey. *International Journal of Production Research*, *41*(16), 3721-3759.

11. Li, J. R., Khoo, L. P., & Tor, S. B. (2002). A novel representation scheme for disassembly sequence planning. *The International Journal of Advanced Manufacturing Technology*, *20*(8), 621-630.

12. Homem de Mello, L. S., & Sanderson, A. C. (1990). AND/OR graph representation of assembly plans. *Robotics and Automation, IEEE Transactions on*, *6*(2), 188-199.

13. Min, S., Zhu, X. & Zhu, X. (2010). Mechanical product disassembly AND/OR graph construction. *2010 International Conference on Measuring Technology and Mechatronics Automation*, 627-631.

14. Li, J. R., Khoo, L. P., & Tor, S. B. (2005). An object-oriented intelligent disassembly sequence planner for maintenance. *Computers in Industry*, *56*(7), 699-718.

15. Kuo, T. C., & Wang, M. L. (2010). Waste electronics and electrical equipment disassembly and recycling using Petri net analysis. *Computers and Industrial Engineering (CIE), 2010 40th International Conference*, 1-6.

16. Chung, C., & Peng, Q. (2004). The selection of tools and machines on web-based manufacturing environments. *International Journal of Machine Tools and Manufacture*, *44*(2), 317-326.

17. Wu, N., & Peng, Q. (2009). Maintainability evaluation based on the product disassembly analysis. *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 101-109.

18. Hopcroft, J. E. (1983). Data structures and algorithms. Published by *Computer Science and Information Processing*.

19. Song, S., Hu, D., Gao, Y., Yang, M. & Zhang, L. (2010). Product disassembly sequence planning based on constraint satisfaction problems. *China Mechanical Engineering*, *21*(17), 2058-2062.

20. Zhang, L., Liu, Z., Yang, M. & Hu, D. (2011). Disassembly sequence planning based on interpretative structural model. *Journal of Computer-Aided Design and Computer Graphics*, 23(4), 667-675.

21. Gao, J., Dong, X., Chen, H., Duan, G. & Wang, J. (2003). Disassembly AND/OR graph model for" disassembly for recycling". *Electronics and the Environment,* 54-59.

22. Cao, T., & Sanderson, A. C. (1995). Task sequence planning using fuzzy petri nets systems*, Man and Cybernetics, 25*(5), 755-768.

23. Moore, K. E., Güngör, A., & Gupta, S. M. (2001). Petri net approach to disassembly process planning for products with complex AND/OR precedence relationships. *European Journal of Operational Research*, *135*(2), 428-449.

24. Zha, X. F. (2000). An object-oriented knowledge based Petri net approach to intelligent integration of design and assembly planning. *Artificial Intelligence in Engineering*, *14*(1), 83-112.

25. Hsieh, F. S. (2006). Robustness analysis of Petri nets for assembly/disassembly processes with unreliable resources. *Automatica*, *42*(7), 1159-1166.

26. Giudice, F. (2010). Disassembly depth distribution for ease of service: a rule-based approach. *Journal of Engineering Design*, *21*(4), 375-411.

27. Tang, Y., Zhou, M., Zussman, E., & Caudill, R. (2000). Disassembly modeling, planning and application: a review. *IEEE International Conference on Robotics and Automation,* 3, 2197-2202).

28. Chung, C., & Peng, Q. (2006). Evolutionary sequence planning for selective disassembly in de-manufacturing. *International Journal of Computer Integrated Manufacturing*, *19*(03), 278-286.

29. Pedrazzoli, P., Rinaldi, R., & Boer, C. R. (2001). A rule based approach to the gripper selection issue for the assembly process. *IEEE International Symposium on Assembly and Task Planning,* 202-207.

30. Zhang, G., Deng, K., Hou, Q. & Lai, S. (2006). Research on assembly modeling and sequence planning based on features. *International Technology and Innovation Conference,* 408-411.

31. Smith, S. S., & Chen, W. H. (2011). Rule-based recursive selective disassembly sequence planning for green design. *Advanced Engineering Informatics*, *25*(1), 77-87.

32. http://en.wikipedia.org/wiki/Genetic_algorithm. Accessed on April. 24, 2014.

33. Mohanta, J. C., Parhi, D. R., & Patel, S. K. (2011). Path planning strategy for autonomous mobile robot navigation using Petri-GA optimization. *Computers & Electrical Engineering*, *37*(6), 1058-1070.

34. Xie, Y. F., Huang, M. F., Zhong, Y. & Kuang, B. (2007). Disassembly sequence planning based on the simulated annealing and genetic algorithm. *Design and Calculate*, 103-105.

35. Dorigo, M. (1992). Optimization, learning and natural algorithms. *Ph. D. Thesis, Politecnico di Milano, Italy*.

36. Wang, L., Cai, R., Jing, L., & Zhang, H. (2012). Object-guided ant colony optimization algorithm with enhanced memory for traveling salesman problem. *Research Journal of Applied Sciences, Engineering and Technology*, 3999-4006.

37. Vollmann, Thomas E., Berry, William L., Whybark, D. Clay, Jacobs, F. Robert (2001). Manufacturing Planning and control for supply chain management. Published by McGraw-Hill/Irwin.

38. http://en.wikipedia.org/wiki/Bill_of_materials. Accessed on April. 16, 2014.

39. Eldos, T., Kanan, A., & Aljumah, A. (2013). Solving the printed circuit board drilling problem by ant colony optimization algorithm. *World Congress on Engineering and Computer Science*, 584-588.

40. Wang, H., Xiang, D., & Duan, G. H. (2006). A genetic algorithm for Product disassembly sequence planning. *Computer Integrated Manufacturing System, 12*(9), 1431.

41. Malik, S. (2010). Performance comparison between ant algorithm and modified ant algorithm. *International Journal of Advanced Computer Science and Applications*, 42-46.

42. Li, W., Peng, Q., & Xing, M. (2011). Bioreactor improvement based on design for assembly in virtual environments. *ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 733-742.

43. Jerry, B., & Carson, J. S. (1984). Discrete-event system simulation. *Pearson Education India*, 3.

44. http://www.eonreality.com/eon-studio. Accessed on Dec. 10, 2013.

45. http://www.solidworks.com. Accessed on Nov. 11, 2013.

46. Yu, J., Xu, L. D., Bi, Z. & Wang, C. (2013). Extended interference matrices for exploded view of assembly planning. *ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, 279-286.