

Product Assembly and Disassembly Sequence Optimization based on Genetic Algorithm and Design for Assembly Methodologies

Azman Yasin^a, Nurnasran Puteh^a, Ruslizam Daud^b

^aCollege of Arts and Sciences
Universiti Utara Malaysia, 06010 Sintok, Kedah
Tel : 04-9284701, Fax : 04-9284753
E-mail : yazman@uum.edu.my, nasran@uum.edu.my

^bSchool of Mechatronic Engineering
Universiti Malaysia Perlis, Kampus Kubang Gajah, 02600 Arau Perlis
Tel : 04-9798008, Fax : 04-9798227
E-mail : ruslizam@unimap.edu.my

ABSTRACT

Product assembly and disassembly sequences are the most crucial operations in product lifecycle. In this paper, an Artificial Intelligence (AI) technique, namely Genetic Algorithm (GA) is proposed to optimize product components assembly and disassembly sequences. The proposed methodology is developed and tested on an industrial product made of plastics with no integrated assembly and permanent joint parts. GA method is applied to determine the accuracy and optimum results based on 20 assembly and disassembly sequence solutions that was generated by the Design for Assembly methodology. The results indicated that GA based approach is able to obtain a near optimal solution for assembly and disassembly sequences.

Keywords

Artificial Intelligence, Genetic Algorithms, Assembly and Disassembly Sequences, Design for Assembly and Disassembly

1.0 INTRODUCTION

Product assembly and disassembly sequences are the most crucial operations in product lifecycle. Reports showed that cost of assembly as well as disassembly of manufactured products often contribute about 10% - 30% (sometimes higher) of the total manufacturing costs (Kara, Pornprasitpol *et al.* 2005). Thus, to reduce such costs, research works have been done to optimise the assembly and disassembly sequences. Optimizations of assembly and disassembly sequences are vital as it has important significance on productivity, product quality and manufacturing lead time (Veerakamolmal, Gupta *et al.* 1997). This paper discusses the development of Genetic Algorithm (GA) approach in optimizing the sequence of product components assembly and disassembly. Although assembly is a process of constructing a mechanical object from its component parts, it is not suitable to consider disassembly as the reverse of assembly. However, it can be relatively accepted that some of the basic principles of assembly being adapted to disassembly. Thus, this paper will demonstrate the

effectiveness and performance of the each proposed method through a case study of an industrial product which is the Phillips Diva Iron box. In addition, the Design for Assembly and Disassembly by the Boothroyd and Dewhurst (Boothroyd and Dewhurst, 1994) methodology is also implemented with the proposed methods to prove that GA can be applied as a tool for assembly and disassembly sequences.

2.0 REVIEW OF PREVIOUS RELATED WORKS

There have been a number of research works reported on methods of generating assembly sequence as well as disassembly sequence. The methods include tree search or graph search methods, neural network-based approaches, simulated annealing as well as genetic algorithms. According to D.S. Hong and H.S. Cho (Hong and Cho, 1999), although the developed methods can find optimal method in assembly sequence, it is limited to small number of part components.

Meanwhile, the work by B. Lazzarini and F. Marcelloni (Lazzarini and Marcelloni, 2000) generates a near-optimal assembly plans from a randomly initialized population of assembly. It is evaluated based on three optimization criteria, namely: (i) minimizing the orientation changes of the product; (ii) minimizing the gripper replacements; and (iii) grouping technologically similar assembly operations.

3.0 SYSTEM DEVELOPMENT

In this section, the paper explains on the method of developing the system. Initially, the research problem is to minimize the total assembly and disassembly time by proper feeder allocation and component sequencing. The standard Design for Assembly (DFA) table of manual handling and manual insertion from (Beno, 2003) were used to estimate the time of assembly and disassembly which is measured in second. The design features that significantly affect part manual handling and manual removal time are the accessibility of assemble and disassemble location, ease of operation of assemble and disassembly tool,

visibility of location, ease of alignment and positioning during assembly and disassembly, and the depth of removal. Meanwhile, the manual insertion represents the time taken for part insertion to form a unit product. On the other hand, the part features that affect manual insertion time significantly are size, thickness, weight, nesting, tangling, fragility, flexibility, slipperiness and stickiness. It also took into consideration the necessity for using two hands, necessity for using grasping tools, necessity for optical magnification and necessity for mechanical assistance.

A graphical representation of the above problem is shown in Table 1. There are three columns of nodes representing, respectively, the time of insertion (for assembly) or removal (for disassembly), the time of handling, and the number of components. The first column is labeled from 0 to $I(i)$, the handling time are from 0 to $H(i)$, and the components are labeled from 1 to i . The first part of the calculations is to calculate the time of part removal, the second part calculates the time of handling, and the last part total up the time of removal and handling to get the disassembly time.

Table 1: Graphical representation

Insertion/ Removal time	Handling time	Number of Components
0	0	1
↓	↓	↓
$I(i)$	$H(i)$	i

The solution to this research problem has been solved separately for assembly sequence and disassembly sequence. Both parts have applied genetic algorithm approach for optimizing assembly and disassembly sequences. Fig 1 shows the framework used for the GA

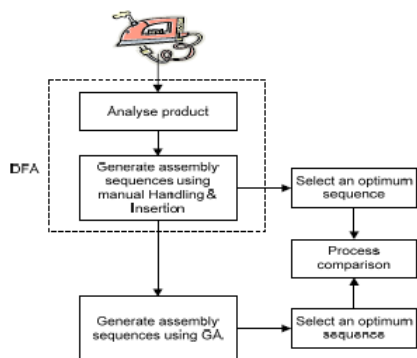


Figure 1: Framework of assembly and disassembly sequence optimization

based assembly and disassembly sequence optimization respectively. In this study, we have compared the GA based assembly and disassembly sequences with the manual assembly sequences using DFA methodology by Boothroyd and Dewhurst (Boothroyd and Dewhurst, 1994).

In order to apply the GA method, its inputs or chromosomes have to be defined. The chromosomes for this study will be parts of the product chosen, which is the Phillips Diva Iron box. Each chromosome has been given a code as shown in Table 2 below.

Table 2: Parts Name and their codes

Parts Name	Code	Item	Module
Outer Screw	1, 2	2	A
Upper Handle	3	1	
Cable Tube	5	1	
Handle Screw	7	1	
Cable Screw	8, 9	2	
Cable Plate	10	1	
Lower Handle	11	1	B
Temperature Dial	12	1	
Power Indicator	13	1	
Temperature Diode	14, 15	2	
Wire	16, 17, 18	3	
Cable Tube	6	1	
Upper Handle	4	1	C
Cable	19	1	
Heat Tuner	20	1	
Soleplate Screw 1	21	1	
Soleplate Screw 2	22, 23	2	
Soleplate Cover	24	1	
Rubber Hinge	25, 26	2	
Soleplate Assy.	27	1	

GA is used as a tool in generating the optimum assembly and disassembly sequence. The detail of GA structure applied in this research is shown in Fig 2. GA engine is built using JAVA programming language and the KAWA integrated development environment. The GA process is explained as followed:

Step 1: Generate populations

The sequence that is generated using Design for Assembly or Disassembly manually will act as the input for the GA process. In this project, there were 30 sequences which are called chromosomes in a population.

Step 2: Selection (Roulette Wheel)

Firstly, the fitness of each chromosome, f_i ($i = 1$ to 30) is calculated, which is the estimated time taken in each of the assembly or disassembly sequence.

$$f_i = TA_i$$

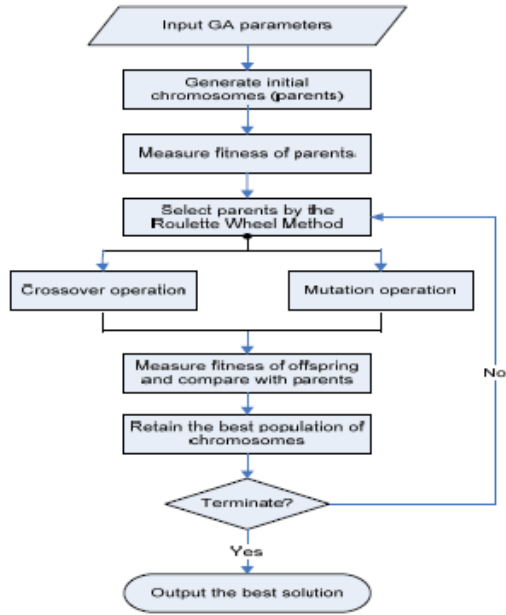


Figure 2: GA framework for assembly and disassembly sequence optimization

Secondly, the total fitness, F of the population is calculated,

$$F = \sum f_i$$

Next, the probability of each chromosome, rF_i ($i = 1$ to 30) is calculated,

$$rF_i (i = 1 = 30)$$

$$rF_i = \frac{f_i}{F}$$

Then the cumulative probability for each chromosome, cF_i is calculated,

$$cF_i = rF_i \quad (i = 1)$$

$$cF_i = rF_i + cF_{i-1} \quad (i = 2 \text{ to } 30)$$

Finally, a random number, r , in the range 0 to 1, is generated

$$\text{If } cF_{i-1} < r < cF_i \quad C_i \text{ is selected}$$

Step 3: Crossover process (one point crossover) A random number in the range (0, gene number – 1) as crossover point,

cr is generated. Two chromosomes at the crossover point is then crossed as shown in Fig 3.

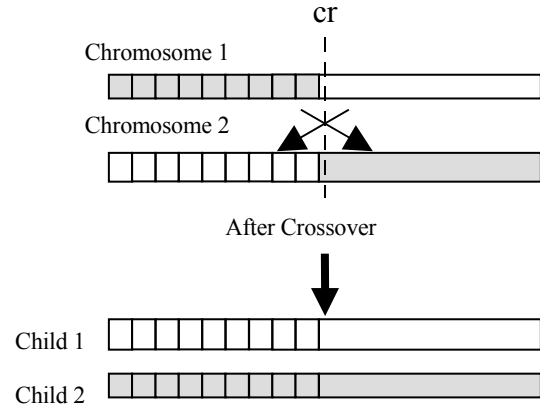


Figure 3: Crossover Process

After crossover process, the fitness of child 1 and child 2 were calculated. They will replace the worst chromosomes in the population where their fitness were lower than the fitness of these children.

Step 4: Mutation process

A random number in the range 0 to 1, r is generated. If r less than the mutation rate, mr , the mutation process will be carried out

$$r < mr \rightarrow \text{mutation process}$$

If after mutation process, the sequence of assembly or disassembly chromosomes did not follow the precedence rules, the mutation process is rejected.

Step 5: Stopping condition

When the fitness of the chromosomes is consistent, it means that the best result has occurred. Then, the GA process will be terminated.

4.0 RESULTS AND DISCUSSIONS

Two main results will be highlighted and discussed. First part concentrates on assembly sequence while the second part focuses on disassembly sequence.

4.1 Assembly sequence results

In this experiment, it shows that by applying GA for a product with n components, the population undergoes most significant variation in an early searching stage. It will ends after approximately n generation when it reaches a stable state. Fig 4 shows the fitness value for each generation. For this study, the program has been executed for 200 generations.

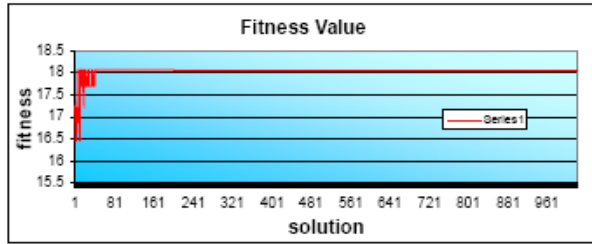


Figure 4: Fitness value for each chromosome

Based on Fig 4, it shows a fluctuation at an early stage in fitness value of each chromosome but it started to be stable after few generations. The highest value of fitness (where it is stable) is considered to be the solution. In this case, it shows that the best fitness value is 18.07991%. The solution that produced the highest fitness value is shown below.

Best Chromosome sequence based on part codes:

- 1, 3, 2, 4, 6, 7, 5, 8, 16, 24, 14, 19, 17, 18, 13, 12, 9, 10, 11, 20, 21, 22, 23, 15, 25, 27, 26
- 1, 2, 3, 4, 5, 62, 19, 18, 17, 14, 24, 16, 8, 72, 12, 13, 10, 11, 9, 20, 21, 22, 23, 15, 25, 26, 27
- 1, 25, 26, 20, 9, 11, 10, 72, 12, 13, 4, 5, 62, 19, 18, 17, 14, 24, 16, 8, 3, 2, 22, 23, 15, 21, 27

Best All Chromosome sequence based on part codes:

- 1, 3, 2, 4, 6, 7, 5, 8, 16, 24, 14, 19, 17, 18, 13, 12, 9, 10, 11, 20, 21, 22, 23, 15, 25, 27, 26

The optimum solution or sequence produced from GA is then tested in DFA worksheet to see its accuracy in producing the best result as manual assembly in DFA. The time to assemble an iron using the sequence produced from GA is efficient as it results in the minimum time among all assembly sequences that is 165.93 minutes. On the other hand, the traditional DFA practice produced 174.03 minutes.

4.2 Disassembly sequence results

Similar to assembly sequence, the population searching underwent most significant variation in the early searching stage. It ended after approximately n generations when it reached the stable state. Fig 5 and 6 show the fitness value for each generation. For this study, 50 generations have been executed.

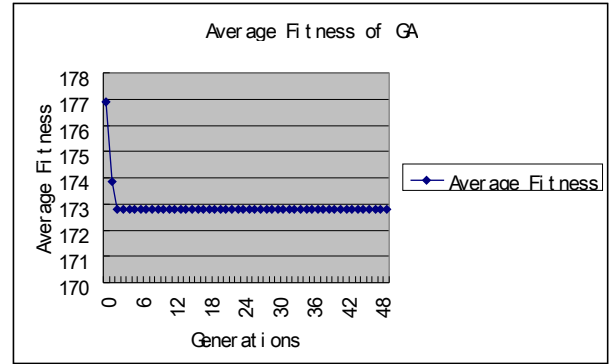


Figure 5: Average fitness value for each chromosome

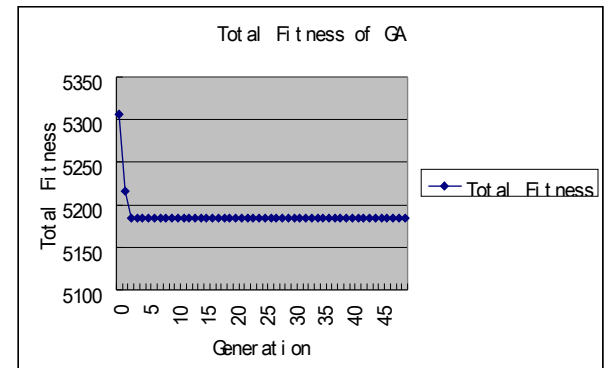


Figure 6: Total Fitness of GA

Based on Fig 5 and 6, the fitness of chromosomes decreased in the early stage. After that, it started to be stable and reached a static condition. This means that the best result has already occurred. The best fitness where it is stable is considered as the solution. In this case, it showed that the best fitness value is 172.81 seconds, while manual disassembly time is about 171.16 seconds. The best chromosome that produced the highest fitness value is shown below and this is the optimum disassembly sequence.

Best Chromosome sequence based on part codes

- 1, 2, 3, 7, 8, 9, 10, 5, 11, 13, 12, 14, 15, 16, 17, 18, 6, 4, 19, 20, 22, 23, 21, 24, 25, 26, 27

5.0 CONCLUSION

As the study showed, the strength of GA in searching an optimum solution has been proven. Results show that a near-optimal solution is produced although the initial convergence began at early generations. However, based on the experiment, GA does not guarantee that an optimal solution can be achieved for each process. The main reason is GA used the roulette wheel method in selecting new populations in this study.

Although one chromosome has the highest fitness, there is no guarantee that it will be selected. On the average, only the chromosome with the proportional fitness will be chosen. Thus, this GA can be further enhanced with other AI techniques to obtain a more efficient selection of optimum assembly and disassembly sequences. Nevertheless, the advantage of using this technique is that, it can be implemented at the design process and can obtain faster result than the traditional methods.

REFERENCES

- Hong, D.S and Cho, H.S. (1999). A genetic-algorithm based approach to the generation of robotic assembly sequences. *Journal of Control Engineering Practice*, 7, pp. 151 – 159.
- Lazzerini, B and Marcelloni, F. (2000). A genetic algorithm for generating optimal assembly plans. *Journal of Artificial Intelligence in Engineering*, 14, pp. 319 – 329.