International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:6, No:6, 2012

A Green Design for Assembly Model for Integrated Design Evaluation and Assembly and Disassembly Sequence Planning

Yuan-Jye Tseng, Fang-Yu Yu, Feng-Yi Huang

***Abstract*—**A green design for assembly model is presented to integrate design evaluation and assembly and disassembly sequence planning by evaluating the three activities in one integrated model. For an assembled product, an assembly sequence planning model is required for assembling the product at the start of the product life cycle. A disassembly sequence planning model is needed for disassembling the product at the end. In a green product life cycle, it is important to plan how a product can be disassembled, reused, or recycled, before the product is actually assembled and produced. Given a product requirement, there may be several design alternative cases to design the same product. In the different design cases, the assembly and disassembly sequences for producing the product can be different. In this research, a new model is presented to concurrently evaluate the design and plan the assembly and disassembly sequences. First, the components are represented by using graph based models. Next, a particle swarm optimization (PSO) method with a new encoding scheme is developed. In the new PSO encoding scheme, a particle is represented by a position matrix defining an assembly sequence and a disassembly sequence. The assembly and disassembly sequences can be simultaneously planned with an objective of minimizing the total of assembly costs and disassembly costs. The test results show that the presented method is feasible and efficient for solving the integrated design evaluation and assembly and disassembly sequence planning problem. An example product is implemented and illustrated in this paper.

***Keywords*—**green design, assembly and disassembly sequence planning, green design for assembly, particle swarm optimization.

I. Introduction

TO produce an assembled product, the components of the product are designed and manufactured first. Subsequently, the components are located and fixed with the assembly operations to construct the final product. The purpose of assembly sequence planning is to determine a proper sequence with which the components can be fixed with the assembly operations. In assembly sequence planning, the spatial and connecting relationships of the components are analyzed.

Yuan-Jye Tseng is with the Department of Industrial Engineering and Management, Yuan Ze University, Chung-Li, Taoyuan 320, Taiwan (phone: +886-3-4638800; fax: +886-3-4638907; e-mail: ieyjt@saturn.yzu.edu.tw).

Feng-Yi Huang and Fang-Yu Yu are with the Department of Industrial Engineering and Management, Yuan Ze University, Chung-Li, Taoyuan 320, Taiwan (e-mail: s948909@mail.yzu.edu.tw).

The components and the assembly operations are arranged in an ordered sequence under the operational constraints and precedence constraints to achieve the assembly cost objectives.

In a complete product life cycle of an assembled product, both an assembly sequence and a disassembly sequence are required. An assembly sequence is required to construct the product at the start of the product life cycle. On the other hand, a disassembly sequence is required to disconnect the components of the product at the end of the product life cycle. A disassembly sequence can be defined as an ordered sequence of components and disassembly operations with which the product can be decomposed into separated components. The purpose of disassembly sequence planning is to arrange the order of disassembly operations based on the disassembly constraints and cost objectives.

In a green product life cycle, it is necessary to plan how a product can be disassembled before the product is actually assembled and produced. In a green product life cycle, although the disassembly operations occur at the end, it is important to plan in advance at the start. In most cases, a low cost in the assembly operations can cause a high cost in the disassembly operations. Therefore, to avoid a high cost in disassembly sequences, the key is to consider the assembly and disassembly sequences in a concurrent way.

In the beginning of the product life cycle, the shapes of the components and product are designed in the design stage. Given the product requirements, there can be different design alternative cases that can be used to fulfill the product requirements. Different shapes of the components can be modeled, manufactured, and assembled, to attain the specified functional purpose and product requirements.

If the components are designed differently in the design alternative cases, the assembly and disassembly sequences can be different. In typical design considerations, the functional purpose is considered as the main concern. As a result, the considerations in design may contradict the considerations in production. For example, a good design with good functions can cause some difficulties in the corresponding assembly and disassembly sequences. In this way, a design case without considering production may result in a high cost in the subsequent assembly and disassembly sequences.

In the traditional concept of product life cycle management, the activities of design, manufacturing, assembly, and disassembly are performed in a sequential way. The drawback is that the considerations in design may not always in accord with the considerations in assembly and disassembly. In the typical

design for assembly models, the assembly and disassembly sequences have not been planned and evaluated in a concurrent way. Furthermore, the effects of the design alternative cases on the assembly and disassembly sequences have not been evaluated with an integrated model. Therefore, it requires an integrated design evaluation and assembly and disassembly sequence planning model to achieve a green product life cycle.

With the above concept, a green design for assembly and disassembly model is developed in this research, as shown in Fig. 1. In this research, an integrated design evaluation and assembly and disassembly sequence planning model is presented. The flow of the model is shown in Fig. 2. Given a product requirement, there may be several design alternative cases for designing the components of the product. If a design case is selected, the shapes of the components may be different from the other cases. The spatial and connection relationships of the components are thus different. The spatial and connection relationships can change the operational constraints and operational costs in the assembly and disassembly sequences. Therefore, if a different design case is selected, the assembly and disassembly sequences can be different.

As shown in Fig. 2, for different design cases, the assembly and disassembly constraints are different. In addition, for different design cases, the assembly and disassembly costs are different. With the constraints and costs, the assembly and disassembly sequences can be arranged in orders. Therefore, given a design case, the corresponding assembly and disassembly sequences can be generated and evaluated. In this way, the design cases and the corresponding assembly and disassembly sequences can be evaluated and planned in one integrated model.

In this paper, first, the designed components are represented by using graph based models. The graph based models represent the spatial and connecting relationships of the components. The graph based models are transformed into assembly and disassembly precedence constraints. In addition, the cost models of the assembly and disassembly operations can be developed. A particle swarm optimization (PSO) approach is presented by encoding a particle using a position matrix defined by the assembly and disassembly sequences. The PSO method simultaneously performs evaluation of assembly and disassembly sequences with an objective of minimizing the total cost. By using the total cost as the fitness function, the design cases and the assembly and disassembly sequences can be evaluated and optimized.

The presented models and algorithms have been implemented and tested. The main contribution lies in the new concept of integrated design evaluation and assembly and disassembly sequence planning model and the new PSO solution method. The test results show that the presented method is feasible and efficient for solving the integrated design evaluation and assembly and disassembly planning problem. In this paper, the test result of an example product is illustrated and demonstrated.

This paper is organized as follows. Section II presents a literature review. Section III describes the model for integrated design evaluation and assembly and disassembly sequence planning.

Section IV presents the PSO method. Implementation and test results are presented in Section V. The conclusions are discussed in Section VI.

II. Literature Review

In the related research, it can be summarized that assembly sequence planning can be performed with three stages: (1) assembly representation and modeling, (2) assembly sequence generation, and (3) assembly sequence evaluation and optimization. Lin and Chang [1] presented an assembly precedence diagram (APD) which is a directed graph representing the precedence of the components and the associated assembly operations. In Abdullah *et al*. [2], a review of assembly sequence planning methods was presented. Lai and Huang [3] presented a systematic approach for automatic assembly sequence generation. Chen and Lin [4] presented optimizing assembly planning through a three-stage integrated approach. Su [5] introduced a geometric constraint analysis method to generate assembly precedence and to evaluate feasible assembly sequences. Dong *et al*. [6] presented an assembly tree hierarchy to analyze geometric and non-geometric information for assembly sequence planning. In the recent research, Tseng *et al*. [7] presented a multi-plant assembly sequence planning model using a GA method to integrate assembly sequence planning and plant assignment. Jin *et al*. [8] presented an assembly sequence optimization method for complex mechanical product by employing a directed graph and an assembly matrix to represent the assembly relation. In Tseng *et al*. [9], the genetic algorithm (GA) was applied to solve the integrated assembly and disassembly sequence planning problem.

With a given set of components, sequencing the components may become a combinatorial problem. In the previous research, the PSO algorithm has been shown to be effective and efficient to find solutions for different optimization problems. The PSO algorithm has been successfully applied to many continuous and discrete optimizations [10]-[11]. Banks *el al*. [12] reviewed and summarized the related PSO research in the areas of combinatorial problems, multiple objectives, and constrained optimization problems.

In review of the previous research, the design evaluation and assembly and disassembly sequences have not been evaluated or planned in an integrated way. Therefore, in this research, the PSO method with a new encoding scheme is developed for concurrently performing design evaluation and assembly and disassembly sequence planning.

III. Representation Models

In this paper, the input design information is defined by using graph based models. An assembly precedence graph (APG) and a disassembly precedence graph (DPG) are modeled to represent the connecting and precedence relationships between the components. The graph based models are transformed to an assembly precedence matrix (APM) and a disassembly precedence matrix (DPM) to represent the precedence

constraints. The PSO method is developed by encoding a particle with a position matrix defined by the assembly and disassembly sequences. The PSO method simultaneously performs assembly and disassembly sequencing with an objective of minimizing the total cost. The best design case can be determined by finding the position with the lowest total cost.

A. Assembly Precedence Graph (APG) and Disassembly Precedence Graph (DPG)

In this research, the APG and DPG are used as the design input for a given product. Given the geometric definition of the components, the spatial and connecting relationships can be analyzed to generate the information in the graph based models of APG and DPG. The APG is modeled for representing the precedence relationships of the components and assembly operations. The DPG is modeled for representing the precedence relationships of the components and the disassembly operations.

$$APG \text{ is a directed graph } G = (C, A), \quad (1)$$
$$DPG \text{ is a directed graph } G = (C, S), \quad (2)$$

where $C = \{c_1, \ldots, c_n\}$ = the set of components,
c_i = (*component node*) = a component, $i = 1, \ldots, n$,
$A = \{a_1, \ldots, a_m\}$ = the set of assembly operation arcs between component nodes,
$S = \{s_1, \ldots, s_m\}$ = the set of disassembly operation arcs between component nodes.

As shown in Fig. 3, product A is a notebook computer with 16 main components. The APG and DPG of product A is shown in Fig. 4.

B. Assembly Precedence Matrix (APM) and Disassembly Precedence Matrix (DPM)

An APG is transformed into an assembly precedence matrix (APM). A DPG is transformed into a disassembly precedence matrix (DPM). The numerical values in the two matrices are used in the PSO solution method. The APM and DPM models are shown as follows

$$APM = \begin{array}{c} \\ c_{i=1} \\ c_{i=2} \\ \vdots \\ c_{i=n} \end{array} \begin{array}{c} \begin{array}{cccc} c_{j=1} & c_{j=2} & \cdots & c_{j=n} \end{array} \\ \begin{bmatrix} p_{11} & p_{12} & \ldots & p_{1n} \\ p_{21} & p_{22} & \cdots & \\ \vdots & \vdots & & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} \end{array}, \quad (3)$$

$$DPM = \begin{array}{c} \\ c_{i=1} \\ c_{i=2} \\ \vdots \\ c_{i=n} \end{array} \begin{array}{c} \begin{array}{cccc} c_{j=1} & c_{j=2} & & c_{j=n} \end{array} \\ \begin{bmatrix} p_{11} & p_{12} & \ldots & p_{1n} \\ p_{21} & p_{22} & \cdots & \\ \vdots & \vdots & & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} \end{array}, \quad (4)$$

where c_i and c_j are components,
p_{ij} is an index where $p_{ij} = 1$ represents that component c_j must be assembled or disassembled before component c_i.

The APM and DPM for product A are listed as follows.

$APM =$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	1	1
1	1	0	0	0	1	0	1	0	0	0	1	0	0	1	1	1
2	0	0	0	0	1	0	0	1	1	0	1	1	0	1	1	0
3	0	0	0	0	1	0	0	1	1	0	1	1	0	1	1	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	1	0	0	1	1	0	1	1	0	1	1	0
6	1	0	0	0	1	0	0	0	0	0	1	0	0	1	1	1
7	0	0	0	0	1	0	0	0	0	0	1	0	0	1	1	0
8	0	0	0	0	1	0	0	0	0	0	1	0	0	1	1	0
9	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
10	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0
11	0	0	0	0	1	0	0	1	1	0	1	0	0	1	1	0
12	1	0	0	0	1	0	1	0	0	0	1	0	0	1	1	1
13	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	1	0	0	0	0	0	1	0	0	1	1	0

$DPM =$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
3	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
6	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
7	1	1	1	1	0	1	1	0	0	1	0	1	1	0	0	1
8	1	1	1	1	0	1	1	0	0	1	0	1	1	0	0	1
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1
11	1	1	1	1	0	1	1	0	0	1	0	0	1	0	0	1
12	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
13	1	1	1	1	0	1	1	0	0	1	0	1	1	0	0	1
14	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
15	1	1	1	1	0	1	1	0	0	1	0	0	1	0	0	0

C. Design Case Table

Design alternative cases are represented in a table format. Given a product requirement, the design of a set of components can be represented as a design case. With the same product requirement, some of the components may be changed for designing the same product. As shown in Fig. 2, if some of the components are changed in design, the assembly constraints and the assembly costs will be changed. In this way, the APG, APM, DPG and DPM are different. Therefore, in different design cases, the assembly sequences can be affected by the constraints and costs. As a result, if a different design case is selected, the corresponding assembly and disassembly sequences need to be generated and evaluated. Using a systematic evaluating with the PSO method, the design and assembly and disassembly sequences can be concurrently evaluated.

A design case table (DCT) is developed for use in the design representation. The general format of a DCT is shown in Table I. In the table, given an original design and its design alternative cases $d_j = 1, \ldots, m$, a value of $t_{ij} = 1$ indicates that the component c_i is changed in design d_j. A value of $t_{ij} = 0$ indicates that the component c_i is not changed in design d_j. The information in the DCT is used to analyze and generate the information in the APG, APM, DPG, and DPM.

If a different design case is selected, a different set of components are used in modeling of APG, APM, DPG, and DPM. If different APG, APM, DPG, and DPM are built, the assembly and disassembly sequences will be affected. Therefore, given a design case, the corresponding assembly and disassembly sequences can be generated and evaluated.

As a result, the design evaluation and assembly and disassembly sequence planning can be integrated.

IV. Solution Using The Particle Swarm Optimization (PSO) Method

A PSO method is presented for concurrently performing design evaluation and assembly and disassembly sequence planning. The PSO algorithm is an evolutionary computation method introduced by Kennedy and Eberhard (1995, 1997). In the PSO method, each particle moves around in the multi-dimensional space with a position and a velocity. The velocity and position are constantly updated by the particle's own experience and the experience of the whole swarm. Given a problem, a particle can be encoded to represent a solution. Each solution, called a particle, flies in the search space towards the optimal position.

In the original PSO method, a particle is defined by its position and velocity. The position of a particle i in the D-dimension search space can be represented as $X_i=[x_{i1}, x_{i2}, \ldots, x_{id}, \ldots, x_{iD}]$. The velocity of the particle i in the D-dimension search space can be represented as $V_i=[v_{i1}, v_{i2}, \ldots, v_{id}, \ldots, v_{iD}]$. Each particle has its own best position $P_i=[p_{i1}, p_{i2}, \ldots, p_{id}, \ldots, p_{iD}]$ representing the particle's personal best objective (*pbest*) at time t. The global best particle is denoted as p_g and the best position of the entire swarm (*gbest*) is denoted as $P_g=[p_{g1}, p_{g2}, \ldots, p_{gd}, \ldots, p_{gD}]$ at time t. To search for the optimal solution, each particle adjusts its velocity according to the velocity updating equation and position updating equation.

$$v_{id}^{new} = w_i \cdot v_{id}^{old} + c_1 \cdot r_1 \cdot \left(p_{id} - x_{id}\right) + c_2 \cdot r_2 \cdot \left(p_{gd} - x_{id}\right) \quad (5)$$

where d =1, …, D, i =1, …, E (number of particles),

v_{id}^{new} :the new velocity of i in the current iteration t,

v_{id}^{old} : the velocity of i in the previous iteration (t - 1),

c_1 and c_2: constants called acceleration coefficients,

w_i : the inertia weight,

r_1 and r_2: two independent random numbers with a uniform distribution [0, 1],

p_{id} : the best position of each individual particle i,

p_{gd} : the best position of the entire swarm.

$$x_{id}^{new} = x_{id}^{old} + v_{id}^{new}, \quad (6)$$

where x_{id}^{new} is the new position in the current iteration t, x_{id}^{old} is in the previous iteration (t - 1).

A. Encoding and Decoding Scheme

This research applies the PSO method to the problem by developing a new encoding and decoding scheme. In the developed encoding scheme, a particle is represented by a position matrix (PM). The position of a particle p = 1, …, E, is represented by a position matrix PM_{AD}. The elements in the position matrix is denoted as X_{ij}, i = 1, 2, 3, and j = 1, …, n, where n is the number of components. The first row, where i = 1, represents an assembly sequence. The second row, where i = 2, represents a disassembly sequence. The third row, where i = 3, represents the design case index of each component. In the heuristic decoding rule, the values in the first row (i = 1) represent the ranked order values of the n components in the assembly sequence. The values in the second row (i = 2) represent the ranked order values of the n components in the disassembly sequence.

$$PM_{AD} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,j} & \cdots & x_{1,m} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,j} & \cdots & x_{2,m} \\ x_{3,1} & x_{3,2} & \cdots & x_{3,j} & \cdots & x_{3,j} \end{bmatrix}, \quad (7)$$

X_{ij} represents the position matrix of particle p, where p = 1, …, E, and i =1, 2, 3, where a value of i = 1 represents an assembly sequence, and i = 2 represents a disassembly sequence, and i = 3 represents the design case index, and j =1, …, n, where n is the number of components.

After the PSO enumeration, the final particle represents an assembly sequence and a disassembly sequence. The position matrix of a particle can be decoded into an assembly sequence and a disassembly sequence. A heuristic rule for decoding is presented as follows. In the first row, the position values in $[x_{11}, \ldots, x_{1j}, \ldots, x_{1n}]$ are sorted in an ascending order. The ranked order value represents the ordered position in the decoded sequence. For example, if x_{1j} is ranked in the 5th position in the sorted ascending order, then the component C_j is assigned to the 5th position in the assembly sequence. In the second row, for example, if x_{2j} is ranked in the 6th position in the sorted ascending order, then the component C_j is assigned to the 6th position in the disassembly sequence. The values in the third row are used to determine selection of the design cases of the components.

B. Fitness Function

The cost functions include two major items. The assembly operational costs are mainly related to assembly sequencing, whereas the disassembly related costs are primarily related to disassembly sequencing.

1) Assembly operation cost (*AOC*) and disassembly operation cost (*DOC*): the basic operational cost for performing an assembly or disassembly operation.
2) Assembly tool change cost (*ATC*) and disassembly tool change cost (*DTC*): if two tools are different, then an assembly or disassembly tool change cost is required.
3) Assembly setup change cost (*ASC*) and disassembly setup change cost (*DSC*): if two consecutive setups are different, then an assembly setup change cost or a disassembly setup change cost is required.
4) Design related cost (*DRC*): Proper design related cost for designing and changing the components in the design alternative cases.

The total cost function (*TC*) can be formulated as follows (unit: dollars).

$$TC = AOC+DOC+ATC+DTC+ASC+DSC+DRC \quad (8)$$

In the PSO evaluation, the objective is to minimize the fitness function as follows. The fitness function value of a particle is represented as *Fitness*.

$$\text{Min } Fitness = TC \quad (9)$$

C. Integrated Design Evaluation and Assembly and Disassembly Sequence Planning

The flowchart of the method is shown in Fig. 5.

Step 1. Setup parameters.
1) Set iteration $t = 0$.
2) T_{Number}: the iteration (generation) number.
3) P_{Size}: the number of particles.

Step 2. Initialize a population of particles $i = 1, \ldots, E$, with random positions and velocities.
1) A particle i is defined by a multi-dimensional position matrix as shown in equation (7).
2) The position of particle i is defined by X_{ij}.
3) The velocity of particle i is defined by V_{ij}.

Step 3. Evaluate the fitness function.
1) $t = t + 1$.
2) $Fitness = TC$.

Step 4. Update the velocity of each particle i.

$v_{id}^{new} = w_i \cdot v_{id}^{old} + c_1 \cdot r_1 \cdot (p_{id} - x_{id}) + c_2 \cdot r_2 \cdot (p_{gd} - x_{id})$,

v_{id}^{new} is the new velocity in the current iteration t,

v_{id}^{old} is the velocity in the previous iteration (t-1),

Step 5. Move the position of each particle i.

$x_{id}^{new} = x_{id}^{old} + v_{id}^{new}$,

where x_{id}^{new} is the new position in the iteration t,

x_{id}^{old} is the position in the iteration (t - 1).

Step 6. Check the feasibility of the solution and the number of iteration t.
1) The precedence is checked by *APM* and *DPM*.
2) The design alternative case is checked by *DCT*.
3) If ($t > T_{Number}$), then go to Step 7, else go to Step 2.

Step 7. Decode the best particle position and interpret the solution.

V. Implementation and Test Results

The presented models were implemented and tested by developing software on a personal computer with a 3.0 GHz CPU and 1 GB memory. Product A as illustrated in Fig. 3 was modeled and tested. Product A is a notebook computer with 16 main components. There are 4 proposed design alternative cases. In one of the design cases, the APG and DPG of product A are illustrated as shown in Fig. 4. The corresponding APM and DPM of product A are listed in section 3. The numerical values of the PSO parameters are tested with experiments using a Taguchi's orthogonal array to find the best combination of parameters of $T_{Number} = 300$, $P_{Size} = 40$, $w_i = (1.2, 0.8)$, and (c_1, c_2) = (2.0, 1.6).

Fig. 6 shows that the computation converges after 106 generations with a cost of 387.625 (unit: dollars) and a computer time of 2.922 (unit: seconds). As shown in Table II, the position matrix of the solution particle is decoded into an assembly sequence, a disassembly sequence, and the selection of a design case. The design case decision is determined by the values in the third row of the position matrix.

The values in the first and the second rows of the position matrix are used to generate the assembly and disassembly sequences. The ranked order represents the ordered sequence. As shown in Table II, The assembly sequence is C_4-C_{14}-C_{13}-C_{10}-C_8-C_7-C_{11}-C_{15}-C_3-C_5-C_2-C_0-C_6-C_{12}-C_1-C_9. In the second row, the disassembly sequence can be decoded as C_9-C_1-C_{12}-C_0-C_5-C_2-C_3-C_6-C_{15}-C_{11}-C_7-C_{14}-C_{13}-C_8-C_{10}-C_4. As observed from the presented method and the tested results, it shows that the developed model and algorithm present a feasible and efficient solution method.

VI. Conclusions

In this research, an integrated design evaluation and assembly and disassembly sequence planning model is presented to achieve the purpose of green design for assembly. In the design stage of a product life cycle, the components can be modeled differently to design the same product to fulfill the requirements. If different design cases are selected, the assembly and disassembly sequences can be different. By planning the activities of design, assembly, and disassembly in an integrated way, the total costs can be reduced to achieve the green purpose. A PSO method is developed for concurrently evaluating the design and optimizing the assembly and disassembly sequences. Given a design case, the information of constraints and costs can be modeled as assembly precedence graph, disassembly precedence graph, assembly precedence matrix, and disassembly precedence matrix. The related assembly and disassembly costs are formulated as the fitness function. The information of design alternative cases is modeled as the design case table. A new encoding scheme of the PSO method is developed for integrated design decision and assembly and disassembly sequencing. The test results show that the PSO method converges in a fast way to reach a minimized cost objective. It can be generally concluded that the developed model and the PSO method are feasible and efficient for solving the integrated design evaluation and assembly and disassembly sequence planning problem. Future research can be concerned with a detailed analysis of the relationships among design specifications, assembly operations, and disassembly operations. In addition, the PSO method can be improved to reduce computational time for practical and larger problems.

ACKNOWLEDGMENT

This research is funded by the National Science Council of Taiwan with project number NSC 100-2221-E-155-021-MY3.

REFERENCES

[1] A. C. Lin, and T. C. Chang, "An integrated approach to automated assembly planning for three-dimensional mechanical products," *International Journal of Production Research*, vol. 31, no. 5, pp. 1201-1227, May 1993.
[2] T. A. Abdullah, K. Popplewell, and C. J. Page, "A review of the support tools for the process of assembly method selection and assembly planning," *International Journal of Production Research*, vol. 41, no. 11, pp. 2391–2410, July 2003.
[3] H.Y. Lai, and C.T Huang, "A systematic approach for automatic assembly sequence plan generation," *International Journal of Advanced Manufacturing Technology*, vol. 24, pp. 752-763, November 2004.
[4] Y. M. Chen, and C. T. Lin, "A particle swarm optimization approach to optimization component placement in printed circuit board assembly," *International Journal of Advanced Manufacturing Technology*, vol. 35, no. 5-6, pp. 610-620, December 2007.
[5] Q. Su, "Computer aided geometric feasible assembly sequence planning and optimizing," *International Journal of Advanced Manufacturing Technology*, vol. 33, pp. 48-57, May 2007.
[6] T. Dong, R. Tong, and L. Zhang, "A knowledge-based approach to assembly sequence planning," *International Journal of Advanced Manufacturing Technology*, vol. 32, pp. 1232-1244, May 2007.
[7] Y.J. Tseng, J.Y. Chen, and F.Y. Huang, "A multi-plant assembly sequence planning model with integrated assembly sequence planning and plant assignment using GA," *International Journal of Advanced Manufacturing Technology* , vol. 48, no. 1-4, pp. 333-345, April 2010.
[8] S. Jin, W. Cai, X. Lai, and Z. Lin, "Design automation and optimization of assembly sequences for complex mechanical systems," *International Journal of Advanced Manufacturing Technology*, vol. 48, no. 9-12, pp. 1045-1059, June 2010.
[9] Y.-J Tseng, H.-T. Kao, and F-Y. Huang, "Integrated assembly and disassembly sequence planning using a GA approach," *International Journal of Production Research*, vol. 48, no. 20, pp. 5991–6013.
[10] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. 1995 IEEE Int. Conf. Neural Networks*, Piscataway, NJ, pp. 1942-1948.
[11] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *1997 Proc. Int. Conf. Systems, Man and Cybernetics*, Piscataway, NJ, pp. 4104-4109.
[12] A. Banks, J Vincent, and C. Anyakoha, "A review of particle swarm optimization. Part II: hybridization, combinatiorial, multicriteria and constrained optimization, indicative applications," *Natural Computing*, vol. 7, no. 1, pp. 109-124, March 2008.

Yuan-Jye Tseng is a professor in the Department of Industrial Engineering and Management at the Yuan Ze University, Taiwan. He received his M.S. and Ph.D. degrees in Industrial Engineering from the Pennsylvania State University, USA. His research interests include computer-aided design and manufacturing, assembly and disassembly planning, and green supply chain management.

Feng-Yi Huang received the MS degree in Industrial Engineering and Management at the Yuan Ze University and is now a Ph.D. candidate at the Department of Industrial Engineering and Management, Yuan Ze University.

Fang-Yu Yu received the MS degree in Industrial Engineering and Management at the Yuan Ze University.

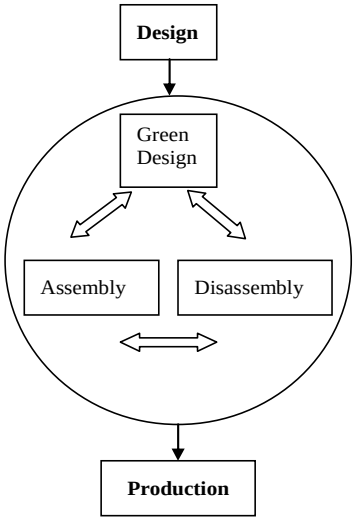

Fig. 1 A green design for assembly model by integrated evaluation of design and assembly and disassembly sequences

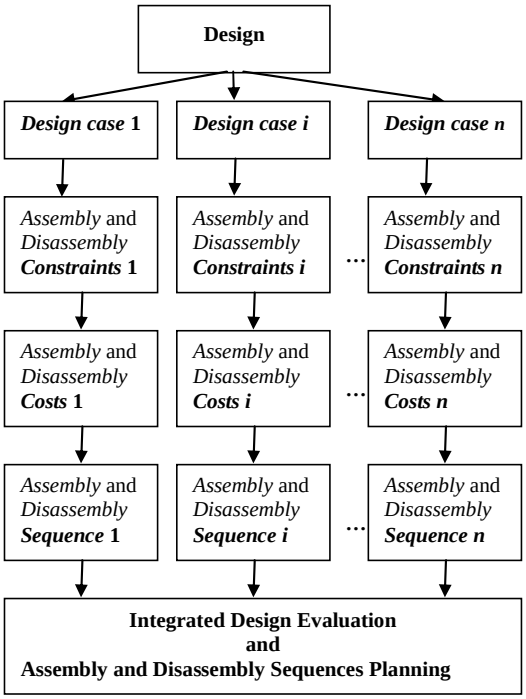

Fig. 2 The flow of the integrated design evaluation and assembly and disassembly sequence planning

Fig. 3 Product A is a notebook computer with 16 main components

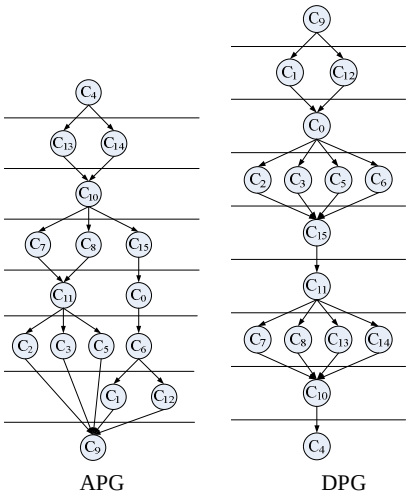

Fig. 4 The APG and DPG of example product A

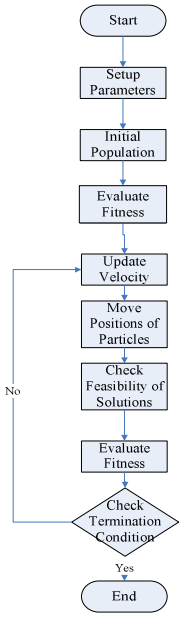

Fig. 5 The flowchart of the PSO method

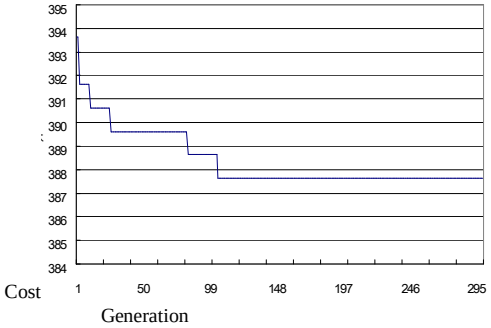

Fig. 6 The test result of the PSO evaluation for product A

TABLE I
THE FORMAT OF DESIGN CASE TABLE (DCT)

Design d_j / Component c_i	1	2	…j…	m
1	t_{11}	t_{12}		t_{1m}
2	t_{21}	t_{22}		t_{2m}
…				…
i	t_{i1}	t_{i2}	t_{ij}	T_{im}
…				…
n	t_{n1}	t_{n2}		t_{nm}

t_{ij} = 1 indicates that c_i is changed in design d_j,
t_{ij} = 0 indicates that c_i is not changed in design d_j.

TABLE II
THE SOLUTION OF THE INTEGRATED DESIGN EVALUATION AND ASSEMBLY AND DISASSEMBLY SEQUENCE PLANNING FOR PRODUCT A

Assembly sequence Order	Component C_i	Disassembly Sequence Order	Component C_i
1	C_4	1	C_9
2	C_{14}	2	C_1
3	C_{13}	3	C_{12}
4	C_{10}	4	C_0
5	C_8	5	C_5
6	C_7	6	C_2
7	C_{11}	7	C_3
8	C_{15}	8	C_6
9	C_3	9	C_{15}
10	C_5	10	C_{11}
11	C_2	11	C_7
12	C_0	12	C_{14}
13	C_6	13	C_3
14	C_{12}	14	C_8
15	C_1	15	C_{10}
16	C_9	16	C_4