2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

A Novel Optimal Assembly Algorithm for the Haptic Interface Application of a Virtual Maintenance System

Christiand and Jungwon Yoon

Abstract—**In a virtual environment, a virtual maintenance process can be used to simulate real-world maintenance, and the efficiency of the simulation depends mainly on the assembly/disassembly task sequence. During the simulation, the path planning of mechanical parts becomes an important factor since it affects the overall efficiency of the maintenance system in terms of saving energy and time. Therefore, planners must consider the path-planning factors under constraints such as obstacles and the initial/final positions of the parts, as well as the assembly sequence, for example, the number of gripper exchanges and direction changes. We propose a novel optimal assembly algorithm that considers the assembly sequence of mechanical parts and the path-planning factors for a virtual maintenance simulation system. The genetic algorithm is used to determine the optimal parts sequence to minimize the numbers of gripper exchanges and direction changes, as well as find a repulsive force radius using the potential field method to generate the shortest optimal distance for each part during the assembly operation. By applying the proposed algorithm to a virtual maintenance system, users can be haptically guided to the optimized assembly solution during mechanical parts assembly operations.**

I. Introduction

Maintenance processes deal with the assembly and disassembly (A/D) of thousands of parts. The level of difficulty of the A/D maintenance process increases with the number of parts, which causes various possible A/D sequences. An optimal A/D sequence can improve the maintenance efficiency by reducing the number of gripper exchanges and orientation changes during the process, or by minimizing the necessary time. The genetic algorithm (GA) has been applied to develop optimal A/D sequence algorithms since it provides a global optimal solution for complex nonlinear systems. Lazzerini and Marcelloni [1] evaluated an optimal assembly sequence based on three optimization criteria: minimizing the orientation changes of the product, minimizing the gripper replacements, and grouping technologically similar assembly operations. Li *et al*. [2] used the GA to search for a near-optimal disassembly sequence from an enormous search space that contained all possible disassembly sequences. In their study, the direction of each disassembly operation and the necessity of tool exchanges were considered in the optimization cost function because both processes result in the expenditure of disassembly time

To increase the efficiency of a real maintenance process, a maintenance preprocess procedure can be simulated in a virtual environment (VE). In this way, the problems of a given maintenance scheme can be predicted by the users (*i.e.*, maintenance operators), who may be able to suggest an alternative scheme. Haptic interfaces have been proposed for maintenance simulation systems to enhance the reality of the VE through physical interactions. Borro *et al*. [3] developed a haptic system for an aeronautical maintenance simulation by adding a haptic device to their system and ensuring that users' movements were the same as those that occur when testing a physical mock-up. A similar study was described by Savall *et al*. [4], while Kuang *et al*. [5] focused on building a virtual fixture assembly language (VFAL). A haptic guidance force can be applied to each mechanical part by combining haptic technology with a maintenance simulation to more actively utilize the haptic information for an A/D process [6]. A maintenance operator can then be trained to move parts from their initial positions to their final positions with respect to the assembly sequence. However, to the best of our knowledge, an assembly optimization algorithm that simultaneously considers both the assembly sequence and the part path trajectories did not exist.

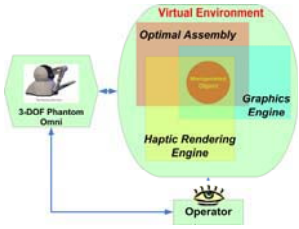
We propose a novel optimal assembly algorithm by considering the relationship between the path planning and assembly sequence, allowing users to utilize a haptic interface in a virtual A/D system. Since the proposed optimization algorithm generates the path trajectory of each part as well as the sequence of the procedure, the path information can be utilized to generate an active haptic force to guide users of a virtual maintenance simulation system during the operation. We introduce a virtual maintenance simulation system in Section II, and Section III describes the proposed optimization scheme for assembling mechanical parts. Section IV provides the optimized results and describes an assembly simulation, while Section V contains our conclusions and a discussion of future work.

II. An Intelligent Virtual Maintenance Simulation System For Aircraft Parts

Since the ability of maintenance operators performing repetitive maintenance tasks depends on their experience, they must be familiar with the basic skills of maintenance

Manuscript received September 14, 2007. This work was supported in part by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (KRF-2005-005-J09902) and was supported by 2nd stage BK21 (Brain Korea 21).

Christiand and Jungwon Yoon are with the Robotic and Intelligent System Laboratory, School of Mechanical and Aerospace Engineering, Gyeongsang National University, Jinju, South Korea (e-mail: tianize@yahoo.com, jwyoon@gnu.ac.kr).



(a)

(b)

Fig. 1. (a) The virtual maintenance system is consisted of a haptic device, haptic and graphic rendering engines, an optimal assembly algorithm. (b) an operator.

tasks, such as A/D of parts. Due to the advancements of virtual reality technologies, training in VEs in a manner similar to physical training will eliminate the need for a physical model and make it easier to change training schemes. The resources for VE training schemes can be obtained from the experience of experts, maintenance databases, or intelligent algorithms. Our intelligent virtual maintenance simulation system was built as a solution to enhance the benefits of virtual training.

The proposed system consists of a virtual environment, a haptic device, and a user, as shown in Fig. 1. The virtual environment includes assembled/disassembled parts (manipulated objects), a graphics engine, and a haptic rendering engine. The files for the assembled/disassembled parts were generated using the CATIA™ software package and converted to object files so that they could be read by the graphics engine. We used the CHAI3D application programming interface (API) [7] for the graphics and haptic rendering engines. The graphical user interface (GUI) was designed to interface between the user (or operator) and the virtual environment. The user can manipulate objects through a window that contains the virtual maintenance scheme. He/she can communicate with the virtual environment using Sensable's Phantom Omni, a three degree-of-freedom haptic device that provides six inputs (force and torque components) and three outputs (force components).

The operator can be guided by a haptic guidance force along a prescribed path. The prescribed path can be generated from the result of the optimal assembly algorithm (see Section III). The operator must ensure that the parts move always inside the path boundary to avoid collisions with obstacles. When the operator begins to move away from the path boundary, a haptic guidance force is generated to ensure that the operator's movement stays inside the path boundary. This enhances the efficiency of the maintenance process by training the maintenance operator to follow the desired path. We used the potential field method [8] for the path-planning algorithm. Repulsive and attractive forces were combined to create a path that avoids obstacles and brings the part to its final position. The operator must move parts from their initial positions to their final positions according to the assembly sequence of the parts, which includes information about the geometric relationship between parts and grippers, and the assembly direction (orientation). To generate both the optimal assembly sequence and path simultaneously, an optimal assembly algorithm was developed using a simple GA [9].

III. Optimization Scheme

A. Effective factors of the assembly optimization

To achieve an efficient maintenance process, two performance criteria were considered in the assembly scheme: the assembly sequence and the path of the mechanical parts. The optimization of the assembly sequence represents a problem of finding the part sequence that will bring about the most efficient maintenance process. To determine the optimal solution of the assembly sequence, the cost function must be defined as an evaluation index. From previous studies, the number of gripper exchanges and direction changes of mechanical parts are typically considered during assembly or disassembly processes [1,2,10]. Many gripper exchanges means that more time is needed to change from one gripper to another. Similarly, many direction or orientation changes requires more effort to perform the maintenance tasks. One of the optimization objectives is to find the optimal assembly sequence that has the least number of gripper exchanges and orientation changes. Pan *et al*. have shown that for both robot and human operator assembly processes, the number of reorientations in an assembly sequences has a significant impact on assembly time [11]. Therefore, in this paper, one of the optimization objectives is to find the optimal assembly sequence that has the least number of gripper exchanges and orientation changes.

Optimization of the path planning can influence the efficiency of the overall maintenance process. Obstacles and the locations of the parts can be considered as path-planning factors. In a maintenance process, the path trajectories that are generated from a formal path-planning analysis contribute greatly to the efficiency of a maintenance task since a long path consumes more travel time and energy. A short path is required to achieve the optimal assembly of mechanical parts. Therefore, the path-planning optimization formulization seeks an optimal setting for the path-planning variable. We selected the boundary value of the repulsive force (ρ_0) as our path-planning variable and implemented the potential field method [9] described in Section IIIb below. The correlations between the path planning and assembly sequence must also be considered. New assembly rules (see Section IIIc) were

proposed to resolve the difficulties that arise when both the path planning and the assembly sequence are considered during the assembly operation.

Finally, we combined the path-planning optimization with the assembly sequence optimization. The cost function components were the number of gripper exchanges (G), the number of assembly direction changes (O), and the actual distance of the path (D_{act}). The number of gripper exchanges and direction changes contributed to the assembly sequence optimization while the path distance of each part contributed to the path-planning optimization.

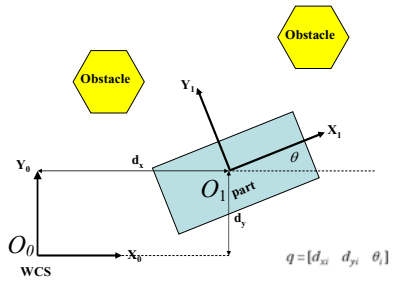


Fig. 2. Parts and obstacles are represented as a polygonal object on the workspace. Position of the object with respect to workspace coordinate is defined by configuration $q = [d_{xi}\ d_{yi}\ \Theta_i]$.

B. Path Planning of the Parts

The term "path planning" means finding a path trajectory for each part from its initial position to its final position. A part is represented as a polygonal object and its center is located on the center of the object boundary box (2-D object), as shown in Fig. 2. Even though 3-D parts will more closely approximate a real maintenance system, the framework for and efficiency of the proposed assembly optimization can be verified through the movement of 2-D parts. The axis-aligned boundary box (AABB) method was used to create the boundary box. The positions of a part and its boundary box in a mobile frame o_1 with respect to the reference frame o_0 can be represented as the transformed points of part vertexes using a 2-D homogenous transformation matrix. The configuration of the parts is represented by $q = [d_{xi}\ d_{yi}\ \Theta_i]$.

$$H = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & dx_i \\ \sin\theta_i & \cos\theta_i & dy_i \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where :

d_{xi}, d_{yi} = translation along the x-axis, translation along the y-axis of the workspace coordinate.
Θ_i = angle between x-axis of workspace coordinate and x-axis of local/frame part coordinate.

The potential field method uses simple design variables and is adequate for real-time applications. This method is composed of attractive and repulsive forces: the attractive force pulls the part toward its final position while the repulsive force repels the part from a collision with obstacles. Due to its simple structure, we can render a potential force at each sampling time for every path point. The repulsive and attractive forces are represented as follows:

if $\rho(o_i(q)) \le \rho_0$:

$$F_{rep,i}(q) = -\eta_i\left(\frac{1}{\rho(o_i(q))} - \frac{1}{\rho_0}\right)\frac{1}{\rho^2(o_i(q))}\nabla\rho(o_i(q)) \quad (2)$$

if $\rho(o_i(q)) > \rho_0$:

$$F_{rep,i}(q) = 0 \quad (3)$$

where :

q = configuration of parts.
F_{rep} = repulsive force.
ρ_0 = repulsive force radius.
$o_i(q)$ = point on the workspace.
η_i = scale factor.
$F_{rep,i}$ = repulsive force on i-th part.
i = index of i-th part.

If $\|o_i(q) - o_i(q_f)\| \le d$:

$$F_{att,i}(q) = -\zeta_i(o_i(q) - o_i(q_f)) \quad (4)$$

If $\|o_i(q) - o_i(q_f)\| > d$:

$$F_{att,i}(q) = -d\zeta_i\frac{(o_i(q) - o_i(q_f))}{\|(o_i(q) - o_i(q_f))\|} \quad (5)$$

Where :

q_f = final configuration of part at final position.
d = distance from final position to current position
ζ = scale factor.
F_{att} = attractive force.

The repulsive force has a conditional value $\rho(o_i(q))$, which is the distance from the center of a part to the center of an obstacle. The summation of the attractive and repulsive forces gives the direction for movement as a normalized vector as follows:

$$\vec{f} = \frac{F_{att} + F_{rep}}{\|F_{att} + F_{rep}\|} \cdot S \quad (6)$$

f = applied force to move the parts.

S = step between path points.

At every path point o_i, the force will generate a direction for the next path point toward the final position. The maintenance operator will follow this path from the part's initial position to its final position.

The assignment of repulsive forces on obstacles must be handled carefully. The value of ρ_0 must not be too large or too small to achieve an optimal path. The value of ρ_0 in Eqs. (1) and (2) will affect the overall results of the path-planning algorithm. The value of ρ_0 must be predetermined to achieve an efficient maintenance process by optimization.

C. Assembly Rules, Assembly Direction, and Gripper Selection.

We proposed new assembly rules to combine the relationship between the assembly sequence and the path planning. The rules regulate the parts movement when entering the final position area.

1st rule: "*Parts has reached its final position will become an obstacle for the next parts.*"

This rule ensures that an arriving part will not collide with the next part when it enters the final position area. Figure 3(a) shows a part entering its final position area (goal position) as an example. Part 2 is one that arrived earlier and has already reached its final position; it exerts a repulsive force. Part 3 is just reaching its final position, and does so without colliding. Since the shape of the accumulated repulsive forces in the final position area depends on the parts that have already arrived, a different assembly sequence will create a different path for each part. For example, consider the case in which the configurations of assembled parts have two possible assembly sequences, shown in Fig. 3(b): A = [1 2 4 3 5 6] and B = [2 3 4 1 5 6]. If Part 4 is traveling to its final position (black dot) from the right side, its path will be directly affected by the accumulated repulsive forces that are generated based on the other parts that have already arrived. Since the path depends on the assembly sequence A(1-2) or B(2-3), the final path for sequence A or B will be different depending on which parts have already arrived .

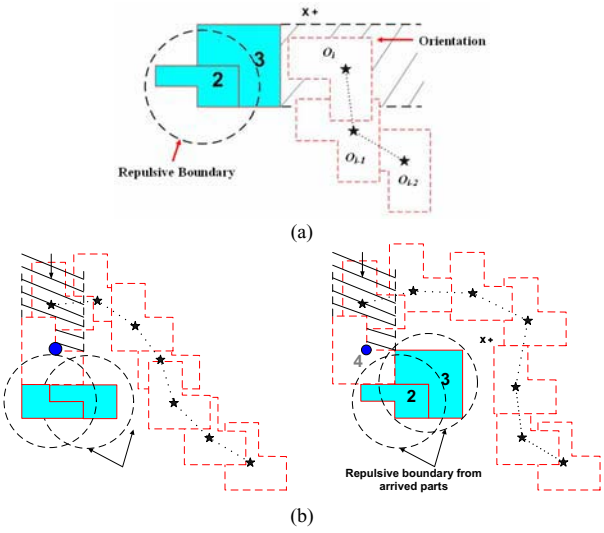


Fig. 3. (a) Schematic picture to illustrate the idea when the parts enter its final position area (goal position). (b) Two different part sequence affect the accumulated repulsive force on final position area.

2nd rule: *"After selecting the possible assembly direction the part must move in that direction when assembled to avoid interference between parts."*

TABLE I
CONNECTION MATRIX

X+

Parts	1	2	3	4	5	6
1	0	1	1	0	0	0
2	0	0	1	0	0	0
3	0	0	0	0	0	0
4	0	0	1	0	1	0
5	0	0	0	0	0	0
6	0	0	0	0	1	0

Y+

Parts	1	2	3	4	5	6
1	0	1	0	1	1	1
2	0	0	1	1	1	0
3	0	0	0	0	1	0
4	0	0	0	0	1	1
5	0	0	0	0	0	0
6	0	0	0	0	0	0

X-

Parts	1	2	3	4	5	6
1	0	0	0	0	0	0
2	1	0	0	0	0	0
3	1	1	0	1	0	0
4	0	0	0	0	0	0
5	0	0	0	1	0	1
6	0	0	0	0	0	0

Y-

Parts	1	2	3	4	5	6
1	0	0	0	0	0	0
2	1	0	0	0	0	0
3	0	1	0	0	0	0
4	1	1	0	0	0	0
5	1	1	1	1	0	0
6	1	0	0	1	0	0

In an assembly process, each part must have a predefined assembly direction to prevent interference between parts. A connection matrix is used to select a feasible assembly direction by observing the relationship between successive parts. Table 1 shows the geometric relationships between parts for each direction (X+, Y+, X–, and Y–) with respect to the local reference frame of final positions. In the table, 0 indicates that the part can be assembled in the given direction, while 1 indicates that the part cannot be assembled in the given direction because other parts will interfere with the movement. For example, if the part sequence is [2 3 4 6 5 1], Part 3 cannot reach its final position in the Y– direction since it is blocked by Part 2 (cell [3,2] of the Y– connection matrix contains 1). Another assembly direction must be found for Part 3. The proposed direction-finding algorithm only considers the directions of the principal axes of the part, which is sufficient for this case. In the numerical calculations, the assembly directions were represented as integer values: X+ = 1, Y+ = 2, X– = 3, and Y– = 4.

3rd rule: *"If the path point (o_i) is perpendicular to the face of the part in a given assembly direction during the path searching, all repulsive forces that are generated from existing parts will disappear automatically."*

The gripper used to grasp and move a part from one position to another is also an important factor in a real assembly process. We assumed that four types of grippers were used. A gripper was assigned for each part and indicated as *G1, G2, G3,* and *G4*. In some cases, it was possible to use the same gripper for several parts; this possibility must be considered in the optimization. The gripper used for each part was determined by checking a gripper table with respect to the assembly sequence. The grippers are listed in Table 2. Grippers in column P1 are preferable to those in column P2. A planner will use a gripper in column P1 if the previous gripper has no similarity or is unsuitable for the current part. For example, if the sequence of parts is [6 2 3 4 5 1], the G2 gripper from column P1 will be used for Part 6. Then we have two gripper options for Part 2: G1 and G2. Since G2 was used for the previous part, the same gripper will be used for Part 2. If a single gripper cannot be used for both Parts 6 and 2, the G1 gripper will be selected from column P1 for Part 2.

TABLE II
GRIPPER TABLE

Part	P1	P2
1	G1	G2
2	G1	G2
3	G3	
4	G3	
5	G3	G4
6	G2	

D. Optimization Assembly Algorithm by Considering Assembly Sequence and Path Planning

Initial populations were randomly generated at the beginning of the optimization process. The population consisted of 50 chromosomes, and each chromosome had two main components: the sequence of parts and a ρ_0 array. The sequence was encoded as a series of integer values, whereas the ρ_0 array was encoded as binary values. These two components were the optimization design parameters (optimized variables). The series of ρ_0 were ordered in sequence. A cost value was calculated for each chromosome in the population. The inputs for the cost value were the actual path distance of each part (D_{act}), the number of orientation changes (O), and the number of gripper exchanges (G):

$$Cost = w_1\left(1-\frac{O}{n-1}\right)+w_2\left(1-\frac{G}{n-1}\right)+w_3\left(\frac{\sum_{i=1}^{n}\frac{(D_{ref})_i}{(D_{act})_i}}{n}\right) \quad (7)$$

w_1, w_2, w_3 = weighting factors
n = numbers of parts

The actual path distance (D_{act}) is a summation of the single distances from one path point o_i to another to reach the final position. D_{ref} is the shortest distance from the initial position to the final position. The comparative value between D_{act} and D_{ref} gives the efficiency of the path. The path is efficient if the comparative value is close to 1 and inefficient if the comparative value is close to 0. For each part, a path trajectory was generated using the path-planning algorithm under the guidelines of the assembly rules. The number of orientation changes was based on how many times the assembly directions changed in one sequence, and was calculated from the connection matrix given in Table 1. The number of gripper exchanges was based on how many times a planner changed the gripper in one sequence and was calculated from the data given in Table 2. The maximum value of the cost components was 1. Weighting factors were assigned to the cost value components so that one component became more important than others. The maximum summation of the cost value components was set to 1.

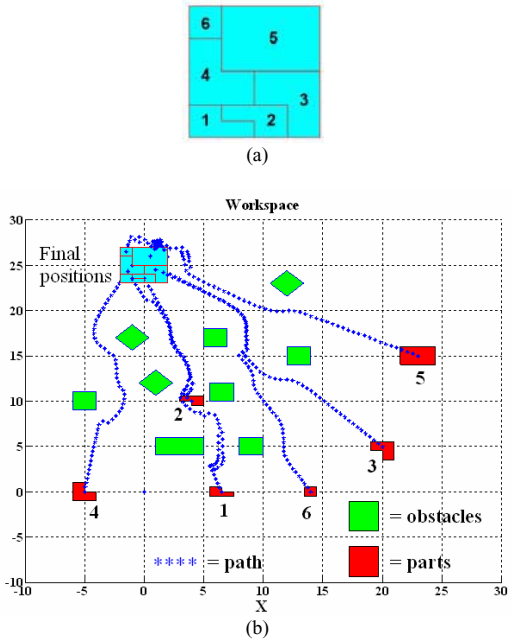


Fig.4. (a) An assembled part consists of several parts that are represented as polygonal objects. (b) The graph shows an optimal solution, which is resulted from the genetic optimization scheme after 90 generations. The optimal assembly sequence is [5 6 2 1 3 4] with the ρ_0 = [3.8041 3.5231 3.3541 4.5858 4.2113 3.2912].

IV. SIMULATIONS AND OPTIMAL RESULTS

An assembly simulation using the proposed algorithm was performed for the parts of a 2-D object shown in Fig. 4(a). Six parts were required to reach their final positions. The initial positions of the parts were at different locations in a 40 × 40 unit square workspace (see Fig. 4(b)). The crossover probability was 0.8 and the mutation probability was 0.01 for purposes of the GA. The value of weighting factors were w_1 = 0.25, w_2 = 0.25, and w_3 = 0.5. While ρ_0 was being optimized, the other variables in the repulsive force and attractive force

equations of the potential field method remained constant.

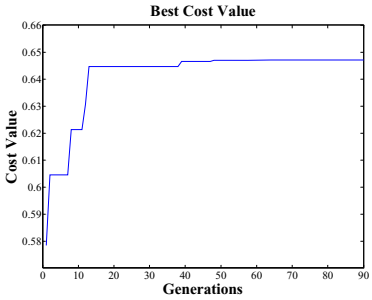


Fig.5. Best cost value graph. The graph shows the best cost values of population in every generation. It was taken after 90 generation.

An average of 14.4 s was required to complete the computation process of one chromosome to generate the path (path planning) for each part (six parts) and run the genetic algorithm itself. After computing 90 generations, the best cost value of each generation was determined, as shown in Fig. 5. All parts reached their final positions with respect to the assembly sequence by traveling in their given direction without encountering any local path minima, as shown in Fig. 4(b). Table 3 shows the comparison of varying ρ_0 value when the sequence is already optimal. The appropriate value of ρ_0 will give a short distance contribution. Even though the sequences are same, but the setting of ρ_0 gave different values to the total actual distance.

TABLE III
MEASURING FOR ACTUAL DISTANCE

Sequence	ρ_{05}	ρ_{06}	ρ_{02}	ρ_{01}	ρ_{03}	ρ_{04}	D_{act}
5,6,2,1,3,4	4.53	4.19	4.70	4.75	3.65	4.54	779.13
5,6,2,1,3,4	3.80	3.52	3.35	4.58	4.21	3.29	204.59
5,6,2,1,3,4	4.19	4.31	4.48	4.20	3.65	2.94	209.60
5,6,2,1,3,4	4.19	3.63	3.35	4.82	3.65	3.77	1660.1

Even though the suggested optimization scheme was verified with simple 2-D parts, it can be extended to more complex 3-D parts. For the algorithm extension, several aspects should be considered. The shape of the repulsive boundary area can be changed from a circle into a sphere where ρ_0 is the radius of the sphere. Therefore, ρ_0 still can be an optimized variable for the optimization process. Also, the assembly direction finding process can conform to the spatial (3-D) problems. The assembly directions in 2-D cases are limited to four principal axes, whereas it will be limited to six principal axes of the mechanical parts in 3-D cases. The simple connection matrix can be easily extended to six principal axes. Otherwise, more advanced methods can be considered such as NDBG (Non-Directional Blocking Graph) that was introduced by Wilson [12]. By using the NDBG, the feasible assembly directions will not be limited to only principal axes. However, it will cause more computation time than our simple method. The appropriate decision for the assembly direction finding can be made by comparing the ratio of computation time and effectiveness from both methods.

V. CONCLUSION

The proposed algorithm can help a planner to approach an optimal assembly sequence more effectively, since the algorithm does not only cover the optimal assembly sequence but also cover a path planning factor (ρ_0) by the application of GA (genetic algorithm). Also, assembly rules are suggested to solve the problems that are happening during combining an assembly sequence and a path planning process. The optimization results with six parts of 2D showed that the parts assembly could be successfully performed by minimizing the desired movements of the parts with an optimal sequence. Moreover, the obtained results can be applied to a haptic application for active assembly training of a user. For future works, the extensive user studies will be performed to show the effectiveness of a haptic guidance with the suggested optimal assembly algorithm with extensions of the parts to 3D.

REFERENCES

[1] B. Lazzerini, F. Marcelloni, "A Genetic Algorithm for generating Optimal Assembly Plans", *Artificial Intelligence in Engineering*, Vol. 14, pp. 319-329, 2000

[2] J.R Li, Li Pheng Khoo, Shu Beng Tor, "An object-oriented intelligent disassembly sequence planner for maintenance", *Computers in Industry*, vol.56, pp. 699-718, Elsevier, 2005.

[3] D. Borro, J. Savall, A. Amundarain, J.J Gil, A. G. Alonso, L. Matey, " A Large Haptic Device for Aircraft Engine Maintainability ", *IEEE Computer Graphics and Applications*, vol. 24, no. 6, pp. 70-74, Nov/Dec 2004.

[4] J. Savall, D. Borro, J.J Gil, L. Matey, "Description of a Haptic System for Virtual Maintainability in Aeronautics", *IEEE International Conference on Intelligent Robots and Systems*, Laussane, Switzerland, October 2002.

[5] A.B. Kuang, S. Payandeh, B. Zheng, F. Henigman, Christine L. MacKenzie, "Assembling Virtual Fixtures for Guidance in Training Environments", haptics, pp. 367-374, *12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Chicago, United States, 2004.

[6] Christiand and J. Yoon, "Intelligent Assembly/Disassembly System with a Haptic Device for Aircraft Parts Maintenance", *International Conference on Computational Science*, Beijing, China. 27-31 May 2007.

[7] F. Conti , F. Barbagli, D. Morris, C. Sewell, " CHAI: An Open-Source Library for the Rapid Development of Haptic Scenes", *IEEE World Haptics*, Pisa, Italy, March 2005.

[8] M. W. Spong, S. Hutchinson, M. Vidyasagar, "Robot Modeling and Control ", John Willey and Sons, United States, 2006.

[9] J. H. Holland, "Adaptation in Natural and Artificial Systems", MIT Press, Cambridge, United States, 1992.

[10] L . M. Galantucci; G. Percoco & R. Spina, "Assembly and Disassembly Planning by using Fuzzy Logic & Genetic Algorithms", *International Journal of Advanced Robotic Systems*, volume 1 Number, 2004.

[11] C. Pan and S. Smith, "Case study: the impact of assembly reorientations on assembly time", *International Journal of Production Research*, Vol. 44 issue 21, pp. 4569-4585, 2006.

[12] R. H. Wilson, "On Geometric Assembly Planning", *PhD Dissertation*, Computer Science Department of Stanford University, USA, 1992.