www.cadanda.com
Computer-Aided Design
AND APPLICATIONS

Optimal Design of Component Layout and Fastening Methods for the Facilitation of Reuse and Recycle

Masakazu Kobayashi[1], Hajime Horiuchi[2] and Masatake Higashi[3]

[1]Toyota Technological Institute, kobayashi@toyota-ti.ac.jp
[2]Toyota Technological Institute, sd12440@toyota-ti.ac.jp
[3]Toyota Technological Institute, higashi@toyota-ti.ac.jp

ABSTRACT

Due to rise of environmental awareness and enactment of legislation in recent years, products that reach their end-of-life need to be collected, disassembled and reused/recycled. However, since it is impractical to reuse/recycle every component that makes up a product from a cost effective standpoint, only high-value components are reused/recycled and the rest of components is discarded. Therefore, there is a need to design a product which high-value components can be removed with less disassembly cost and work for the facilitation of reuse and recycle. In this research, design of component layout and fastening methods inside a product during conceptual design phase is focused on and a new method of optimizing them is developed. The proposed method consists of layout optimization and fastening method optimization and explores the optimal component layout and fastening methods which high-value components can be removed with minimum effort by executing two optimizations cooperatively.

Keywords: computer aided design, design for environment, design for disassembly, layout optimization, fastening method optimization.

1. INTRODUCTION

Due to rise of environmental awareness and enactment of legislation in recent years, products that reach their end-of-life need to be collected, disassembled and reused/recycled. However, since it is impractical to reuse/recycle every component that makes up a product from a cost effective standpoint, only high-value components are reused/recycled and the rest of components is discarded. Therefore, there is a need to design a product in which high-value components can be removed with less disassembly cost and work for the facilitation of reuse and recycle.

For years, many researches have been done to make products to be disassembled more efficiently. Design for Disassembly (DfD) [1,12] is the guideline to design products to be easily disassembled for maintenance, repair, recovery and reuse of components/materials. Disassembly sequence planning [3, 5, 8] is the method to obtain the optimal disassembly sequence which a product is disassembled with minimum cost and work. Layout optimization considering disassemblability [10,11] is the method to explore the layout which components of a product can be disassembled with less cost and work. However, compared to the researches of DfD and disassembly sequence planning, layout optimization considering disassemblability has not been researched sufficiently. In addition, layout of components comprising the product is roughly decided during conceptual design phase, so the decision during the conceptual design phase has a major impact on the disassemblability of the product. Therefore, in order to obtain the disassemblable layout, component layout needs to be designed or optimized by evaluating disassemblability during conceptual design phase. Fastening methods are another factor that affects disassemblability of a product, various researches concerning design of fastening methods have also been done [2,6,7,9]. In addition, the difficulty of removing fasteners is affected by the workspace around the fasteners and the amount of the workspace depends on the component layout and the disassembly sequence. Therefore, to design more disassemblable product, component layout, disassembly sequence and fastener methods need to be designed or optimized simultaneously.

Computer-Aided Design & Applications, 12(5), 2015, 537–545, http://dx.doi.org/10.1080/16864360.2015.1014731

Taylor & Francis
Taylor & Francis Group

However, the method of optimizing them has not been developed.

In this research, to design a disassemblable product for the facilitation of reuse and recycle, a new method of optimizing component layout and fastening methods inside a product is developed. The proposed method consists of layout optimization and fastening method optimization and explores the optimal component layout and fastening methods in which high-value components can be removed with minimum effort by executing two optimization methods cooperatively. As for disassembly sequence planning, a new rule based on the value of components is introduced.

The rest of this paper is organized as follows. Section 2 describes the details of the proposed method. In Section 2, the overview of the proposed method is described and fitness function of the proposed method is defined. And then, details of layout optimization and fastening method optimization are explained. Section 3 describes the case study. In the case study, the proposed method is applied to the design of internal devices of a laptop computer. The results are compared with the laptop actually sold. Finally, Section 4 summarizes the results of this paper.

2. COMPONENT LAYOUT & FASTENING METHODS OPTIMIZATION

2.1. Problem Definition

- Component shape is represented by rectangular box and three-dimensional layout of components is represented by sequence triple [13].
- Each component has the value of 2R (reuse/recycle) and maintenance. The former means how much the component is worth recycling or reusing whereas the latter means how much maintenance the component requires. If the component requires frequent maintenance, the component should be easily removed from the product. The component of high 2R or maintenance value is simply named "high-value component". It is assumed that only high-value components are reused/recycled or require maintenance.
- Components can be removed in only one direction. One component can be removed at a time.

2.2. Overview of the Proposed Method

The purpose of the proposed method is to design the component layout and the fastening methods inside a product which high-value components can be removed from a product with minimum effort. Fig. 1. shows the overview of the proposed method.

The proposed method consists of layout optimization and fastening method optimization. Layout optimization is the main part of the proposed method and executed just one time. Layout optimization explores the optimal component layout according to the fitness function as described below. Fastening method optimization is applied to each proposal of component layout generated during layout optimization to obtain the optimal fastening methods that are removed with minimum time and the removing time is returned to layout optimization and used as a part of the fitness function of layout optimization. To explore the optimal component layout and fastening methods, genetic algorithm is used. Fitness function of layout optimization is defined by the below equation.

$$\text{Maximize } F = f - T \tag{2.1}$$

Where f is the disassemblability of the component layout and T is the time required to remove fasteners of high-value components. f shows how easily high-value components can be removed from the product. f depends on the positional relationships between components but not fastening methods. On the other

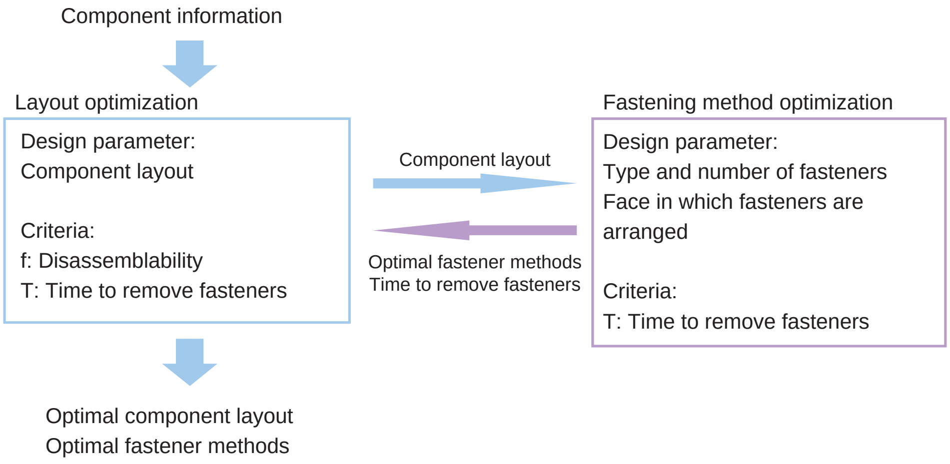


Fig. 1: Overview of the proposed method.

hand, since T depends on both the component layout and fastening methods, minimum T needs to be explored by executing fastening method optimization for each proposal of component layout.

In the proposed method, sequence triple is used to represent 3D component layout. Sequence triple represents 3D relative positions of rectangular boxes by using three rectangle name sequences. Three rectangle name sequences indicate the order of rectangular boxes in three orthogonal axes arranged on diagonal of x, y and z axes. Therefore, three component name sequences are handled as design variables of the GA for component layout optimization. Fitness function of the GA is F defined in Eqn. (2.3). As for constraint conditions of the GA, constraints concerning the layout of components, size and volume constraints and thermal constraint are handled. Their details are described in section 2.5. As for the algorithm of the GA, special crossover and mutation operator are required to use sequence triple representation. Their details are explained in the reference [13].

2.3. Evaluation of Component Layout

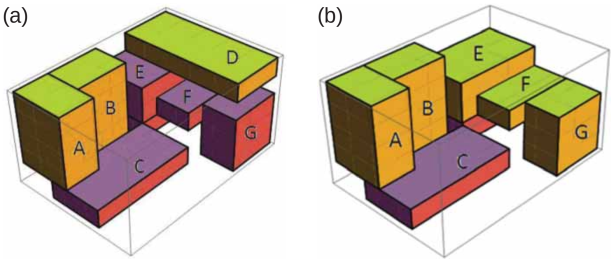
To evaluate the disassemblability of the component layout from the viewpoint of reuse and recycle, "disassembly sequence" and "disassembly sequence depth" are introduced. Disassembly sequence is first obtained and then disassembly sequence depth is obtained based on the disassembly sequence. Disassemblability of the component layout is defined by the disassembly sequence depth and the value of components.

2.3.1. Disassembly sequence

Disassembly sequence is the sequence of components to be removed from a product. Since the purpose of the proposed method is to obtain the component layout which high-value components can be removed from a product with minimum effort for the facilitation of reuse and recycle, the basic rule for obtaining disassembly sequence is introduced: "If more than one component can be removed at the same time, the component with higher value is removed preferentially". According to this rule, disassembly sequence is uniquely obtained by repeating the below two steps.

Step 1: List every component that can be removed at this time.
Step 2: Remove the most valuable component among the list. Go back to Step1.

Fig. 2, Tab. 1 and Tab. 2 show an example of obtaining disassembly sequence. In this example, it is assumed that components are removed in upper direction. Tab. 1 shows the component value. The component value is equal to the 2R (Reuse/recycle) value plus the maintenance value of the component explained in section 2.3.3. Fig. 2(a) shows the initial component layout. In the case of this figure, 3 components named A, B and D can be removed. Since component D is most valuable among them as shown in Tab. 1., component D is removed at the start. Fig. 2(b) shows the component layout after component D is removed. In the case of this figure, components A, B, E, F and G can be removed. Since component F is most valuable among them, component F is then removed. By repeating the above tasks, the entire disassembly sequence is obtained. The entire disassembly sequence of the example is D-F-A-G-C-B-E-H.



Initial layout

2nd layout

Fig. 2: Flow of obtaining disassembly sequence.

A	B	C	D	E	F	G	H
6	3	4	9	2	8	5	10

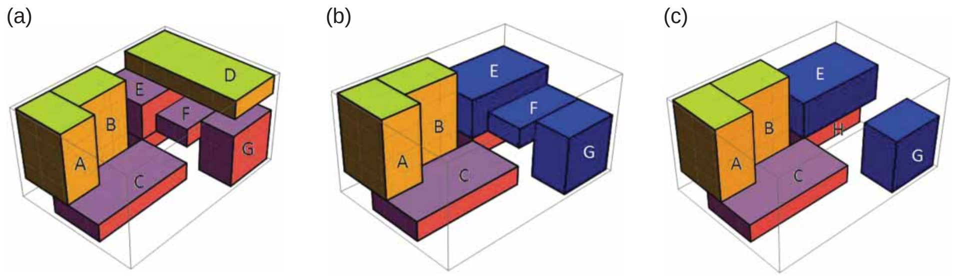
Tab. 1: Component value.

No.	Disassemblable components	Disassembly sequence
1	A, B, D	D
2	A, B, E, F, G	F
3	A, B, E, G	A
.	.	.
.	.	.

Tab. 2: Flow of obtaining disassembly sequence.

2.3.2. Disassembly sequence depth

Disassembly sequence depth shows the number of steps required for each component to become removable from a product. Depth of the components that can be removed from the beginning is called "Level 1". Depth of the components that become removable after removing 1 component is called "Level 2". So, depth of the components that become removable after removing n components is called "Level $n+1$". If



Initial layout 2nd layout 3rd layout

Fig. 3: Flow of obtaining disassembly sequence depth.

no new component becomes removable after removing n components, “Level n + 1” becomes empty set. Disassembly sequence depth is obtained by repeating the below four steps.

Step 1: List every component that can be removed at this time.
Step 2: Remove the components already listed in the lower level from the list.
Step 3: Set the components remained on the list to the current level.
Step 4: According to the disassembly sequence, remove the next component. Go back to Step1.

Fig. 3. and Tab. 3. show as an example of obtaining disassembly sequence depth. Since this example is based on the example described in section 2.3.1, the disassembly sequence is D-F-A-G-C-B-E-H. In this example, components A, B and D can be removed at the beginning, as show in Fig. 3(a), so their depth is set to Level 1. Then, according to disassembly sequence, component D is removed. Next, components A, B, E, F and G become removable, as shown in Fig. 3(b). But, components A and B are already set to Level 1, so only components E, F and G is set to Level 2. Then, component F is removed. Next, components A, B, E and G become removable, as shown in Fig. 3(c). But, these components are already set to Level 1 or 2, so no component is set to Level 3.

Level of DSD	Component	Disassembly sequence
Level1	A, B, D	D
Level2	E, F, G	F
Level3	0	A
Level4	C	G
Level5	0	C
Level6	0	B
Level7	0	E
Level8	H	H

Tab. 3: Flow of obtaining disassembly sequence depth (DSD).

2.3.3. *Disassemblability of component layout*

To explore the component layout which high-value components can be removed with minimum effort, disassemblability of component layout f is defined by the disassembly sequence depth and the value of components, as described in Eqn. (2.2).

$$f = \sum_{i=1}^{n} \{(D_{\max} - D_i) \times (V_{r,i} + V_{m,i})\} \tag{2.2}$$

Where $D_{\max}$ is the maximum level of disassembly sequence depth, D_i is the level of disassembly sequence depth of component i, $V_{r,i}$ is the 2R (recycle/reuse) value of component i, $V_{m,i}$ is the maintenance value of component i, and n is the number of all components. 2R value and maintenance value of each component is rated on a scale of 1 to 10. 2R value of the component is based on the market value when the component is reused/recycled. Specifically, 2R value $V_{r,i}$ is based on the ratio of the market value of component i divided by the market value of the highest-value component. Maintenance value $V_{m,i}$ is based on the ratio of the life-span of the longest life component divided by the life-span of component i. According to this definition, maintenance value of a short life component becomes high. This means that since such component requires frequent maintenances, it should be placed where it can be easily removed from the product.

2.4. **Fastening Method Optimization**

The purpose of fastening method optimization is to design the fastening methods that can be removed with minimum time. As described in Section 2.2, fastening method optimization is applied to each proposal of component layout generated during layout optimization and removing time T is returned to layout optimization. In fastening method optimization, three design parameters are considered: The face in which fasteners are arranged (More than one face can be selected), the type of fasteners and the number of fasteners. Removing time T is calculated by the below equation and handled as the fitness function of fastening method optimization.

$$\text{Minimize } T = \sum_{i=1}^{n} \{(T_{fastener,i} \times k_i + T_{tool,i}) \times G_i\} \quad (2.3)$$

Where $T_{fastener,i}$ is the basic time required to remove a fastener used in component i in the case where no obstacles are placed around the fastener, $T_{tool,i}$ is the basic time required to prepare the tools to remove fasteners from component i, k_i is the number of fasteners used in component i, G_i is the difficulty of removing fasteners from component i and n is the number of high-value components. Different from Eqn. (2.2), n is the number of high-value components, not the number of all components, and T is calculated by evaluating only high-value components. A designer configures high-value components and n based on the 2R value and maintenance value of components. $T_{fastener,i}$ and $T_{tool,i}$ are basic time and only depend on the type of a fastener and tools. The value of $T_{fastener}$ and T_{tool} is cited from the references [2,7,9]. If the same tool is continuously used for more than one components, there is no need to change a tool. Therefore, T_{tool} of second and later components is set to 0. G_i is based on the workspace around component i at the time of removing. The minimum workspace required to remove a fastener undisturbedly within the basic time $T_{fastener}$ is configured for each tool. If the actual workspace around component i at the time of removing is larger than the minimum workspace, G_i is set to 1. If not, G_i is based on the ratio of the minimum workspace divided by the actual workspace around component i. Workspace around a component at the time of removing varies according to the component layout and the disassembly sequence, so it needs to be evaluated for each proposal of component layout during layout optimization. To explore the optimal fastening method with minimum T, genetic algorithm is used.

Design variables of the GA for fastening method optimization are the face in which fasteners are arranged (More than one face can be selected) and the type of fasteners for each component. The number of fasteners required to fix the component is calculated as the required fastener strength of the component divided by the fastener strength of the selected fastener. Required fastener strength is one of constraint conditions and configured for each component and fastener strength is configured for each type of fastener. Fitness function of the GA is the removing time T defined in Eqn. (2.3). As for constraint condition, assemblability constraint is considered. Required assemblability is configured for each component and the assemblability of the component should be smaller than the required assemblability. The assemblability of the component is calculated as the assemblability of the selected fastener multiplied by the number of fasteners used in the component. If any of components does not satisfy the constraint, the fitness T receives a penalty according to the number of unsatisfied components. As for the algorithm, traditional GA is used.

2.5. Constraint Conditions

To design practical component layout and fastener methods, the following constraint conditions are configured. If a design proposal does not satisfy the below constraints conditions except connected constraints, its fitness receives a penalty for each unsatisfied constraint condition. Since the value of the penalty affects diversity and convergency of the GA optimization process, the value needs to be configured by trial and error for each optimization problem. If a design proposal does not satisfy connected constraints, its fitness is set to 0.

(1) Constraints concerning the layout of components

In practical layout design, some components need to be arranged on the specified position inside the products or contacted each other in order to perform their function appropriately. Therefore, positional, adjacent and connected constraints are considered during layout optimization. As for positional constraint, specified components need to be arranged on the specified position inside the product. As for adjacent constraint, specified components need to be contacted each other. Connected constraint is the special case of adjacent constraint. Some adjacent components need to be strongly connected to perform their function. Such components are handled as a single component and removed at a same time like a single component when obtaining disassembly sequence.

(2) Size and volume constraint

Since size and volume of a product is also important, their maximum acceptable values are handled as constraint conditions during layout optimization.

(3) Thermal constraint (Constraint concerning internal temperature)

For some products including heat generating components such as a personal computer, temperature distribution inside the product needs to be considered in order to guarantee the product's performance and lifetime and to protect a user. Therefore, internal temperature distribution is evaluated using thermal network method [4] and allowable temperature of each component is handled as a constraint condition during layout optimization.

(4) Constraint concerning fastening strength

If the time to remove fasteners is minimized without any constraint in fastening method optimization, no fastener is the best solution. However this solution is not practical. In a practical design, required fastening strength is configured for each component to guarantee that the product functions well. Therefore required fastening strength for each component is configured and handled as a constraint condition during fastening method optimization.

(5) Assemblability constraint

In practical product design, in addition to disassemblability, assemblability is also important. Therefore, required assemblability of each component is configured and handled as a constraint condition during fastening method optimization.

3. CASE STUDY

To test the effectiveness of the proposed method, the proposed method is applied to design of internal devices of a laptop computer. “Internal devices” means that input devices, a display and an enclosure are not included.

3.1. Details of the Case Study

Tab. 4 is the list of components. The size of components is based the laptop actually sold in 2007. This laptop is also used to compare with the optimal results in the next section. Since the 2R and maintenance value of components written in red in Tab. 4 are high, they are considered as high-value components. Tab. 5 shows the list of fastening methods. The value of $T_{fastener}$ and T_{tool} is cited from the references [2,7,9]. Since the component fixed by adhesion can not be removed with usual way, $T_{fastener}$ and T_{tool} are not set. Using these components, the laptop with 15 inch display is designed. Size constraints are 23 cm length, 30 cm width and 3 cm height. Volume constraint is 2100 cm^3.

As for positional constraints, optical drive, speaker, base 1&2, USB port 1&2, connector socket 1&2&3 and cooling fan need to be placed in contact with the enclosure. As for adjacent constraints, cooling fan needs to be contacted with one of three motherboards. As for connected constraints, motherboard 1&2&3 and base 1&2 need to be connected

No.	Component	2R value	Maintenance value	Required fastener strength	Required assemblability	Allowable temperature (deg C)	Heat generation (W)	Color
1	HDD	7	8.8	9	20	60	10	Gray
2	Optical drive	8	7.6	10	20	60	-	Pink
3	Battery	5	10	5	10	70	15	Blue
4	Speaker	3.5	5	4	20	60	-	Purple
5	Motherboard 1	10	7.6	6	15	75	15	Green
6	Motherboard 2	10	7.6	5	15	75	10	Green
7	Motherboard 3	10	7.6	5	15	75	10	Green
8	Base 1	7	6	4.5	20	70	5	Green
9	Base 2	7	6	4.5	20	70	5	Green
10	USB port1	3	5	3	15	60	-	Black
11	USB port2	3	5	3	15	60	-	Black
12	Connector sockets1	2	5	2.5	30	60	-	Black
13	Connector sockets2	2.5	5	2.5	30	60	-	Black
14	Connector sockets3	2	5	2.5	30	60	-	Black
15	Cooling fan	5.5	6	5	30	60	-	Orange

Tab. 4: List of components.

Type of fastener	Fastener strength	Assemblability	$T_{fastener}$ (s)	T_{tool} (s)
Screw 1	0.9	5	7	10
Screw 2	2	9	17	18
Snap fit 1	0.7	13	9	12
Snap fit 2	2.5	30	25	15
Adhesion	10	40	-	-

Tab. 5: List of fastener methods.

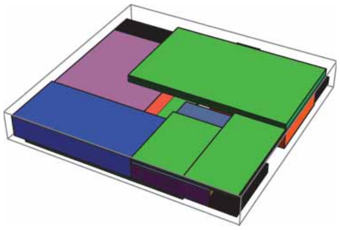


Fig. 4: Optimal layout.

respectively. As for cooling system, most of laptops have cooling fans to exhaust heat mainly generated by CPU. The laptops designed in the case study also have a cooling fan. A cooling fan should be placed in contact with the enclosure in order to take in air from outside the enclosure. Cooling air is taken from a cooling fan, flows through the interstices and is exhausted from the side of the enclosure. Temperature distribution inside the enclosure is analyzed by thermal network method. The heating value and allowable temperature of each component is shown in Tab. 4.

Parameters of GA are as follows. For Layout optimization, population is 50, crossover rate is 0.5, mutation rate is 0.01 and terminal generation is 1000. For fastening method optimization, population is 100, crossover rate is 0.5, mutation rate is 0.01 and terminal generation is 200. These parameters are configured by trial and error.

3.2. Optimization Results

Fig. 4 shows the optimal layout of the 15 inch laptop computer. Size is 23 cm length, 30.5 cm width and 3 cm height. The volume is 2100 cm^3. Fig. 5 shows the disassembly sequence. Tab. 6 shows the type and the number of fasteners, the face in which fasteners are arranged and component temperature. The number of steps required to remove all high-value components written in red in Tab. 4 is 6. The time required to remove high-value components is 482.3 s. Tab. 6 shows that the temperature of all components is lower than the allowable temperature.

To validate the effectiveness of the proposed method, the results are compared with the laptop actually sold in 2007. Fig. 6. shows the component layout. Tab. 7 shows the type and the number of fasteners, the face in which fasteners are arranged. Fig. 7 shows the disassembly sequence obtained by the disassembly rule used in the proposed method.

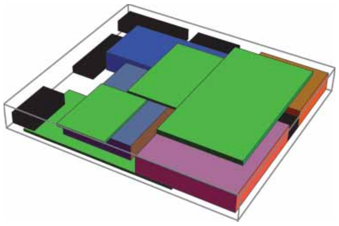


Fig. 6: Layout of the existing laptop sold in 2007.

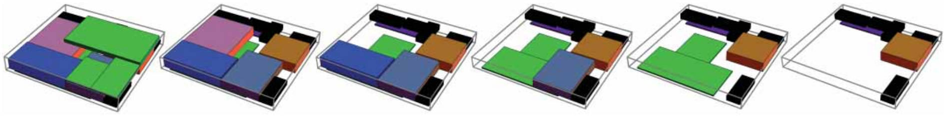


Fig. 5: Disassembly sequence of the optimal layout.

No.	Component	Type of fastener	# of fasteners	Face	Temperature (deg C)
1	HDD	Screw 2	4	Z	57.3
2	Optical drive	Screw 2	5	Z	52.2
3	Battery	Screw 1	4	Z	55.3
4	Speaker	Screw 1	5	Z	51.5
5	Motherboard 1	Screw 1	6	Z	61.3
6	Motherboard 2	Screw 1	4	Z	56.5
7	Motherboard 3	Screw 1	4	Z	55.2
8	Base 1	Screw 1	5	Z	53.6
9	Base 2	Screw 1	4	Z	52.3
10	USB port1	Screw 1	3	Z	52.1
11	USB port2	Screw 1	3	Z	48.3
12	Connector sockets1	Snap fit 1	4	X	42.2
13	Connector sockets2	Snap fit 1	4	X	41.1
14	Connector sockets3	Snap fit 1	4	Y	40.1
15	Cooling fan	Screw 2	3	Y	46.7

Tab. 6: Information about fastening method and component temperature.

No.	Component	Type of fastener	# of fasteners	Face
1	HDD	Screw	5	Z
2	Optical drive	Screw	6	Z
3	Battery	Snap fit	4	Z
4	Speaker	Screw	4	Z
5	Motherboard 1	Screw	5	Z
6	Motherboard 2	Screw	5	Z
7	Motherboard 3	Screw	5	Z
8	Base 1	Screw	4	Z
9	Base 2	Screw	4	Z
10	USB port1	Screw	3	X
11	USB port2	Screw	3	X
12	Connector sockets1	Screw	3	Y
13	Connector sockets2	Screw	3	Y
14	Connector sockets3	Screw	3	X
15	Cooling fan	Screw	4	Z

Tab. 7: Information about fastening method.

The number of steps required to remove all high-value components is 8. The removing time is 524.1 s. These comparisons show that the proposed method can design the component layout and the fastening methods in which high-value components can be removed with less effort.

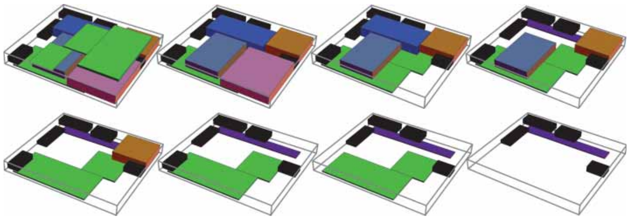


Fig. 7: Disassembly sequence of the existing laptop.

4. CONCLUSION

To make disassembly processes more efficient for the facilitation of reuse and recycle, a new method of optimizing component layout and fastening methods inside a product is developed. The proposed method consists of layout optimization and fastening method optimization. Layout optimization is the main part of the proposed method. Fastening method optimization is applied to each proposal of component layout generated during layout optimization to obtain the optimal fastening methods that can be removed with minimum time and the time is returned to layout optimization. Fitness function of the layout optimization is based on the disassemblability of the component layout and the time to remove fasteners. To explore the optimal component layout and fastening methods, genetic algorithm is used. The proposed method is applied to design of internal devices of a laptop computer. The comparisons between the optimal results and the laptop computer actually sold show that the proposed method can obtain the component layout and fastening methods which high-value components can be removed with less effort.

REFERENCES

[1] Boothroyd, G.; Alting, L.: Design for assembly and disassembly, Annals of CIRP, 41(22), 1992, 625–636. http://dx.doi.org/10.1016/S0007-8506(07)63249-1

[2] Desai, A.; Mital, A.: Evaluation of disassemblability to enable design for disassembly in mass production, International Journal of Industrial Ergonomics, 32(4), 2003, 265–281. http://dx.doi.org/10.1016/S0169-8141(03)00067-2

[3] Edmunds, R.; Kobayashi, M.; Higashi, M.: Using constraint-satisfaction to optimise disassembly sequences generated from AND/OR information, International Journal of Production Research, 50(15), 2012, 4105–4126. http://dx.doi.org/10.1080/00207543.2011.591188

[4] Ellison, G. N.: Thermal computations for electric equipment: Conductive, Radiative, and Convective Air Cooling, CRC Press, 2010.

[5] Gungor, A.; Gupta, S.: Disassembly sequence plan generation using a branch-and-bound algorithm, International Journal of Production Research, 39(3), 2001, 481–509. http://dx.doi.org/10.1080/00207540010002838

[6] Gungor, A.: Evaluation of connection types in design for disassembly (DFD) using analytic network process, Computers & Industrial

Engineering, 50(1–2), 2005, 35–54. http://dx.doi.org/10.1016/j.cie.2005.12.002

[7] Kroll, E.; Carver, B. S.: Disassembly analysis through time estimation and other metrics, Robotics and Computer-Integrated Manufacturing, 15(3), 1999, 191–200. http://dx.doi.org/10.1016/S0736-5845(99)00026-5

[8] Lambert, A. J. D.: Determining optimum disassembly sequences in electronic equipment, Computers & Industrial Engineering, 43(3), 2002, 553–575. http://dx.doi.org/10.1016/S0360-8352(02)00125-0

[9] Sodhi, R.; Sonnenberg, M.; Das, S.: Evaluating the unfastening effort in design for disassembly and serviceability, Journal of Engineering Design, 15(1), 2004, 69–90. http://dx.doi.org/10.1080/09544820310001150152

[10] Takeuchi, S.; Saito, K.: Design for product embedded disassembly, ASME 2005 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, Long Beach, CA, USA, 2005.

[11] Takeuchi, S.; Saito, K.: Design for optimal end-of-life scenario via product embedded disassembly, ASME 2006 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, Philadelphia, Pennsylvania, USA, 2006.

[12] Umeda, Y.: Handbook of inverse manufacturing, Maruzen, 2002.

[13] Yamazaki, H.; Sakanushi, K.; Nakatake S.; Kajitani, Y.: The 3D-Packing by Meta Data Structure and Packing Heuristics, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E83-A(4), 2000, 639–645.