1C-3-3F

Design for Product-Embedded Disassembly with Maximum Profit

Shingo Takeuchi and Kazuhiro Saitou
Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI, USA
{stakeuch, kazu}@umich.edu

Abstract

This paper describes an extension of a method for designing products with built-in disassembly means developed in our previous work, as applied to a realistic example of a desktop computer assembly. Given component geometries and revenues, the method simultaneously determines, through an optimization process, the spatial configuration of component, locator and fasteners such that the product can be most economically disassembled via a domino-like "self-disassembly" process triggered by the removal of one or a few fasteners. A multi-objective genetic algorithm is utilized to search for Pareto-optimal designs in terms of four objectives: 1) satisfaction of the distance specification among components, 2) efficient use of locators on components, 3) profit of overall disassembly process, and 4) mass fraction of retrieved components. The method is applied to a simplified model of Power Mac G4 cube®, and the results inspired a modification to the current design that can improve the ease of disassembly.

Key words: *Design for disassembly, configuration design, disassembly sequence planning, multi-objective genetic algorithm*

1. Introduction

Reduce, reuse and recycle (3R's) are commonly recognized as the essential ways to decrease the excess of abandoned products generated in the modern society. To facilitate material recycling and component reuse, Design for Disassembly (DFD) has focused on enhancing the ease of disassembly at the end of product life cycle. Although many guidelines suggested by the Design for Assembly (DFA) methodology can be applicable to disassembly, products designed for easy assembly do not necessarily facilitate easy disassembly [1], mainly because components to be disassembled are determined based on the trade-off between profit of disassembly and environmental impact [2].

Simply put, the profit of disassembly u of a disassembly sequence consisting of n disassembly steps is given as:

$$u = \sum_{i=1}^{n} (revenue_i - cost_i) \tag{1}$$

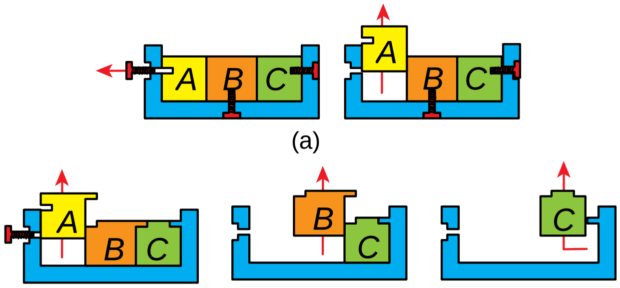
where $revenue_i$ and $cost_i$ are the revenue of the component disassembled at the i-th step and the cost of the i-th disassembly step. While $revenue_i$ depends only on the disassembled components, $cost_i$ is a function of both the disassembled component and the spatial configuration of the component and fasteners in the subassembly just prior to the i-th disassembly step. For economical disassembly of a product, components whose removal would decrease u should not be disassembled.

Environmental impact is another consideration for the determination of components that should be disassembled. Although the retrieval of all components is ideally desired to minimize the environmental impact, this does not usually maximize the profit of disassembly. Hence, regulatory requirements are often imposed in many countries to set minimum requirements manufacturers must satisfy to reduce the environmental impact. For instance, since the EU (European Union) directive on Waste Electrical and Electronic Equipment (WEEE) will require manufacturers to recycle greater than 50 % of the total mass of a product by 2006, certain components may need to be disassembled regardless of the profit of disassembly. It is therefore highly desired to design products that meet regulatory requirements for minimum mass fraction of recycled components and yet provide high profit of disassembly.

Unlike assembly, current practices of product disassembly mostly rely on manual operations, since no product is in the same condition at its end-of-life. Accordingly, labor cost dominates disassembly cost. Since the removal of fasteners is time consuming [3], it is desired to reduce the number of fasteners to reduce the disassembly cost.

These thoughts motivated us to develop a concept of product-embedded disassembly [4], where locators (such as catches and lugs) integral to components are utilized to constrain their relative motions such that the removal of one or a few fasteners activates a built-in optimal disassembly means via a domino-like "self-disassembly" process. Figure 1 (a) shows an example of the conventional assembly, where three components A, B and C are fixed to the container with three fasteners. Figure 1 (b) shows the same assembly designed for product-embedded disassembly, where the fasteners fixing components *B* and *C* are replaced by the locators integral to the components and container.

For the assembly in Figure 1 (a), it does not make an economical sense to remove the fasteners that fix the component whose profit is lower than the labor cost of removing a fastener. This leads to the situation only a few components (*eg*., component *A* only in Figure 1 (a)) can be disassembled for maximum profit, which may not satisfy regulatory requirements. For the assembly in Figure 1 (b), on the other hand, the removal of the fastener fixing component *A* activates the domino-like self-disassembly process of *A*, *B*, and then *C*. Since only one fastener needs to be removed to disassemble all components, the design has a better chance of achieving high profit while satisfying regulatory requirements.



(a)

(b)

Figure 1. (a) conventional assembly and (b) assembly designed for product embedded disassembly [4].

In our previous work [4], a computational method for designing products with such a embedded disassembly means was developed, and its feasibility was demonstrated using a simple example of a hypothetical product. In this paper, the method is extended to allow an application to a realistic example of a desktop computer. Given component information such as component geometries and revenues, the method simultaneously determines the spatial configuration of component, locator and fasteners such that the product can be disassembled in the most preferred sequence, via a domino-like "self-disassembly" process triggered by the removal of one or a few fasteners. A multi-objective genetic algorithm [5, 6] is utilized to search for Pareto-optimal designs of the posed optimization problem in terms of four objectives: 1) satisfaction of a distance specification among components, 2) efficient use of locators on components, 3) profit of overall disassembly process, and 4) mass fraction of retrieved components. The method is applied to a simplified model of Power Mac G4 cube®, and a design alternative inspired by the results for the ease of disassembly is suggested.

2. Related work

2.1 Design for disassembly

While Design for environment (DFE) [7] aims at reducing the environmental impact of products throughout their life cycle, Design for Disassembly (DFD) focuses on the disassembly of products for reuse and recycling at the end of product life. Due to their similarities, many guidelines suggested by the Design for Assembly (DFA) methodology can be applied to DFD [8,9]. Chen *et al.* [10] proposed a cost benefit analysis model for assessing the economics of designing for recyclability. Das *et al.* [11] introduced the Disassembly Effort Index (DEI) score in which seven factors such as time, tools and hazard are evaluated for estimating the ease of disassembly. The concept of product embedded disassembly illustrated in Figure 1 (b) was inspired by Matsui *et al.* [12], who developed a cathode-ray tube (CRT) with a Nichrome wire embedded along the desired separation line to induce thermal stress to crack the tube upon the application of electric current.

These works, however, addresses local modification of an existing product design for the ease of disassembly. Since the disassembly cost heavily depends on the spatial configuration of components and fasteners within an assembly, this would seriously limit the opportunities for improving the disassembleability of a product.

2.2 Disassembly sequence planning

Disassembly sequence planning (DSP) deals with the generation and analysis of feasible disassembly sequences for a given product design. Since disassembly cost depends on the sequence of disassembly, DSP is an integral part of design for disassembly considerations. Techniques originally developed for assembly sequence planning (ASP) [13-17] are utilized for automated disassembly sequence generation [18-22]. Subramani *et al.* [23] focused on the serviceability of a product, proposing a method to generate feasible disassembly sequences to retrieve a component needing service. Zussman *et al.* [24] incorporated end-of-life (EOL) strategies to DSP to obtain the optimal recycling strategy for a product. Dini *et al.* [25] reported a similar work, where a TV set is analyzed under several disassembly scenarios. Hulla *et al.* [2] discussed the effect of situational factors on the optimal disassembly sequence and end-of-life strategy.

These works, however, focuses on analyzing the disassembly sequence of a given product design and do not address the design of the spatial configuration of component and fasteners, which have profound impact on the feasibility and quality of disassembly sequences.

2.3 Configuration design problem

While rarely discussed in the context of disassembly, the design of spatial configuration of given shapes has been an active research area by itself. Among the most popular flavors is the bin packing problem (BPP), where the total volume (or area, for a 2D problem) occupied by components is minimized by varying their spatial configurations. Since this problem is NP-complete, heuristic methods are commonly used. Fujita *et al*. [26] proposed hybrid approaches for 2D plant layout problem (a variant of BPP), where the topology and geometry of a layout are determined by simulated annealing (SA) and the generalized reduced gradient (GRG) method, respectively. Kolli *et al*. [27] also applied simulated annealing to solve a 2D bin packing problem using multi-resolution quad trees. Corcoran and Wainwright [28] solved a 3D bin packing problem with genetic algorithm (GA) using multiple crossover methods. Jain and Gea [29] proposed a geometry-based crossover operation for a 2D packing problem. Grignon and Fadel [30] presented a configuration design optimization method by using a multi-objective GA, where static and dynamic balance and maintainability are considered in addition to volume.

These works, however, do not address the integration with DSP.

3. Method

The proposed method can be summarized as the following optimization problem:

- **Given:** component information, functional information, locator library
- **Find:** spatial configuration of components and locators
- **Subject to:** no overlap among components, no unfixed components prior to disassembly, satisfaction of contact specification, assembleability of components.
- **Minimizing:** violation of distance specification, redundant use of locators
- **Maximizing:** profit of disassembly, mass fraction of retrieved components.

Due to the existence of four objectives in the problem, a multi-objective genetic algorithm (MOGA) [4, 5] is utilized to obtain Pareto optimal solutions.

3.1 Input

The following three inputs are assumed as given: 1) the component information, 2) the function information, and 3) the locator library.

The component information consists of component geometries and component revenues through recycling or reuse (*revenue* in Equation (1)). Due to the efficiency in checking contacts and the simplicity in modifying geometries, component geometries are represented by voxels [4, 31, 32].

The spatial configurations of components are often constrained by their functional relationships. For example, in a computer assembly, a heat sink should physically contact the CPU (*contact specification*), and a hard disk should connect to a battery via the shortest possible wire (*distance specification*). Accordingly, the function information consists of the contact specification and the distance specification. The contact specification is defined as a set of pairs of components requiring adjacency to each other. This is newly introduced in this paper, in order to better model the adjacency of components absolutely required for product function. As defined in [4], the distance specification is a set of weights to pairs of components, which signify the relative importance of minimizing the respective distance.

Since the types of feasible locators depend on manufacturing and assembly processes, they are pre-specified by a designer as the locator library. It is a set of locators for a specific application domain, which can be potentially added on each component to constrain its relative motion. Based on the observation of locators used in the Power Mac G4 cube®, the following case study uses the locator library consisting of the six locators[1] in Figure 2. Note that Slot can be used between two motherboards only.

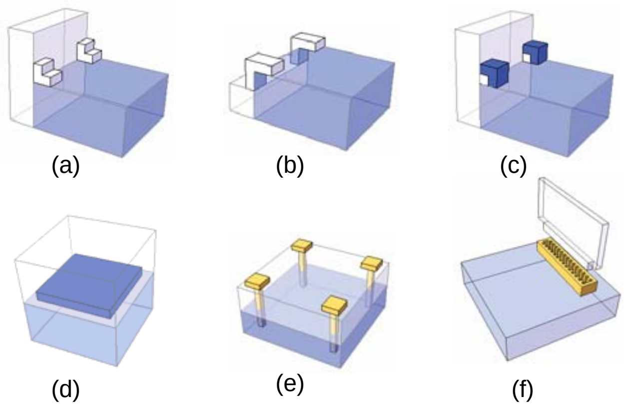


Figure 2. Sample locator library: (a) catch, (b) lug, (c) track, (d) boss, (e) screw, and (f) slot.

3.2 Design variables

There are two design variables for the problem. The first design variable, the *configuration vector,* represents the spatial configuration and dimensional change of each component:

$$\boldsymbol{x} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}) \quad (2)$$

---

[1] Fasteners are considered as a special case of locators and are included in a locator library as Screw.

$$\boldsymbol{x}_i = (\boldsymbol{t}_i, \boldsymbol{r}_i, \boldsymbol{d}_i,);\ \ i = 0, 1, \ldots\ n\text{-}1 \qquad (3)$$

where n is the number of components in the assembly, $\boldsymbol{t}_i$ and $\boldsymbol{r}_i$ are the vectors of the translational and rotational motions of component i with respect to the global reference frame, and $\boldsymbol{d}_i = (d_0, d_1, \ldots, d_{f\text{-}1})$ is a vector of the offset values of the faces of component i in their normal directions. Due to the voxel representation, translations and offsets are limited to integer multiples of the size of a voxel. Similarly, rotations are limited to $+90^o$, -90^o and $+180^o$. Note that the dimensional changes are considered only for the components whose dimensions are assumed unfixed.

The second design variable, the *locator vector*, represents the spatial configuration of the locators on each component:

$$\boldsymbol{y} = (\boldsymbol{y}_0, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{m\text{-}1}) \qquad (4)$$

where $m = n\,(n\text{-}1)/2$ is the number of pairs of components in the assembl, and $\boldsymbol{y}_i$ (i = 0, 1, … m-1) is a vector containing the information on the motion constraints between the i-th pair of components and the configuration of locator that realizes them. The detailed description of the locator vector is found in [4].

3.2 Constraints

The spatial configuration of components and locators as specified by design variables $\boldsymbol{x}$ and $\boldsymbol{y}$ must satisfy the following constraints:

1. No overlap of components
2. No unfixed components prior to disassembly
3. Satisfaction of contact specification
4. Assembleability of components

While constraint 1 is for physical feasibility of a configuration, constraints 2 and 3 are for product function. Constraint 4 is for ensuring all components, weather they are to be disassembled or not, can be assembled when the product is first put together.

3.4 Objective functions

A candidate design as specified by $\boldsymbol{x}$ and $\boldsymbol{y}$ is evaluated according to four criteria: (1) satisfaction of the distance specification, (2) efficient use of locators, (3) profit of disassembly, (4) the mass fraction of the retrieved components.

The first objective function (to be minimized) is for the satisfaction of the distance specification, given as:

$$f_1(\boldsymbol{x}, \boldsymbol{y}) = \sum_i w_i d_i \qquad (5)$$

where w_i is the weight of the importance of distance d_i in the distance specification between two designated voxels.

The second objective function (to be minimized) is for the efficient use of locators, given as the total increase in the manufacturing cost due to the addition of locators to components:

$$f_2(\boldsymbol{x}, \boldsymbol{y}) = \sum_i mc_i \qquad (6)$$

where mc_i is the manufacturing cost of the i-th locator in the assembly.

The third objective function (to be maximized) is for the profit of the overall disassembly process. Since assembly $a(\boldsymbol{x}, \boldsymbol{y})$ specified by $\boldsymbol{x}$ and $\boldsymbol{y}$ can generally be disassembled in multiple sequences, the objective function is defined as the profit of the best (most profitable) disassembly sequence:

$$f_3(\boldsymbol{x}, \boldsymbol{y}) = \max_{q \in Q_{xy}} \{ \max_{pq \in P_q} u(a, pq) \} \qquad (7)$$

where:

- Q_{xy} is the set of feasible 2-disassembly sequences [4, 18, 33] (each disassembly step consists of less than two consecutive translational motions) of a.
- P_q is the set of sub-sequences of $q \in Q_{xy}$ in which a is partially disassembled
- $u(a, pq)$ is the profit of disassembling a in $pq \in P_q$

Given a 2-disassembly sequence $q \in Q_{xy}$, the maximum profit $u_a \equiv u\ (a,\ pq^*)$ among all partial disassembly sequence of q in Equation (7) can be obtained by following the disassembly steps in q until the continuation is unprofitable. Considering a disassembly step in q that separates subassembly s into two subassemblies ss and st, the maximum profit u_s of partially disassembling s in sub-sequences of q can be recursively defined as follows:

$$u_s = \begin{cases} r_s & \text{if } s \text{ is a component} \\ 0 & \text{if } u_{ss} + u_{st} - c_s < 0 \\ u_{ss} + u_{st} - c_s & \text{otherwise} \end{cases} \qquad (9)$$

where r_s is the revenue of s (if s is a component) and c_s is the cost of disassembling s into ss and st.

The disassembly cost c_s in Equation (9) depends on the orientation changes, the amount of work required for disassembly, and the accessibility of fasteners (screw and slot) during the disassembly operation, given as:

$$c_s = \omega_0 \cdot dc_0 + \omega_1 \cdot dc_1 + \omega_2 \cdot dc_2 \qquad (10)$$

where ω_j is weight, dc_0 is the number of orientation changes, dc_1 is the sum of the moved distances multiplied by the weight of disassembled components, dc_2 is the weighted sum of accessibilities of removed fasteners. The accessibility ac_f of fastener f is defined as:

$$ac_f = 1.0 + \omega_f / (aa + 0.01) \tag{11}$$

where ω_f is weight and aa is the area of the mounting face of the fastener accessible from outside of the product in its normal direction.

Finally, the last objective function (to be maximized) is the mass fraction of the retrieved components given as:

$$f_4(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{M} \sum_i m_i \tag{12}$$

where M is the total mass of an assembly and m_i is the mass of the i-th retrieved components in the disassembly sequence that maximizes the profit of disassembly (Equation (7)).

4. Case study

The Power Mac G4 Cube® manufactured by Apple Computer, Inc. (Figure 3) is considered as a realistic application of the proposed method. The assembly is first simplified to extract ten (10) essential components and their primary liaisons (Figure 4 (a)). Geometries of these essential components are then approximately represented by voxels, whereas functional relationships are represented by either the contact specification or the distance specification (Figure 4 (b)). The contact is specified between component B (heat sink) and C (CPU), and C (motherboard) and G (memory) since physical contacts between these components are important for product function. Component A (container) is considered as fixed. The locator library in Figure 2 is used. Table 1 lists the approximate revenues of each component, reflecting their current relative values in the PC recycling markets in the United States.

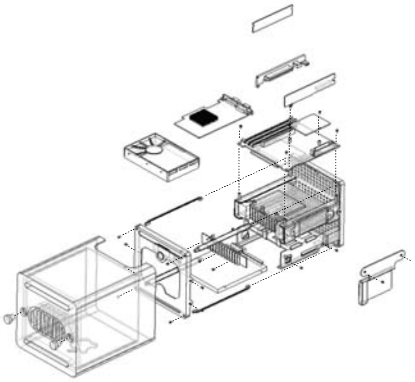


Figure 3. Assembly of Power Mac G4 Cube®.

Given the component geometry and function information in Figure 4, the component revenues in Table 1, and the locator library in Figure 2, the proposed method found thirty (30) Pareto optimal designs, as a result of an optimization run with the number of population of 100 and the maximum number of generation of 300. The running time is approximately 240 hours (10 days) with a standard desktop PC.

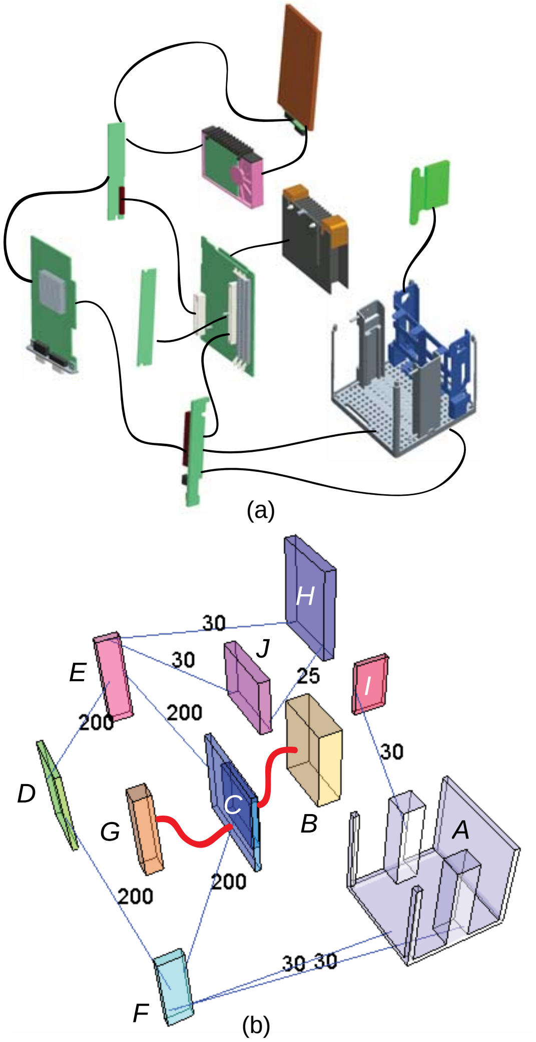


Figure 4. (a) major components and liaisons of assembly in Figure 3, and (b) simplified geometries with functional information.

Since there are four objective functions f_1, f_2, f_3, and f_4, the resulting 4-dimensional space is projected onto six 2-dimensional spaces in Figure 5 (a)-(f). Figure 6 shows four representative Pareto optimal designs (without fasteners), annotated as R_1, R_2, R_3 and R_4 Designs R_1, R_2 and R_3 are the best results only considering f_1, f_2 and f_3 (also f_4), respectively, while R_4 is a balanced result in all four objectives. The objective function values of these representative designs are listed Table 2 and also plotted on a spider web diagram in Figure 7. Figure 8 shows details of R_3. Four fasteners (two screws and two slots) shown in Figure 8 (a) are used to fix components, and Figure 8 (b)-(k) shows an optimal disassembly sequence that maximizes the profit of disassembly. Thanks to the ingenious use of locators, the number of fasteners is reduced, which in turn realizes the retrieval of all components with the maximum profit of disassembly.

Table 1. Revenues of components in Figure 4 (b).

A	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*
80	60	300	150	80	80	450	300	60	500

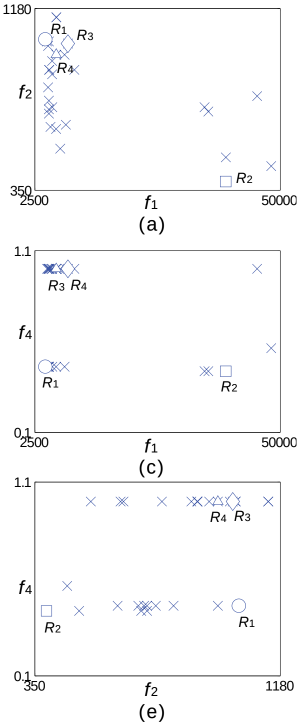


Figure 5. Distribution of Pareto optimal designs in six 2-dimensional spaces (a)-(f).

Since the proposed method does not consider thermal and strength factors, the obtained designs cannot be directly used as final product designs. However, they can provide early insights to designers for the potential improvements of current designs. For instance, consider the retrieval of the hard disk (HD) and the compact disk (CD) drives in the current design shown in Figure 9. Despite the use of locators on the frame, many screws still need to be removed to retrieve the drives as shown in Figures 9 (b) and 9 (d). On the other hand, Figure 10 shows an alternative design inspired by the part of the optimal disassembly sequence in Figure 8 (b)-(e), where the frame is divided into two pieces. The first piece shown in Figure 10 (d) functions as component *I* in Figure 8 (b), whereas the second piece integrates with the subsidiary heat sink shown in Figure 9 (e). Since the HD drive is completely fixed by locators and other components as in Figure 8 (b)-(e), no screws are necessary to fix the HD drive, improving the ease of disassembly.

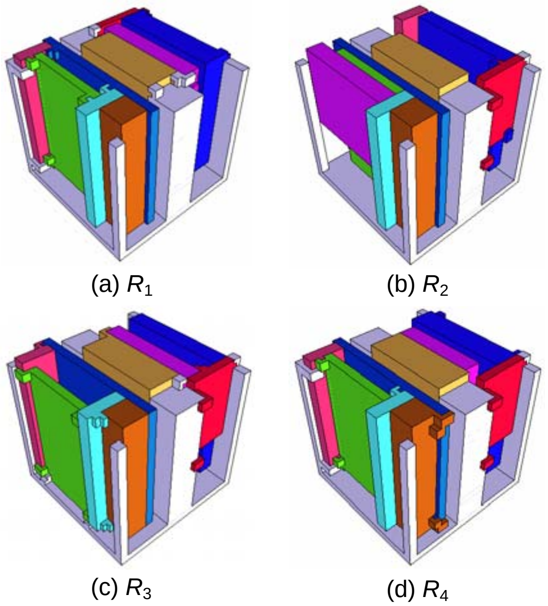

(a) R_1
(b) R_2
(c) R_3
(d) R_4

Figure 6. Representative Pareto optimal designs (a) R_1, (b) R_2, (c) R_3 and (d) R_4.

Table 2. Objective function values for R_1 – R_4.

	f_1	f_2	f_3	f_4
R_1	4530.07	1040	754.774	0.461538
R_2	39469.1	390	1101.66	0.435897
R_3	8929.98	1020	1352.96	1.0
R_4	6647.91	970	1213.66	1.0

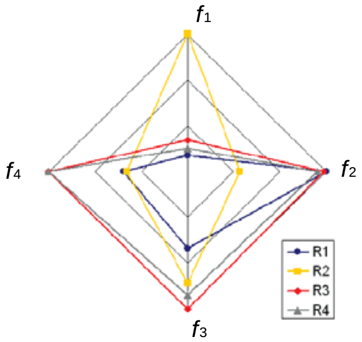


Figure 7. Spider web diagram for the objective function values of R_1 – R_4.

5. Conclusion and future work

This paper presented an extension of our previous work on product-embedded disassembly [4]. The method simultaneously determines, through an optimization process, the spatial configuration of component, locators, and fasteners, such that a product can be most economically disassembled via a domino-like “self-disassembly” process initiated by the removal of one or a few fasteners. The method was successfully applied to a realistic example of a desktop computer assembly, and the results inspired a modification to the current design that can improve the ease of disassembly.

Future work includes the integration with life-cycle assessment to quantify the trade-off between economical profitability and environmental impact, and the development of more efficient optimization algorithms to reduce computational time. In addition, the incorporation of thermal evaluation is sought, due to its importance in the configuration design of personal computers and other consumer electric appliances.

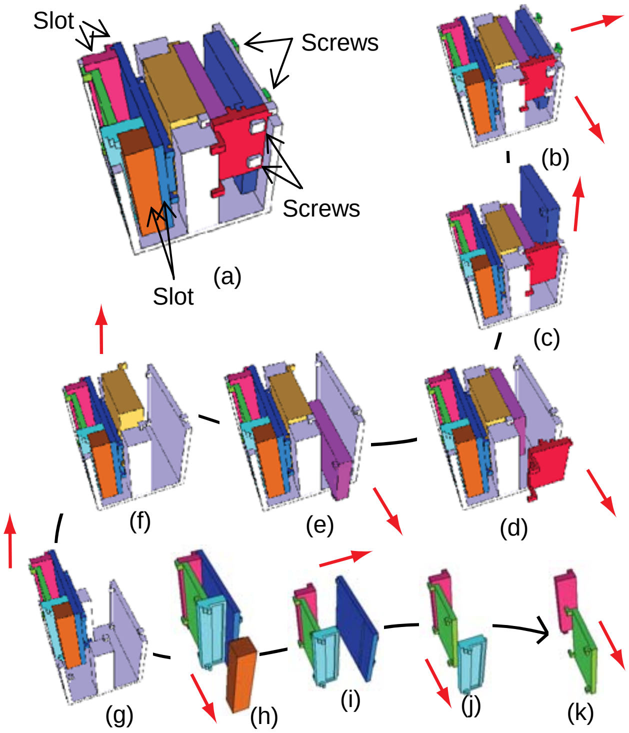


Figure 8. Details of R_3: (a) 4 screws and 4 slots, and (b)-(k) an optimal disassembly sequence maximizing the profit of disassembly.

Acknowledgements

The funding for this research was provided by the National Science Foundation of the United States through grant # BES-0124415. Any options, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

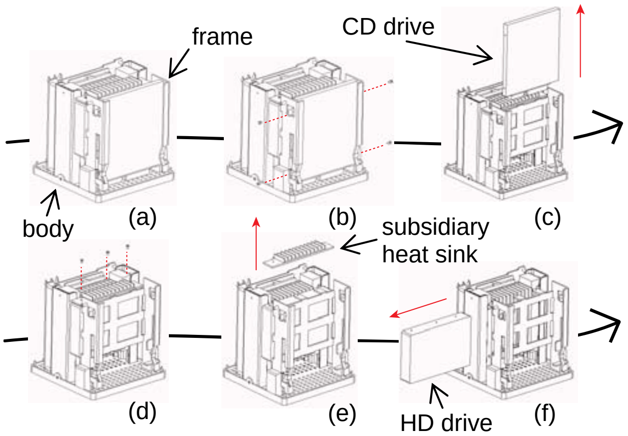


Figure 9. Disassembly of the current design.

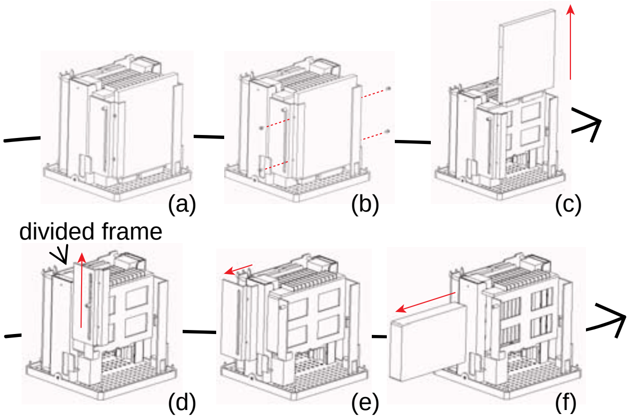


Figure 10. Disassembly of the suggested design.

References

[1] T. Harjula, B. Rapoza, W. Knight, and G. Boothroyd, 1996, “Design for Disassembly and the Environment,” *Annals of the CIRP*, Vol. 45, No. 1 pp. 109-114.

[2] A. Hulla, K. Jalali, K. Hamza, S. Skerlos, and K. Saitou, 2003, “Multi-criteria Decision Making for Optimization of Product Disasembly under Multiple Situatioins,” *Environmental Science and Technology*, vol. 37, No. 23, pp. 5303 -5313.

[3] D. Shetty, K. Rawolle, and C. Campana, 2000, “A new methodology for ease-of-disassembly in production design,” *Recent Advances in Design for Manufacture (DFM),* ASME.

[4] S. Takeuchi, and K. Saitou, 2005, .“Design for Product-Embedded Disassembly”, *Proceedings of the 2005 ASME Design Engineering Technical Conferences*, Long Beach, California, September 24-28, DETC2005-85260.

[5] C. M. Fonseca and P. J. Fleming, 1993, “Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization,” *Proceedings of the Fifth International Conference on Genetic Algorithms*, San Mateo, California, pp. 416-423.

[6] C. A. Coello, D. A. Veldhuizen, and G. B. Lamont, 2002*, Evolutionary algorithms for solving multi-objective optimization problems*, Kluwer Academic Publishers.

[7] D. Navinchandra, 1991, "Design for Environmentability", Proceedings of the *3rd International Conference on Design Theory and Methodology,* Sep 22-2, Miami, FL, USA, pp. 119-125

[8] G. Boothroyd and L. Alting, 1992, "Design for assembly and disassembly," *Annals of CIRP*, vol. 41, no. 2.

[9] E. Kroll, B. Beardsley, and A. Parulian, 1996, "A Methodology to Evaluate Ease of Disassembly for Product Recycling," *IIE Transactions*, vol. 28, No 10, pp. 837-845.

[10] R. W. Chen, D. Navin-Chandra, and F. Prinz, 1993, "Product Design for recyclability: A cost benefit analysis model and its application" *IEEE International Symposium on Electronics & the Environment*., Virginia, pp. 178-183.

[11] S. K. Das, P. Yedlarajiah, and R. Narendra, 2000, "An Approach for Estimating the End-Of-Life Product Disassembly Effort and Cost", *International Journal of Production Research*, vol. 38, No. 3, 657-673.

[12] K. Matsui, K. Mizuhara, K. Ishii, and R. M. Catherine, 1999, "Development of Products Embedded Disassembly Process Based on End-Of-Life Strategies," *EcoDesign 1999, First International Symposium on Environmentally Conscious Design and Inverse Manufacturing*, February 1-3, pp. 570-575.

[13] L. S. Homem dé Mello and A. C. Sanderson, 1990, "AND/OR Graph Representation of Assembly Plans", *IEEE Transactions on Robotics and Automation*, vol. 6, pp. 188-99.

[14] T. L. De Fazio and D. E. Whitney, 1987, "Simplified Generation of All Mechanical Assembly," *IEEE Journal of Robotics and Automation*, vol. 3, No. 6, pp. 640-658.

[15] S. Lee and Y. G. Shin, 1990, "Assembly Planning Based on Geometric Reasoning," *Computer and Graphics*, vol. 14, No. 2, pp. 237-250.

[16] L. S. Homem dé Mello and A. C. Sanderson, 1991, "A Correct and Complete Algorithm for the Generation of Mechanical Assembly Sequences," *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 228-40.

[17] D. F. Baldwin, T. E. Abell, M.-C. M. Lui, T. L. De Fazio, and D. E. Whitney, 1992, "An Integrated Computer Aid for Generating and Evaluating Assembly Sequences for Mechanical Products," *IEEE Transactions on Robotics and Automation*, vol. 7, No. 1, pp. 78-94.

[18] T. C. Woo and D. Dutta, 1991, "Automatic Disassembly and Total Ordering in Three Dimensions," *Transactions of ASME, Journal of Engineering for Industry*, vol. 113, pp. 207-213.

[19] D. Dutta and T. C. Woo, 1995, "Algorithm for Multiple Disassembly and Parallel Assemblies," *Transactions of ASME, Journal of Engineering for Industry*, vol. 117, pp. 102-9.

[20] S.-F. Chen, J. H. Oliver, S.-Y. Chou, and L.-L. Chen, 1997, "Parallel disassembly by onion peeling," *Transactions of ASME, Journal of Mechanical Design*, vol. 119, pp. 267-274.

[21] H. Srinivasan and R. Gadh, 2000, "Efficient Geometric Disassembly of Multiple Components from An Assembly Using Wave Propagation," *Transactions of ASME*, *Journal of Mechanical Design*, vol. 122, pp. 179-184.

[22] S. G. Kaufman, R. H. Wilson, R. E. Jones, T. L. Calton, and A. L. Ames, 1996, "The Archimedes 2 Mechanical Assembly Planning System," Proceedings of the *1996 IEEE International Conference on Robotics and Automation,* April, 1996, Minneapolis, Minnesota.

[23] A. K. Subramani and P. Dewhurst, 1991, "Automatic generatin of disassembly sequences", *Annals of CIRP*, vol. 40, no. 1, pp. 115-118.

[24] E. Zussman, A. Kriwet, and G. Seliger, 1994, "Disassembly-Oriented Assessment Methodology to Support Design for Recycling", *Annals of CIRP*, vol. 43, no. 1, pp. 9-14.

[25] G. Dini, G., Failli F., and Santochi, M., 2001, "A Disassembly Planning Software System for the Optimization of recycling processes", Production Planning and Control, 2001, vol. 12, no. 1, pp. 2-12.

[26] K. Fujita, S. Akagi, and S. Shimazaki, 1996, "Optimal Space Partitioning Method Based on Rectangular Duals of Planar Graphs," *JSME International Journal*, vol. 60, pp. 3662-3669.

[27] A. Kolli, J. Cagan and R. Rutenbar, 1996, "Packing of Generic, Three-Dimensional Components Based on Multi-Resolution Modeling," Proceedings of the *1996 ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, August 18-22, 1996, Irvine, California.

[28] A. L. Corcoran III and R. L. Wainwright, 1992, "A Genetic Algorithm for Packing in Three Dimensions," Proceedings of the *1992 ACM/SIGAPP Symposium on Applied Computing*, Kansas City, Missouri.

[29] S. Jain and H. C. Gea, 1998, "Two-Dimensional Packing Problems Using Genetic Algorithm," *Journal of Engineering with Computers*, 1998, vol. 14, pp. 206-213.

[30] P. M. Grignon and G. M. Fadel, 1999, "Configuration Design Optimization Method," Proceedings of the *1999 ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, (Paper No. DETC99/DAC-8575), September 12-15, Las Vegas, Nevada.

[31] S. Minami, K. F. Pahng, M. J. Jakiela, and A. Srivastave, 1995, "A Cellular Automata Representation for Assembly Simulation and Sequence Generation," *IEEE International Symposium on Assembly and Task Planning*, pp.56-65, August 10-11, Pittsburgh, Pennsylvania.

[32] R. C. W. Sung, J. R. Corney, and D. E. R. Clark, 2001, "Automatic Assembly Feature Recognition and Disassembly Sequence Generation," *Transactions of ASME,* J*ournal of Computing and Information Science in Engineering*, vol. 1, pp. 291-299.

[33] D. Beasley and R. R. Martin, 1993, "Disassembly Sequences for Objects Built from Unit Cubes," *Journal of Compute-Aided Design*, vol. 25, No. 12.