Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2012, Article ID 569429, 13 pages
doi:10.1155/2012/569429

Research Article

A Label Correcting Algorithm for Partial Disassembly Sequences in the Production Planning for End-of-Life Products

Pei-Fang (Jennifer) Tsai

Department of Industrial Engineering and Management, National Taipei University of Technology, 10608 Taipei, Taiwan

Correspondence should be addressed to Pei-Fang (Jennifer) Tsai, ptsai@ntut.edu.tw

Received 8 February 2012; Accepted 8 April 2012

Academic Editor: Jung-Fa Tsai

Copyright © 2012 Pei-Fang (Jennifer) Tsai. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Remanufacturing of used products has become a strategic issue for cost-sensitive businesses. Due to the nature of uncertain supply of end-of-life (EoL) products, the reverse logistic can only be sustainable with a dynamic production planning for disassembly process. This research investigates the sequencing of disassembly operations as a single-period partial disassembly optimization (SPPDO) problem to minimize total disassembly cost. AND/OR graph representation is used to include all disassembly sequences of a returned product. A label correcting algorithm is proposed to find an optimal partial disassembly plan if a specific reusable subpart is retrieved from the original return. Then, a heuristic procedure that utilizes this polynomial-time algorithm is presented to solve the SPPDO problem. Numerical examples are used to demonstrate the effectiveness of this solution procedure.

1. Introduction

Product recovery, or remanufacturing, has been considered as one of the most profitable options in dealing with the end-of-life (EoL) products. The benefit of product recovery is especially more attractive when a facility is capable of performing both manufacturing and remanufacturing processes, and the coordination of these two processes can be included in the production planning and scheduling. Griese et al. [1] discussed the economic challenges for reuse and the main technical obstacles in three product categories: medical equipment, automotive electronics, and computers. They argued that, for personal computers, reuse and repair appeared to have more potential than pure recycling materials. Similar benefit was confirmed by Grenchus et al. [2] from the practice at the IBM Endicott asset recovery center. They found that, with little disassembly effort, functional parts that were recoverable had more resale value than plain material recovery.

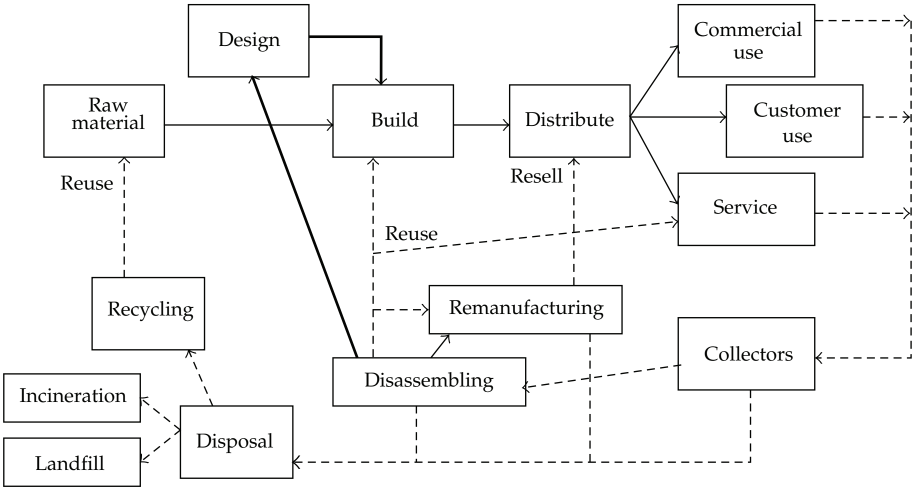


Figure 1: Integrated supply chain for original equipment manufacturers. Adopted from [4].

For any original equipment manufacturer (OEM) with the capacity in performing assembly and disassembly operations, the retrieved components from returned products can be integrated with their forward production. As shown in Figure 1, a complete product life cycle is defined by forward logistics (solid line) and reverse logistics (dotted line). Information (bold line) from disassembly operation can further assist in design and manufacturing new products. Retrieving spare parts from returned products has been one of the most prominent strategies. Fleischmann [3] observed that, when the returned used equipments were integrated into the spare part planning, a push policy in which returned equipments were tested and dismantled as soon as available can achieve higher service level in IBM's spare part management.

It remains as one of the biggest challenges to develop the techniques in production planning system for product recovery to be sustainable [5]. For regular assembly, demand can be determined in advance and hence the required resources can be planed and scheduled along the time horizon. However, for disassembly process planning, variation in quantity and quality of returned products is so huge that it fails to fit into any available planning scheme. Even when the demand for remanufactured products is known with a set of available returned products, it is still challenging to decide how these products should be dismantled to minimize the disassembly effort for those refurbished products. Kasmara et al. [6] used an integer programming model that included sales and returns in each period with the objective function as maximization of profits. Clegg et al. [7] presented a linear programming model of production planning for both new and remanufactured products. Some studies focused on the effect of average flowtime for both assembly and disassembly operations under different scenarios in planning mixes [8–13].

The ability to salvage the value of these returned products relies both on the disassembly capacity and the ability to find the most cost-effective disassembly sequences to retrieve valuable parts. This research is motivated to find disassembly sequences with minimum operation costs in the production planning for EoL products. A single period planning is considered due to the inherent fluctuation in the demand and supply of EoL products in different periods. Moreover, a partial disassembly policy is considered for better profit in product recovery.

2. Literature on Disassembly Planning

Disassembly is the process of dismantling a product through successive steps at the end of its life so as to recover useful components or subassemblies, for resale, reuse, or proper disposal. A product, as a whole, may not be repairable at the end of its conventional useful life, yet there might be components and subassemblies that are in good enough condition for use in a new product or in the remanufacture of an old one [14]. The efficient retrieval of these parts/subassemblies will not only cut down the production cost but will also reduce the associated disposal costs and, consequently, the environmental hazards.

Taleb and Gupta [15] addressed the scheduling problem of disassembly operations with commonality in the parts or materials. Then, a scheduling mechanism was provided to determine the total disassembly requirement as well as the time to release along the planning horizon. Langella [16] developed heuristics for the disassembly planning when a returned product was used in multiple purposes. Gungor and Gupta [17] used a heuristic algorithm to evaluate total costs among different disassembly strategies for returned products using the precedence relationship among subassemblies. Deterministic formulations originated from variations of travelling salesman problem (TSP) were used to find optimal disassembly sequences [18, 19]. Johnson and Wang [20] considered the disassembly precedence relationship according to a bill of material (BOM) of the product and formulated the problem as a two-commodity network formulation. Lambert [21] solved a similar formulation as in [20] with sequence-dependent setup time by using an iterative heuristic approach.

Graph-based representation is often used to find all feasible disassembly sequences. De Fazio and Whitney [22] proposed a network of liaison sequences, or precedence sequences, that are generated from the bill of material (BOM) of a product. Sarin et al. [19] incorporated the precedence constraints into a tree-structured network with additional nodes representing the connection and joints between subparts. Zhang and Kuo [23] and Kuo [24] proposed a component-fastener graph to represent the assembly relationship to find the most profitable dismantle sequence for returned products. Homem de Mello and Sanderson [25, 26] introduced an AND/OR graph representation of assembly plan which includes all possible sequences of operations. Lambert [27] utilized the AND/OR graph to generate sequences in retrieving subassemblies with the consideration of disassembly line balancing.

The objective of this research is to find an optimal partial disassembly sequence to retrieve reusable subassemblies or subparts from EoL products in single planning period or referred to as single period partial disassembly optimization (SPPDO) problem herein. A label correcting algorithm is proposed to solve the AND/OR graph as a shortest path problem. Furthermore, a heuristic procedure is developed to utilize this label correcting algorithm in solving the SPPDO problem. The remainder of the paper is organized as follows. In the next section, the representations of disassembly sequences and AND/OR graph are briefly discussed. The subsequent sections further introduce the mathematical formulation of the problem, the label correcting algorithm, and a heuristic procedure for solving SPPDO. This is followed by a section that illustrates how the heuristic algorithm works with numerical examples. Then, some concluding remarks are summarized in the last section.

3. Formulation of Partial Disassembly Optimization Problem

3.1. Disassembly Sequence and AND/OR Graph

For a given return, the feasible disassembly sequence is restricted to the design of that product. After specifying the structure of a product, it is prerequisite to develop the relations

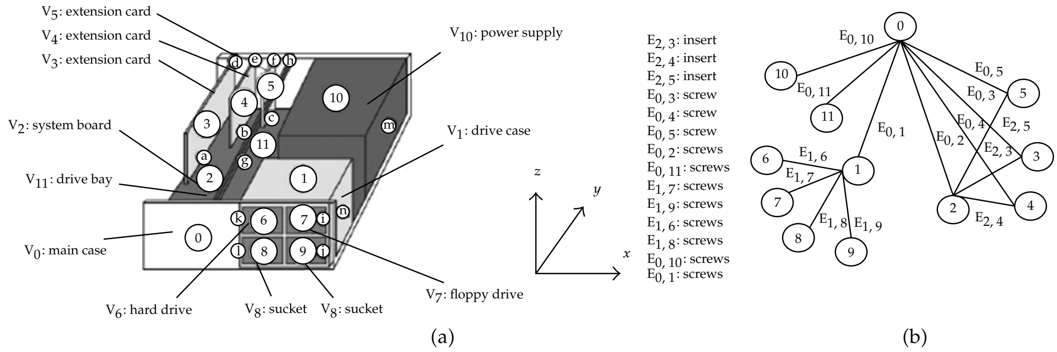
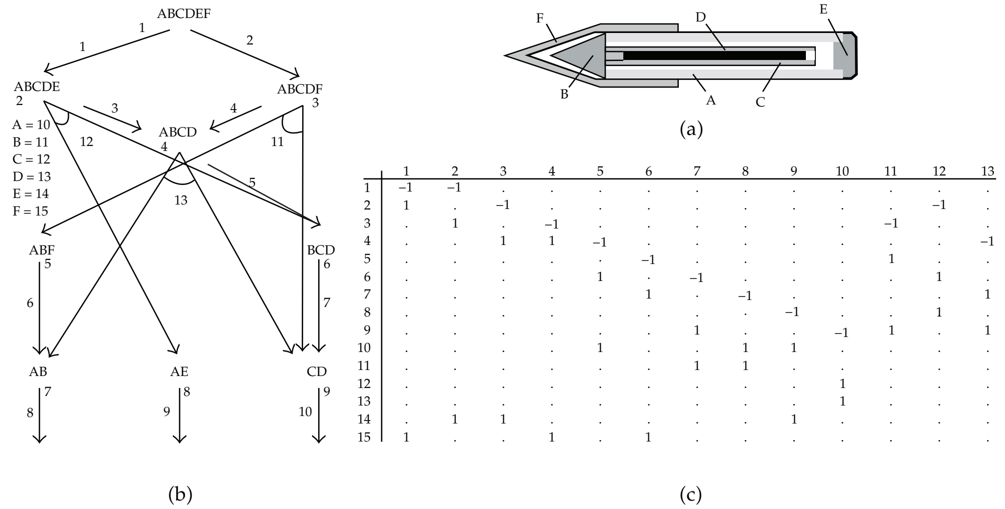


Figure 2: (a) The structure and (b) component-fastener graph for a partially disassembled PC [24].



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

Figure 3: (a) The structure, (b) AND/OR disassembly tree, and (c) transition matrix for a ballpoint [21].

between subassemblies and parts where separation operations are possible. A component-fastener graph can be used to represent the assembly relationship [23, 24]. As shown in Figure 2, if two components are attached or joined by fasteners, then these two components are connected by an undirected edge in the component-fastener graph.

A disassembly AND/OR graph is another useful representation for all possible disassembly sequences. An AND/OR graph is a directed hypergraph where subassemblies are represented as nodes. If more than one subassembly/part can be separated from a parent assembly in one disassembly operation, a hyperarc (AND arc) is used to indicate this operation and connects the parent node to all child nodes. Otherwise, a directed arcs (OR arcs) are used. Figure 3(a) illustrates the drawing of a ballpoint product with associated AND/OR graph in Figure 3(b). The assembly {ABCDE} yields two subassemblies {BCD} and {AE} through the disassembly operation {12}, and this operation is represented as an AND arc ("∪" arc). Moreover, this hypergraph is compact which requires a reduced number of nodes/arcs to enumerate all partial sequences of disassembly operations [25].

This AND/OR graph can also be represented completely via a transition matrix T. Suppose I be the set of subassemblies/subparts and J be the set of operations, the element T_{ij} has a value of 1 if a subassembly $i \in I$ is released by some operation $j \in J$ or -1 if subassembly

$i \in I$ is to be dismantled via operation $j \in J$. Figure 3(c) defines the associated transition matrix for the ballpoint product in Figure 3(a), where $I = \{1 \cdots 15\}$ and $J = \{1 \cdots 13\}$. For example, the operation $j = 12$ will dismantle one subassembly $i = 2$ or assembly {ABCDE}, for two corresponding subassemblies $i = 6$ and $i = 8$ or assemblies {BCD}, and {AE}, respectively.

3.2. Mathematical Model

The partial disassembly optimization problem considered in this research is defined as a single-period partial disassembly optimization (SPPDO) problem. The problem assumed that the disassembly decision is to consider myopic optimal sequences given the quantity of returned products and the demands for reusable subparts in the period. For the product with I as the set of feasible subassemblies/subparts and J as the set of disassembly operations, two more sets are defined as follows: I_0 as the set of original products, $I_0 \subseteq I$; I_f as set of reusable subparts or target subparts, $I_f \subseteq I$.

Consider the AND/OR graph G for this returned product with I nodes and J arcs and the transition matrix T for the product. A feasible disassembly sequence to retrieve reusable parts from a returned product can be defined as a path from a source node in I_0 to the corresponding target nodes in I_f. Let c_j be the cost of operation j, and $c_j \geq 0$, for all $j \in J$. In this model, the available quantity is defined as a negative number $b_i \leq 0$, for original returns i, for all $i \in I_0$. For each reusable subpart i, the demand is defined as $b_i \geq 0$, for all $i \in I_f$. Intermediate subassembly has zero demand; that is, $b_i = 0$, for all $i \in I \setminus \{I_0 \cup I_f\}$. Without loss of generality, we assume that this model has enough supply of original returns; that is, $\sum_{i \in I} b_i \leq 0$. Suppose y_j be the number of operation j, $j \in J$, needed in this planning period, then an optimal partial disassembly sequence with minimum total disassembly cost can be obtained by solving the formulation proposed as follows:

SPPDO

$$\text{Minimize} \quad \sum_{j \in J} c_j \cdot y_j, \tag{3.1}$$

$$\text{Subject to} \quad \sum_{j \in J} T_{ij} \cdot y_j \geq b_i, \quad \forall i \in I_0 \cup I_f, \tag{3.2}$$

$$\sum_{j \in J} T_{ij} \cdot y_j \geq 0, \quad \forall i \in I \setminus \{I_0 \cup I_f\}, \tag{3.3}$$

$$y_j \geq 0 \text{ and integer } \forall j \in J. \tag{3.4}$$

This SPPDO model is a generalized minimum cost flow problem where the arc in a graph include both hyperarcs (AND arcs) and regular directed arcs (OR arcs). The constraint set (3.2) is to ensure that the demands of reusable subparts are fulfilled or the required quantities for original returns are still sufficient. The constraint sets (3.3) and (3.4) are nonnegative constraints on the resulting quantity of intermediate subassemblies/parts and the number of operations needed. It is worth noting that this AND/OR graph is acyclic, and the sum of degrees from all nodes might not be zero. Further note that, since the transition matrix T does not have the property of total unimodularity, this SPPDO formulation can only be solved as a pure integer programming (IP) problem.

In the literature, a searching algorithm for an AND/OR graph with different interpretation is available, but it is not applicable to solve the SPPDO problem directly. AND/OR graph is often used to represent a problem-solving process which transforms the original problem into several subproblems [28, 29]. Each node represents a distinct problem. The node represents the original problem is referred to as the starting node or root node. A terminal node, or leaf node, in this graph represents a problem whose solution is either known to exist or not to exist. A directed arc is linked from a node (problem) to its associated successive nodes (subproblems). For an OR arc, the problem is solved when the immediately successive subproblem is solved. If the problem is linked by an AND arc, the problem is only solved when all the successive subproblems are solved. Hence, the solution for the original problem is to search for a tree that connects the root node with terminal nodes only [28].

Zhang and Kuo [23] had extended this searching algorithm in finding a solution tree from an AND/OR graph to obtain the optimal assembly sequence toward the final product. Even with the assumption of reversible operation, the sequence obtained from the solution tree might not be directly used for generating disassembly sequence if partial disassembly is allowed. The main reason is that the disassembly level needs to be identified if full disassembly is not desirable. The selection in a proper set of terminal nodes can be combinatorial [25]. Considering the example as in Figures 4(a) and 4(b), the searching algorithm can be used to find the best trees for the assembly operations from leaf nodes to the root node as in Figure 5. But to retrieve a stick from the returned problem, the optimal sequences for partial disassembly operations $(\{1\}, \{8\})$ or $(\{4\}, \{7\})$ can only be obtained if and only if the set of terminal nodes selected are $\{7, 9, 10\}$ or $\{6, 10, 12\}$, respectively.

3.3. Label Correcting Algorithm

Here we propose a label correcting algorithm to find an optimal disassembly sequence from an AND/OR graph. This algorithm maintains a label $L_i = [d_i, p_i]$ for each node i, $i \in I$, where d_i is the minimum disassembly cost to retrieve node i from some starting node in I_s, and p_i is the set of immediately predecessor nodes in the shortest path. Let $f(j)$ and $t(j)$ denote the from-node and to-node of some arc j, $j \in J$. Further, $F_i = \{j \in J : f(j) = i\}$ and $R_i = \{j \in J : t(j) = i\}$ define the forward star and reverse star for each node i, $i \in I$. The detailed steps of this algorithm are as follows.

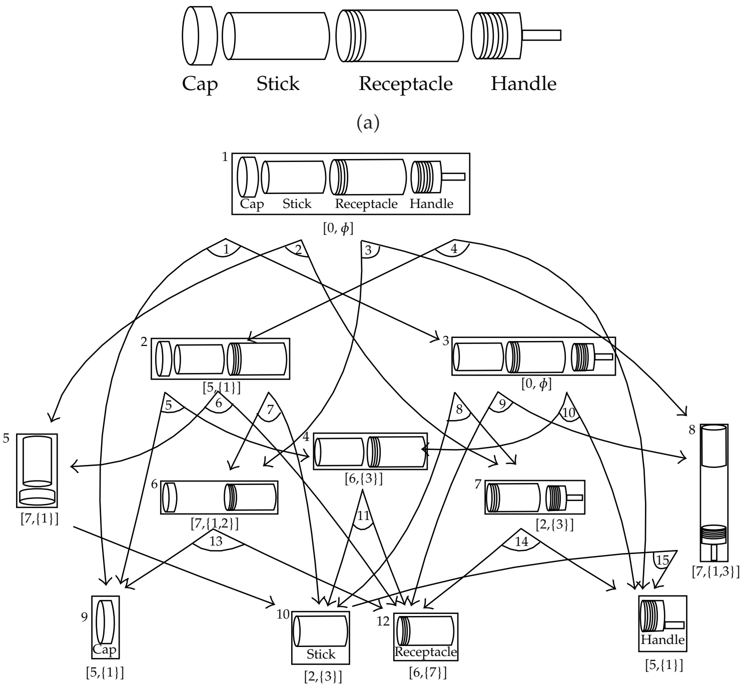
Initialization

For each source node k in $k \in I_s$, set the minimum cost $d_k = 0$, the predecessor set $p_k = \phi$, and update the set of labelled node $L = L \cup \{k\}$. For node k in $I \setminus I_s$, set the minimum cost $d_k = \infty$, the predecessor set $p_k = \phi$. Select the first labelled node k in L.

Step 1. Determine the set S_k for unlabelled nodes that are immediately connected from node k; that is, $S_k = \{n \mid n \in t(j), j \in F_k\} \setminus L$. If S_k is not empty, then go to Step 2. Otherwise, go to Step 3.

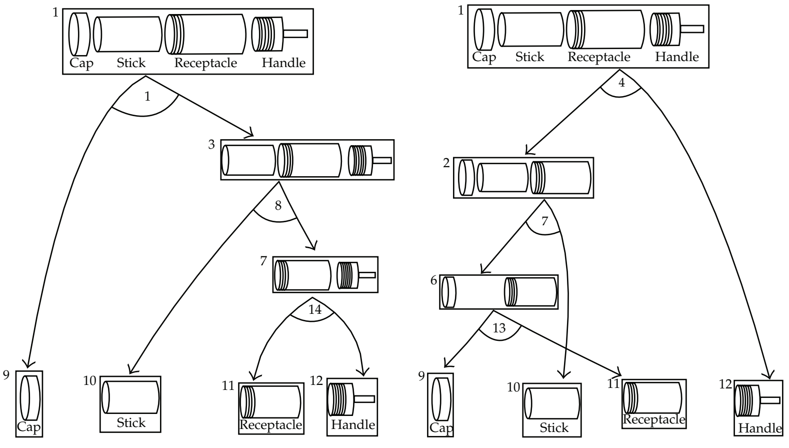
Step 2. For each node i in S_k, do the following

(1) For each arc j where $j \in F_k \cap R_i$, if $d_k + c_j < d_i$, update the minimum cost $d_i = d_k + c_j$ and set $p_i = \{k\}$. Otherwise, if $d_k + c_j = d_i$, then update $p_i = p_i \cup \{k\}$.



(b)

Figure 4: (a) The structure and (b) the AND/OR disassembly tree for a simple product [25].



(a) (b)

Figure 5: Alternated trees of disassembly sequence to remove the stick for the simple product in (a) [25].

(2) Determine the set U_i for unlabelled nodes that are immediately connected toward node i; that is, $U_i = \{n | n \in f(j), j \in R_i\} \setminus L$. If U_i is empty, make node i as labelled and update the set of labelled nodes $L = L \cup \{i\}$.

Step 3. Mark the current node k as solved. Move to the next unsolved node in L as new node k, go to Step 1. If all nodes are solved, stop.

If only original returns can be dismantle for reusable subparts, we have $I_s = I_0$. For any solved node i, the d_i from the label $L_i = [d_i, p_i]$ represents the minimum disassembly cost to retrieve one subpart i directly from a starting assembly. Moreover, the complexity of this procedure is polynomial bound by the number of nodes; that is, $O((|I|)^2)$ [30].

Lemma 3.1. *If only one target subpart has positive demand, the label correcting algorithm solves the SPPDO problem.*

Proof. In the SPPDO formulation, it is assumed that target subparts are retrieved from returned products directly. If only one target subpart i has positive demand, it is equivalent to find an optimal sequence of disassembly operations with the minimum total cost to reach the destination node i from the source node. This minimum cost is also defined as d_i in the label $L_i = [d_i, p_i]$ for node i, $i \in I_f$. □

Lemma 3.2. *If more than one target subpart has positive demands, then the corresponding d_i from the label correcting algorithm for associated nodes forms an upper bound for the optimal solution in the SPPDO problem. That is, suppose y_j^*, $j \in J$, be the optimal solution for the SPPDO problem, then one has $\sum_{j\in J} c_j \cdot y_j^* \leq \sum_{i\in I_f} b_i \cdot d_i$.*

Proof. Suppose two target subparts, say l and m, $l \neq m$, have positive demands. Let $P_l = \{j_{l1}, j_{l2}, \ldots, j_{lp}\}$ and $P_m = \{j_{m1}, j_{m2}, \ldots, j_{mp}\}$ be the optimal disassembly sequences for node l and node m, respectively. Without loss of generality, we assume that $b_l \geq b_m > 0$. Suppose there exists some node $k \in t_j$, $j \in P_l$ and also $k \in f_j$, $j \in P_m$. Then the shortest disassembly path from the source node to node k is overlapped in the optimal disassembly sequences from the source node to node l and node m. Hence, an upper bound for the optimal disassembly cost is $\sum_{j\in J} c_j \cdot y_j^* \leq (b_l \cdot d_l + b_m \cdot d_m) - b_m \cdot d_k$. □

Moreover, it can be concluded that $\sum_{j\in J} c_j \cdot y_j^* = \sum_{i\in I_f} b_i \cdot d_i$ if and only if each original return can be retrieved for no more than one target subpart only.

3.4. A Heuristic Procedure for Solving Partial Disassembly Optimization Problem

Next, we presented heuristic procedure that utilizes the label correcting algorithm in the previous section. This procedure is to find a good solution for larger instances of SPPDO problem in a real-world setting within a reasonable computation effort. The detail of this iterative procedure is described as follows.

Initialization

Apply the label correcting algorithm with $I_s = I_o$ to obtain the initial labels $L_i = [d_i, p_i]$ for all subparts i, $i \in I$. Set the variable $y_j = 0$, $j \in J$. Let x_i be the quantity of subassembly i available for further dismantle, set $x_i = 0$, $i \in I$, and $x_i = |b_i|$, $i \in I_0$. Note that $\sum_{i\in I_0} |b_i| \geq \sum_{i\in I_f} b_i$.

Phase 1 (Path construction). Select a target node i which has the maximum total potential cost in unfulfilled demand; that is, $i = \arg_{k\in I_f} \max\{(b_k - x_k) \cdot d_k \mid b_k > x_k\}$. Break ties arbitrarily.

Obtain the minimum cost disassembly sequences, $P_i = \{j_{i1}, j_{i2}, \ldots, j_{ip}\}$, which forms a directed path from source node s, $s = f(j_{i1})$ toward the demand node i; $i \in t(j_{ip})$. Break ties arbitrarily. Find the maximum flow for this path, $\Delta = \min\{x_s, (b_i - x_i)\}$.

For each arc j, $j \in P_i$, start from the first arc, perform the following updates sequentially.

$$\begin{aligned} y_j &= y_j + \Delta, \\ x_k &= x_k - \Delta, \quad \text{if } k \in f(j), \\ x_k &= x_k + \Delta, \quad \text{if } k \in t(j). \end{aligned} \tag{3.5}$$

Phase 2 (Termination test). Update the total disassembly cost $z = z + \Delta \cdot d_i$ and check for any unfulfilled demand; that is, $x_k < b_k$, $k \in I_f$. If all demands are fulfilled, then stop. The current solution of y_j, $j \in J$, is feasible for the SPPDO problem.

Otherwise, update the set of source nodes to include intermediate nodes with positive quantity; that is, $I_s = I_0 \cup \{k\}$, if $x_k > 0$ for $k \in I \setminus \{I_0 \cup I_f\}$. Update labels $L_i = [d_i, p_i]$ for all subparts i, $i \in I$ using the label correcting algorithm in Section 3.3 with the new set of source nodes I_s. Go to Phase 1.

Lemma 3.3. *The objective value of solution obtained from this heuristic procedure, referred to as z_H, is at least as good as that obtained from the label correcting algorithm, referred to as z_{LC}.*

Proof. In Phase 2, the labels for subparts are altered only when there exists some intermediate subassembly with positive quantity to be further dismantled for those target subparts with lower costs. Otherwise, the initial labels remain unchanged. So, the contribution of disassembly cost for a target subpart in z_H is no higher than that in z_{LC}. □

Furthermore, let $z^* = \sum_{j \in J} c_j \cdot y_j$ be the optimal solution for the SPPDO problem and $z_{LC} = \sum_{i \in I_f} b_i \cdot d_i$. We have $z^* \leq z_H \leq z_{LC}$ from Lemmas 3.2 and 3.3.

4. Numerical Examples

In this section, the simple product in Figure 4(a) is used to demonstrate how the heuristic procedure in Section 3.4 works to generate partial disassembly sequences for the SPPDO problem. There are totally twelve different (sub)assemblies, $I = \{1, \ldots, 12\}$. The original return is represented by node $\{1\}$, and reusable subparts are nodes $\{9\}, \{10\}, \{11\}, \{12\}$ for cap, stick, receptacle, and handle, respectively, that is, $I_0 = \{1\}$ and $I_f = \{9, 10, 11, 12\}$. A total of fifteen disassembly operations can be used to dismantle this product, $J = \{1, \ldots, 15\}$ with the associated costs $C = \{5, 7, 7, 5, 6, 7, 2, 2, 7, 6, 2, 1, 4, 4, 1\}$.

The construction of labels using the label correcting algorithm is shown in Figure 6. The minimum disassembly cost to retrieve node $\{10\}$ (stick) is 7 with two alternative optimal disassembly sequences: $P_{10} = \{4, 7\}$ or $P_{10} = \{1, 8\}$. This optimal solution is consistent with observation in Section 3.2.

Next, we demonstrate the use of heuristic algorithm with the consideration of the following demands for target subparts: three caps (node$\{9\}$) and one stick (node$\{11\}$); that is, $b_9 = 3$. $b_{11} = 1$. It implies that the supply of original returns should be at least four; that is, $b_1 = -4$. In the initialization step, all variables $\mathbf{x}, \mathbf{y}$ are set to zero except $x_1 = 4$ and the initial labels are obtained from Figure 6. In Phase 1, since node $\{9\}$ (cap) has a higher total unfulfilled cost than node $\{11\}$ (stick), node $\{9\}$ is selected along with the associated directed

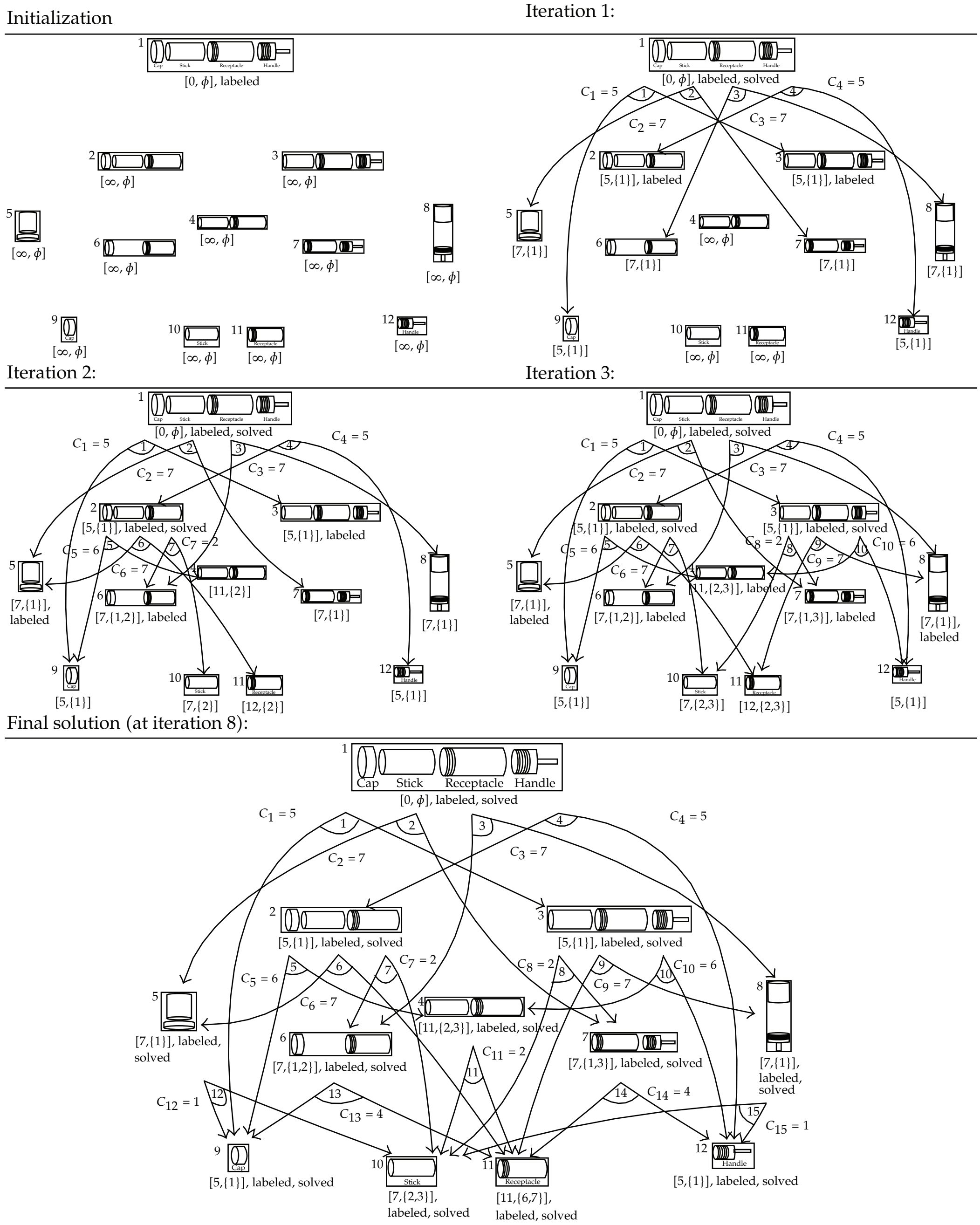


Figure 6: Brief illustration of stepwise label correcting algorithm for the product structure as in Figure 5(a).

path $P_9 = \{1\}$. Calculate the flow $\Delta = \min\{x_1, (b_9 - x_9)\} = \min\{4, (3 - 0)\} = 3$ on this path. Since P_9 has only one arc $j = 1$, update the solution $y_1 = 3$, $x_1 = 4 - 3 = 1$, $x_3 = 3$, and $x_9 = 3$.

Continue to Phase 2 to check for the termination criteria. Update the current total disassembly cost $z = z + \Delta \cdot d_9 = 0 + 3 * 5 = 15$. There still exists unfulfilled demand since $x_{11} = 0 < b_{11}$. So, first update the set of source nodes $I_s = \{1, 3\}$ to include the intermediate node $\{3\}$ as $x_3 = 3 > 0$. Then labels for subparts are updated using the label correcting

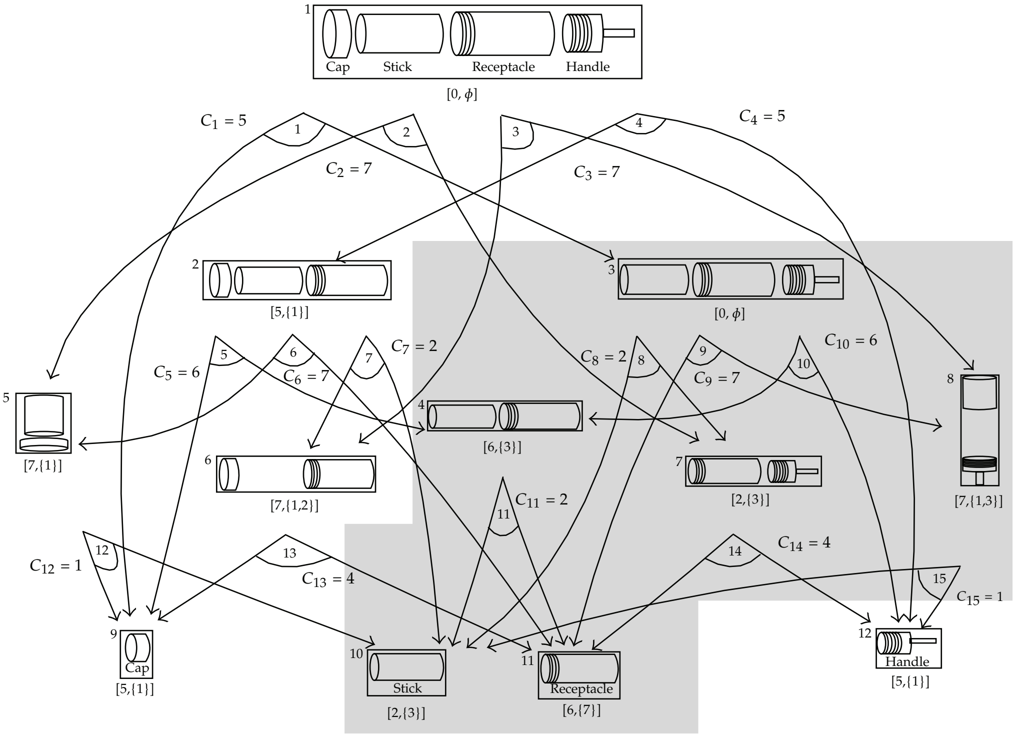


Figure 7: Updated labels with an additional source node {3} as in the shaded area.

algorithm. It is worth noting that, as shown in Figure 7, not every label is changed, and the effected nodes with associated arcs are shown in bold in the shaded area.

In the second iteration, select node {11} to fulfil the remaining demand $b_{11} = 1$ with the disassembly path $P_{11} = \{8, 14\}$. Calculate the maximum flow $\Delta = \min\{x_3, (b_{11} - x_{11})\} = \min\{3, (1–0)\} = 1$ and update the solution starting from the first arc $j = 8$: $y_8 = 1$, $x_3 = 3–1 = 2$, $x_7 = 1$, and $x_{10} = 1$. Then update for the second arc $j = 14$: $y_{14} = 1$, $x_7 = 1 - 1 = 0$, $x_{11} = 1$, and $x_{12} = 1$. This procedure terminates since all demands are fulfilled with the total cost $z = 15 + 1 * 6 = 21$. The resulting quantities of subparts are $x_1 = 1$, $x_3 = 2$, $x_9 = 3$, $x_{11} = 1$, and $x_{12} = 1$, and the required disassembly operations are $y_1 = 3$, $y_8 = 1$ and $y_{14} = 1$ for three caps and one stick.

5. Conclusions

In this paper, we investigate a single period partial disassembly optimization (SPPDO) problem to generate an optimal disassembly sequence in product recovery of the end-of-life (EoL) products. An AND/OR graph representation and associated transition matrix are used in the mathematical formulation of the SPPDO problem to minimize the total disassembly cost. Since the transition matrix does not have the property of total unimodularity, this SPPDO model can only be solved as a pure integer programming (IP) problem, which is NPcomplete.

A label correcting algorithm is proposed to find an optimal disassembly sequence when the reusable subpart is retrieved directly from original return. To solve the SPPDO problem in general case, this paper presents a heuristic procedure that utilizes this polynomial-time algorithm to find a solution. This heuristic procedure can quickly provide a good disassembly plan for problems with more complicated disassembly structures in a real-world setting within a reasonable computation effort. It can be further integrated in the production planning for end-of-life (EoL) products to improve the profitability of product recovery.

Acknowledgment

This research was supported by the National Science Council of Taiwan, China, under Grant no. NSC 98-2218-E-027-019.

References

[1] H. Griese, H. Poetter, K. Schischke, O. Ness, and H. Reichl, "Reuse and lifetime extension strategies in the context of technology innovations, global markets, and environmental legislation," in *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 173–178, Washington, DC, USA, May 2004.

[2] E. J. Grenchus, R. A. Keene, and C. R. Nobs, "A decade of processing end of life IT equipment—lessons learned at the IBM endicott asset recovery center," in *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 195–198, Washington, DC, USA, May 2004.

[3] M. Fleischmann, *Quantitative Models for Reverse Logistics*, Springer, New York, NY, USA, 2001.

[4] E. Grenchus, S. Johnson, and D. McDonnell, "Improving environmental performance through reverse logistics at IBM," in *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 236–240, May 2001.

[5] V. D. R. Guide Jr., "Production planning and control for remanufacturing: industry practice and research needs," *Journal of Operations Management*, vol. 18, no. 4, pp. 467–483, 2000.

[6] A. Kasmara, M. Muraki, S. Matsuoka, A. Sukoyo, and K. Suryadi, "Production planning in remanufacturing/ manufacturing production system," in *Proceedings of the 2nd International Symposium on Environmentally Conscious Design and Inverse Manufacturing*, pp. 708–713, 2001.

[7] A. J. Clegg, D. J. Williams, and R. Uzsoy, "Production planning for companies with remanufacturing capability," in *Proceedings of the IEEE International Symposium on Electronics and the Environment (ISEE '95)*, pp. 186–191, May 1995.

[8] V. D. R. Guide Jr., "Scheduling using drum-buffer-rope in a remanufacturing environment," *International Journal of Production Research*, vol. 34, no. 4, pp. 1081–1091, 1996.

[9] V. D. R. Guide Jr. and M. S. Spencer, "Rough-cut capacity planning for remanufacturing firms," *Production Planning and Control*, vol. 8, no. 3, pp. 237–244, 1997.

[10] V. Daniel, V. D. R. Guide Jr., and R. Srivastava, "An evaluation of order release strategies in a remanufacturing environment," *Computers and Operations Research*, vol. 24, no. 1, pp. 37–47, 1997.

[11] V. D. R. Guide Jr., R. Srivastava, and M. E. Kraus, "Product structure complexity and scheduling of operations in recoverable manufacturing," *International Journal of Production Research*, vol. 35, no. 11, pp. 3179–3199, 1997.

[12] V. D. R. Guide Jr., M. E. Kraus, and R. Srivastava, "Scheduling policies for remanufacturing," *International Journal of Production Economics*, vol. 48, no. 2, pp. 187–204, 1997.

[13] M. E. Ketzenberg, G. C. Souza, and V. D. R. Guide Jr., "Mixed assembly and disassembly operations for remanufacturing," *Production and Operations Management*, vol. 12, no. 3, pp. 320–335, 2003.

[14] P. Dewhurst, "Product design for manufacture: design for disassembly," *Industrial Engineering*, vol. 25, no. 9, pp. 26–28, 1993.

[15] K. N. Taleb and S. M. Gupta, "Disassembly of multiple product structures," *Computers and Industrial Engineering*, vol. 32, no. 4, pp. 949–961, 1997.

[16] I. M. Langella, "Heuristics for demand-driven disassembly planning," *Computers and Operations Research*, vol. 34, no. 2, pp. 552–577, 2007.

[17] A. Gungor and S. M. Gupta, "An evaluation methodology for disassembly processes," *Computers and Industrial Engineering*, vol. 33, no. 1-2, pp. 329–332, 1997.

[18] D. Navin-Chandra, "The recovery problem in product design," *Journal of Engineering Design*, vol. 5, no. 1, pp. 65–86, 1994.

[19] S. C. Sarin, H. D. Sherali, and A. Bhootra, “A precedence-constrained asymmetric traveling salesman model for disassembly optimization,” *IIE Transactions*, vol. 38, no. 3, pp. 223–237, 2006.

[20] M. R. Johnson and M. H. Wang, “Economical evaluation of disassembly operations for recycling, remanufacturing and reuse,” *International Journal of Production Research*, vol. 36, no. 12, pp. 3227–3252, 1998.

[21] A. J. D. Lambert, “Optimizing disassembly processes subjected to sequence-dependent cost,” *Computers and Operations Research*, vol. 34, no. 2, pp. 536–551, 2007.

[22] T. L. De Fazio and D. E. Whitney, “Simplified generation of all mechanical assembly sequences,” *IEEE Journal of Robotics and Automation*, vol. 3, no. 6, pp. 640–658, 1987.

[23] H. C. Zhang and T. C. Kuo, “Graph-based approach to disassembly model for end-of-life product recycling,” in *Proceedings of the IEEE/CPMT 19th International Electronics Manufacturing Technology Symposium*, pp. 247–254, October 1996.

[24] T. C. Kuo, “Disassembly sequence and cost analysis for electromechanical products,” *Robotics and Computer-Integrated Manufacturing*, vol. 16, no. 1, pp. 43–54, 2000.

[25] L. S. Homem de Mello and A. C. Sanderson, “AND/OR graph representation of assembly plans,” *IEEE Transactions on Robotics and Automation*, vol. 6, no. 2, pp. 188–199, 1990.

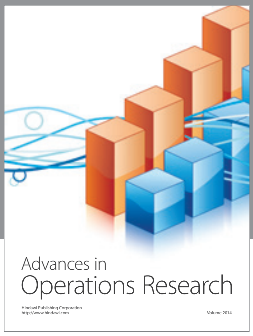
[26] L. S. Homem de Mello and A. C. Sanderson, “A correct and complete algorithm for the generation of mechanical assembly sequences,” *IEEE Transactions on Robotics and Automation*, vol. 7, no. 2, pp. 228–240, 1991.

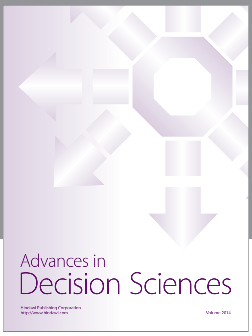
[27] A. J. D. Lambert, “Exact methods in optimum disassembly sequence search for problems subject to sequence dependent costs,” *Omega*, vol. 34, no. 6, pp. 538–549, 2006.

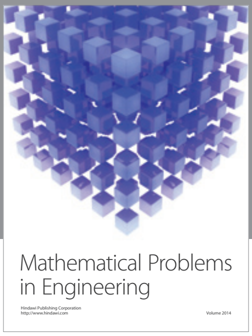
[28] C. L. Chang and J. R. Slagle, “An admissible and optimal algorithm for searching AND/OR graphs,” *Artificial Intelligence*, vol. 2, no. 2, pp. 117–128, 1971.

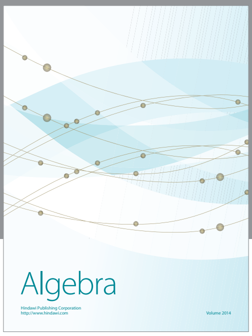
[29] G. Levi and F. Sirovich, “Generalized and/or graphs,” *Artificial Intelligence*, vol. 7, no. 3, pp. 243–259, 1976.

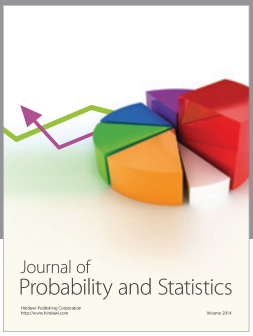
[30] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.











Hindawi

Submit your manuscripts at
http://www.hindawi.com