# A Heuristic for Service-Parts Lot-Sizing with Disassembly Option

Jin-Myeong Jang* · Hwa-Joong Kim*[†] · Dong-Hoon Son** · Dong-Ho Lee***

*Graduate School Logistics, Inha University
**Department of Civil Engineering, The Hong Kong University of Science and Technology
***Department of Industrial Engineering, Hanyang University, Korea

# 분해옵션 포함 서비스부품 로트사이징 휴리스틱

장진명* · 김화중*[†] · 손동훈** · 이동호***

*인하대학교 물류전문대학원
**홍콩과학기술대학교 토목환경공학과
***한양대학교 산업공학과

Due to increasing awareness on the treatment of end-of-use/life products, disassembly has been a fast-growing research area of interest for many researchers over recent decades. This paper introduces a novel lot-sizing problem that has not been studied in the literature, which is the service-parts lot-sizing with disassembly option. The disassembly option implies that the demands of service parts can be fulfilled by newly manufactured parts, but also by disassembled parts. The disassembled parts are the ones recovered after the disassembly of end-of-use/life products. The objective of the considered problem is to maximize the total profit, i.e., the revenue of selling the service parts minus the total cost of the fixed setup, production, disassembly, inventory holding, and disposal over a planning horizon. This paper proves that the single-period version of the considered problem is NP-hard and suggests a heuristic by combining a simulated annealing algorithm and a linear-programming relaxation. Computational experiment results show that the heuristic generates near-optimal solutions within reasonable computation time, which implies that the heuristic is a viable optimization tool for the service parts inventory management. In addition, sensitivity analyses indicate that deciding an appropriate price of disassembled parts and an appropriate collection amount of EOLs are very important for sustainable service parts systems.

Keywords : Service Parts Logistics, Reverse Supply Chain, Lot-Sizing, Disassembly, Simulated Annealing
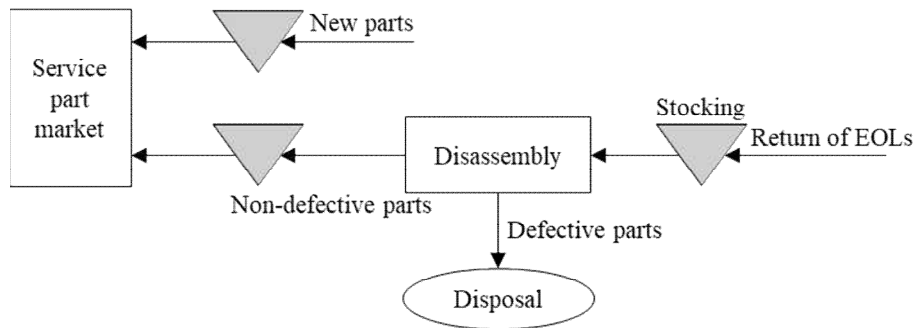
## 1. Introduction

Due to increasing awareness on the treatment of end-of-use/life products (EOLs), disassembly has been a fast-growing research area of interest for many researchers over recent

decades. Disassembly is the environmental process of retrieving and recovering valuable parts or components from EOLs that are returned from end users. Due to its importance, major car makers such as BMW and Volkswagen operate sustainable disassembly processes within their companies. IBM also operates the disassembly process to obtain service parts through disassembling returned EOLs such as PCs, large computers, network servers, and printers [3].
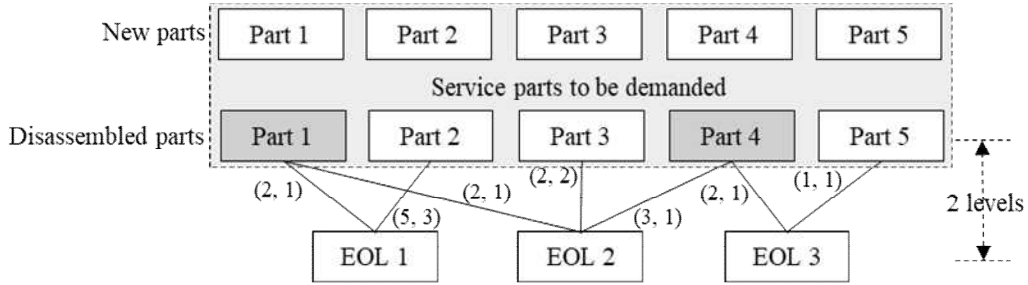
<Figure 1> Service Parts Inventory System with Disassembly Option

This paper addresses a service-parts lot-sizing problem with the disassembly option (SLSD). Lot-sizing is a mid-term production planning to determine production quantity or lot size and production timing over a planning horizon. In the SLSD, the demand for the service parts can be satisfied by newly manufactured parts, but also by disassembled parts. The disassembled parts are the ones that are recovered through recovery operations such as test and upgrade operations after the disassembly of EOLs, as depicted in <Figure 1>. Returned EOLs are kept in inventory, valuable (non-defective) parts are retrieved through disassembly and recovery operations, while defective parts are disposed of. The SLSD was motivated from Fleischmann et al. [3] that addressed the IBM cases. Research areas closely related to the SLSD are disassembly lot-sizing and lot-sizing with remanufacturing option. The two problems are the special case of the SLSD in the problem setting.

The literature on the disassembly lot-sizing, called as disassembly scheduling in the literature, has been growing in recent decades. From a modelling point of view, the key difference from the conventional lot-sizing is the existence of the multiple demand source of parts taken from the disassembly of EOLs. This characteristic of the multiple demand source in the problem is called the divergence characteristic, i.e., a divergence of EOLs into multiple parts/components [14]. The literature on the disassembly lot-sizing can be classified by the existence of parts commonality implying that a product and subassembly shares parts or components. For the case without parts commonality, Gupta and Taleb [8] suggested a reverse MRP procedure. Lee and Xirouchakis [18] proposed a two-phase heuristic for the objective of minimizing the costs related to disassembly operations. Kim et al. [13] suggested a polynomial algorithm for the two-level disassembly problem. Kim et al. [16] proved that the multi-level disassembly lot-sizing problem is NP-hard and sug-

gested a branch and bound algorithm. For the case with parts commonality, Taleb and Gupta [28] and Taleb et al. [29] proposed reverse MRP heuristic. Neuendorf et al. [23] proposed a Petri-net based heuristic. Kim et al. [17] suggested a heuristic algorithm based on a linear programming relaxation. Lee et al. [21] suggested integer programs which are the reverse version of the multi-level lot-sizing model. Kim et al. [15] proposed a two-phase heuristic by extending the algorithm of Kim et al. [17]. Langella [19] extended the heuristic of Taleb and Gupta [28] by considering a purchase quantity limit and disposal options. Finally, several literatures considered the resource capacity restrictions [9, 10, 12, 18, 22] and random demand [11], and recently extended the problem to economic order quantity models, namely economic disassembly quantity models [5, 6, 27].

Along with the disassembly lot-sizing, the literature on the lot-sizing with remanufacturing option has been also growing in recent decades. Remanufacturing is another environmental process of recovering EOLs almost up to the level of newly produced products. The key characteristics of the lot-sizing with remanufacturing option is that the demand can be satisfied by producing new products and/or by remanufacturing EOLs. Golany et al. [7] proved the NP-hardness of the problem with general concave costs and solved the problem as a dynamic programming (DP). Teunter et al. [30] considered two different setup-cost cases: a joint setup cost for manufacturing and remanufacturing; and separated setup costs. They suggested a polynomial-time DP algorithm for the problem with a time-invariant joint-setup cost and has shown that the problem with separated setup costs is NP-hard. Yang et al. [31] also proved that the problem with concave costs is NP-hard. Pan et al. [24] has shown that the problem with disposal or remanufacturing is convertible to a capacitated lot-sizing problem and polynomially solvable if the capacity is constant. Pineyro and Viera [25] suggested

New parts

| Part 1 | Part 2 | Part 3 | Part 4 | Part 5 |

Service parts to be demanded

Disassembled parts

| Part 1 | Part 2 | Part 3 | Part 4 | Part 5 |

(2, 1)   (2, 1)   (2, 2)   (2, 1)   (1, 1)

(5, 3)          (3, 1)

EOL 1        EOL 2        EOL 3

2 levels

<Figure 2> Two-Level Disassembly Structure and Service Parts

a Tabu-search heuristic for the problem with one-way substitution implying that the demand for remanufactured products can be fulfilled by new products, but not vice versa. Heuristics or reformulations have been suggested by Baki et al. [2], Sifaleras and Konstantara [26], and Attila et al. [1].

In this paper, we aim to contribute to the literature by addressing the SLSD that no previous literatures have studied to the best of our knowledge. Specifically, we consider the SLSD with the two-level disassembly structure and parts commonality. The two-level structure defined in <Figure 2> implies that there is no intermediate subassembly between EOLs and parts in the disassembly product structure. From a modeling point of view, the lot-sizing with remanufacturing option is a special case of the SLSD, and hence SLSD is also NP-hard according to the above literature review. However, we prove in this paper that even the single-period problem of the SLSD is NP-hard. In addition, we suggest an efficient heuristic by combining a simulated annealing algorithm and a linear-programming relaxation approach. Finally, we observe some meaningful insights through sensitivity analyses.

In the next section, we present first a detailed explanation of the problem and then an integer programming model. In Sections 3, we present the NP-hardness of the problem and the solution algorithm. We then present the results of computational experiments in Section 4, which evaluates the performance of the suggested heuristic and sensitivity to input parameters. Finally, we conclude with some key remarks and potential future studies.

## 2. Problem description

In this section, we first introduce the disassembly product structure and define the SDSL. We then provide the integer programming model.

The SLSD is based on a two-level disassembly product structure, in which the first level represents EOLs to be disassembled and the second level does a set of parts obtained from disassembling the EOLs. This means that there are no intermediate subassemblies in the structure. The two-level disassembly product structure has been considered in the literature [5, 6, 11, 13, 19, 27]. <Figure 2> shows an example of the two-level disassembly product structure. In the figure, three EOLs are disassembled into five different parts 1, 2, 3, 4, and 5, i.e., EOL 1 is disassembled to parts 1, 2, and EOL 3 to parts 4, 5. In parentheses $(a, b)$, $a$ represent the total unit obtained from disassembling one unit of an EOL if no defectives exist, and $b$ represents non-defective units out of $a$. For example, EOL 2 is disassembled into two units of part 1 as its total unit and one unit as its non-defective unit. Defective parts are sorted out and disposed of by paying disposal costs. It is assumed that non-defective parts are fully recovered up to the resalable state through remanufacturing processes. The shaded parts are common items, i.e., part 4 can be obtained from disassembling EOL 1 or 4. It is also assumed that the same parts are substitutable for each other with different prices when fulfilling the demand of that part. That is, part 2 in the set of new parts and part 2 in the set of disassembled parts are the same parts and hence they can be used to meet the demand of part 2.

Now, the SLSD is formally defined as follows: for a given two-level disassembly structure, the problem is to determine the quantity and period of producing new service parts and disassembling parts from returned EOLs. The objective is to maximize the total profit, i.e., the revenue of selling the service parts minus the cost of production setup, disassembly setup, production, disassembly, inventory holding, and disposal generated over a planning horizon. The revenue is gained from selling the new and disassembled service parts. There are two types of setup costs: production setup and disassembly setup costs. The production (disassembly) setup

cost is a fixed cost incurred in a period if any production (disassembly) is performed in that period. The production cost is the cost of producing new service parts varying with the production quantity while the disassembly cost is the variable cost of disassembling EOLs and recovering parts. The inventory holding cost occurs if parts or EOLs are stocked to satisfy future demands and they are counted based on the end-of-period stock level. The disposal cost is the cost incurred for disposing of defects leftover from a disassembly process.

The notations used in the integer program are summarized below.

*Set*

$\Omega_i$    set of parent EOLs of part $i$

$E$    set of EOLs

$P$    set of service parts

$T$    set of periods

*Parameters*

$a_{ei}$    total number of units of part $i$ taken from disassembling one unit of EOL $e$

$b_{ei}$    non-defectives out of $a_{ei}$

$c_i^N$    unit production cost of new service part $i$

$c_e^D$    unit disassembly cost of EOL $e$

$d_{it}$    demand of service part $i$ in period $t$

$h_i^N$    unit inventory holding cost of new service part $i$

$h_i^D$    unit inventory holding cost of disassembled service part $i$

$h_e^R$    unit inventory holding cost of EOL $e$

$M$    arbitrarily large number

$p_i^N$    unit revenue of new service part $i$

$p_i^D$    unit revenue of disassembled service part $i$

$r_{et}$    returned amount of EOL $e$ in period $t$

$s_i^N$    production setup cost of new service part $i$

$s_e^D$    disassembly setup cost of EOL $e$

$w_i$    unit disposal cost of part $i$

*Decision variables*

$D_{it}$    sales amount of disassembled service part $i$ in period $t$

$I_{it}^N$    inventory level of new service part $i$ at the end of period $t$

$I_{it}^D$    inventory level of disassembled service part $i$ at the end of period $t$

$I_{et}^R$    inventory level of EOL $e$ at the end of period $t$

$N_{it}$    sales amount of new service part $i$ in period $t$

$W_{it}$    disposed amount of part $i$ in period $t$

$X_{it}^N$    production amount of new service part $i$ in period $t$

$X_{et}^D$    disassembly amount of EOL $e$ in period $t$

$Y_{it}^N$    equals 1 if any production of new service part $i$ occurs in period $t$ and zero, otherwise

$Y_{et}^D$    equals 1 if any disassembly of EOL $e$ occurs in period $t$ and zero, otherwise

The integer programming model for the SLSD is as follows.

[P] Maximize

$$\sum_{i\in P}\sum_{t\in T}p_i^N N_{it} + \sum_{i\in P}\sum_{t\in T}p_i^D D_{it} - \sum_{i\in P}\sum_{t\in T}s_i^N Y_{it}^N - \sum_{e\in E}\sum_{t\in T}s_e^D Y_{et}^D$$

$$-\sum_{i\in P}\sum_{t\in T}c_i^N X_{it}^N - \sum_{e\in E}\sum_{t\in T}c_e^D X_{et}^D - \sum_{i\in P}\sum_{t\in T}h_i^N I_{it}^N$$

$$-\sum_{i\in P}\sum_{t\in T}h_i^D I_{it}^D - \sum_{e\in E}\sum_{t\in T}h_e^R I_{et}^R - \sum_{i\in P\cup W}\sum_{t\in T}w_i W_{it}$$

subject to

$$N_{it}+D_{it}=d_{it} \qquad\qquad \forall i\in P, t\in T \qquad (1)$$

$$I_{it}^N = I_{i,t-1}^N + X_{it}^N - N_{it} \qquad \forall i\in P, t\in T \qquad (2)$$

$$I_{it}^D = I_{i,t-1}^D + \sum_{e\in\Omega_i}b_{ei}X_{et}^D - D_{it} \qquad \forall i\in P, t\in T \qquad (3)$$

$$I_{et}^R = I_{e,t-1}^R + r_{et} - X_{et}^D \qquad \forall e\in E, t\in T \qquad (4)$$

$$X_{it}^N \le M\cdot Y_{it}^N \qquad\qquad \forall i\in P, t\in T \qquad (5)$$

$$X_{et}^D \le M\cdot Y_{et}^D \qquad\qquad \forall e\in E, t\in T \qquad (6)$$

$$W_{it} = \sum_{e\in\Omega_i}(a_{ei}-b_{ei})X_{et}^D \qquad \forall i\in P, t\in T \qquad (7)$$

$$N_{it}, D_{it}, I_{it}^N, I_{it}^D, X_{it}^N, W_{it} \ge 0 \qquad \forall i\in P, t\in T \qquad (8)$$

$$X_{et}^D \ge 0 \text{ and integer} \qquad \forall e\in E, t\in T \qquad (9)$$

$$Y_{it}^N, Y_{et}^D \in \{0, 1\} \qquad\qquad \forall i\in P, e\in E, t\in T \qquad (10)$$

The objective function denotes maximizing the total profit, i.e., revenues minus setup, production, disassembly operation, inventory holding, and disposal costs. Constraint (1) represents that the demand is fulfilled by selling the new and/or disassembled service parts. Constraints (2)~(4) represent the inventory flow conservation that defines the inventory level at the end of each period of new service parts, disassembled parts, and EOLs, respectively. Constraints (5) and (6) guarantee that a setup cost in a period is incurred if there is any production or disassembly operation at that period. Constraint (7) determines the amount of defective parts. Finally, the other constraints (8)~(10) are the conditions on the decision variables.

## 3. NP-Hardness and Heuristic Algorithm

As described above, the lot-sizing with remanufacturing option from a modeling point of view is a special case of the SLSD, and hence SLSD is also NP-hard. However, we prove here that even the single-period problem of the SLSD is NP-hard.

**Proposition 1.** *The single-period problem of the SLSD [P] is NP-hard.*

**Proof.** The decision version of the integer knapsack problem is NP-Complete [4]. We show that the integer knapsack problem is a special case of problem [P]. Define an instance of problem [P] with EOLs, 1, 2, ⋯ $n$, one part (indexed $n$+1), the number of periods $|T| = 1$ where $|\cdot|$ is the cardinality of set •, and no new service part. The yield rate of part $n$+1 from all EOLs is set to one, and the return amount is infinite. The revenue, the setup cost, the inventory holding cost, and the disposal cost are set to zero. Finally, the rest is the same as problem [P]. For a clear definition, the model of the instance problem is given as follows.

[**I1**] Minimize $\sum_{i=1}^{n} c_i^D X_{i1}^D$

subject to

$$I_{n+1,1}^D = I_{n+1,0}^D + \sum_{j=1}^{n} b_{i,N+1} X_{i1}^D - d_{n+1,1} \qquad (11)$$

$$I_{it}^D \geq 0 \qquad (12)$$

$$X_{it}^D \geq 0 \text{ and integer} \qquad (13)$$

Integrating (11) and (12) results in

[**I2**] Minimize $\sum_{i=1}^{n} c_i^R X_{i1}^D$

subject to $\sum_{j=1}^{n} b_{i,n+1} X_{i1}^D \geq d_{n+1,1} - I_{n+1,0}^D$ and (13).

The model [I2] is clearly the same as the model of the integer knapsack problem. This completes the proof. ■

Since the SLSD is NP-hard, we try to focus on developing an efficient heuristic. The following subsections present an overview of the simulated annealing (SA) algorithm and the method of generating an initial solution and neighborhood solutions.

### 3.1 Overview of the SA Algorithm

An initial solution is generated using the linear programming (LP) relaxation approach in Section 3.2 and it is set to a current solution. Then, the SA algorithm generates a new neighborhood solution of the current solution using the method in Section 3.3. The move to the new neighborhood solution is accepted if the solution is improved. Otherwise, the moving is accepted with a probability controlled by a *temperature*. As in many SA algorithms, our SA algorithm starts with a high initial temperature decreased by a certain rule called the *cooling schedule*. The temperature is kept at the same for some iterations called the *epoch length*. The epoch length $L$ is set to $\alpha|T|(|T|-1)/2$ where $\alpha$ is a parameter. The initial temperature $t_0$ is set as $t_0 = -\Delta/\ln F_0$ where $\Delta$ is the average increase in the objective value obtained after $|T|$ moves and $F_0$ is a parameter to be determined. The temperature at the $k$-th epoch is set as $t_k = \beta \cdot t_{k-1}$ where $\beta$ is a parameter called the *cooling ratio*. The SA algorithm is terminated if the best solution is not updated for a certain iteration $Q$.

### 3.2 Initial Solution Generation

The initial solution for the SA algorithm is generated through the LP relaxation approach. That is, the integer program [P] is solved using a commercial optimization solver like CPLEX after relaxing constraints (5), (6), (9), and (10). The obtained LP solution (real values) is then rounded down. The rounded-down solution is modified so as to obtain a good feasible solution while considering the profit changes.

For the modification, we first define *balance* as follows.

**Definition 1.** Based on the rounded-down solution, the balances $(B_{it}^S, B_{it}^N, B_{it}^D, B_{et})$ are defined as

$$B_{it}^S = N_{it} + D_{it} - d_{it} \qquad \forall i \in P, t \in T$$

$$B_{it}^N = I_{it}^N - I_{i,t-1}^N - X_{it}^N + N_{it} \qquad \forall i \in P, t \in T$$

$$B_{it}^D = I_{it}^D - I_{i,t-1}^D - \sum_{e \in \Omega_i} b_{ei} X_{et}^D + D_{it} \qquad \forall i \in P, t \in T$$

$$B_{et} = I_{et}^R - I_{e,t-1}^R - r_{et} + X_{et}^D \qquad \forall e \in E, t \in T$$

The balances defined above are used to check the feasibility of the rounded-down solution. More specifically, constraints (1)-(4) are feasible if and only if the balances equal zero where the rounded-down solution is used to calculate the balances. If a balance is non-zero, the corresponding

rounded-down solution should be modified. The solution is modified by considering decision variables and the sign of the balance (positive and negative). We explain here only the method for the cases with a positive sign, i.e., $B_{it}^S$, $B_{it}^N$, $B_{it}^D$, $B_{et} > 0$ since negative sign cases can be considered easily by reversing positive sign cases.

**Case 1 : Positive EOL balance case** $B_{et} > 0$. We start from the EOL, i.e., the method for the $B_{et} > 0$ case. In order to satisfy constraint (4), decreasing the values of $X_{et}^D$ and/or $I_{et}^R$ is needed while considering the profit increase defined as

$$
\alpha_{et} = \begin{cases}
c_e^D B_{et} + s_e^D \theta_{et} + \sum_{i \in \Phi_e} (a_{ei} - b_{ei}) w_i B_{et} & \text{if } I_{et}^R \geq B_{et} \\
\quad + \sum_{i \in \Phi_e} (|T| - t + 1) b_{ei} h_i^D B_{et} & \\
c_e^D X_{et}^D + s_e^D + \sum_{i \in \Phi_e} (a_{ei} - b_{ei}) w_i X_{et}^D & \text{otherwise} \\
\quad + \sum_{i \in \Phi_e} (|T| - t + 1) b_{ei} h_i^D X_{et}^D & \\
\quad + (|T| - t + 1)(B_{et} - X_{et}^D) h_e^R &
\end{cases}
$$

$$
\beta_{et} = \begin{cases}
(|T| - t + 1) h_e^R B_{et} & \text{if } I_{et}^R \geq B_{et} \\
(|T| - t + 1) h_e^R I_{et}^R + c_e^D (B_{et} - I_{et}^R) + s_e^D \theta_{et} & \\
\quad + \sum_{i \in \Phi_e} (a_{ei} - b_{ei})(B_{et} - I_{et}^R) w_i & \text{otherwise} \\
\quad + \sum_{i \in \Phi_e} (|T| - t + 1)(B_{et} - I_{et}^R) b_{ei} h_i^D &
\end{cases}
$$

where $\theta_{et} = 1$ if $X_{et}^D = B_{et}$ and 0 otherwise, and $|T|$ is the cardinality of set $T$. Here, $\alpha_{et}$ and $\beta_{et}$ denote the amount of profit increase when $X_{et}^D$ and $I_{et}^R$ are decreased up to $X_{et}^{'D}$ and $I_{et}^{'R}$, respectively, defined as

$$
X_{et}^{'D} = \begin{cases} X_{et}^D - B_{et} & \text{if } X_{et}^D \geq B_{et}, \\ 0 & \text{otherwise} \end{cases}
$$

$$
I_{et}^{'R} = \begin{cases} I_{et}^R - B_{et} & \text{if } I_{et}^R \geq B_{et}. \\ 0 & \text{otherwise} \end{cases}
$$

We select the choice with more profit increase, i.e., $\Delta_{et} = max\{\alpha_{et}, \beta_{et}\}$. The balance $B_{et}$ is updated into new balance $B_{et}^{'}$ defined below, and this routine continues until the balance becomes zero.

$$
B_{et}^{'} = \begin{cases} B_{et} - X_{et}^D & \text{if } X_{et}^D < B_{et} \text{ and } \Delta_{et} = \alpha_{et} \\ B_{et} - I_{et}^D & \text{if } I_{et}^D < B_{et} \text{ and } \Delta_{et} = \beta_{et}, \\ 0 & \text{otherwise} \end{cases}
$$

**Case 2 : Positive disassembled-part balance case** $B_{it}^D > 0$. Like Case 1, we consider decreasing the value of $D_{it}$ and/or $I_{it}^D$ to satisfy constraint (3). The profit increase is defined as

$$
\alpha_{it} = \begin{cases}
p_i^N B_{it}^D - p_i^D B_{it}^D & \text{if } D_{it} \geq B_{it}^D \\
\quad + (|T| - t + 1) h_i^N B_{it}^D & \text{and } I_{it}^N \geq B_{it}^D \\
p_i^N B_{it}^D - p_i^D B_{it}^D + (|T| - t + 1) h_i^N I_{it}^N & \text{if } D_{it} \geq B_{it}^D \\
\quad - c_i^N (B_{it}^D - I_{it}^N) - s_i^N \theta_{it} & \text{and } I_{it}^N < B_{it}^D \\
p_i^N D_{it} - p_i^D D_{it} & \text{if } D_{it} < B_{it}^D \\
\quad + (|T| - t + 1)(h_i^N D_{it} + h_i^D (B_{it}^D - D_{it})) & \text{and } I_{it}^N \geq D_{it} \\
p_i^N D_{it} - p_i^D D_{it} & \\
\quad + (|T| - t + 1)(h_i^N I_{it}^N + h_i^D (B_{it}^D - D_{it})) & \text{otherwise} \\
\quad - s_i^N \theta_{it} - c_i^N (B_{it}^D - D_{it} - I_{it}^N) &
\end{cases}
$$

$$
\beta_{it} = \begin{cases}
(|T| - t + 1) h_i^D B_{it}^D & \text{if } I_{it}^D \geq B_{it}^D \\
(|T| - t + 1)(h_i^D I_{it}^D + h_i^N I_{it}^N) & \\
\quad + (p_i^N - p_i^D)(B_{it}^D - I_{it}^D) & \text{if } I_{it}^D < B_{it}^D \text{ and} \\
\quad - c_i^N (I_{it}^N - (B_{it}^D - I_{it}^D)) - s_i^N \theta_{it} & I_{it}^N < B_{it}^D - I_{it}^D \\
(|T| - t + 1)(h_i^D I_{it}^D + h_i^N (B_{it}^D - I_{it}^D)) & \\
\quad + (p_i^N - p_i^D)(B_{it}^D - I_{it}^D) & \text{otherwise}
\end{cases}
$$

where $\theta_{it} = 1$ if $X_{it}^N = 0$ and 0 otherwise. Here, $\alpha_{it}$ and $\beta_{it}$ denote the amount of profit increase when $D_{it}$ and $I_{it}^D$ are decreased up to $D_{it}^{'}$ and $I_{it}^{'D}$, respectively, defined as

$$
D_{it}^{'} = \begin{cases} D_{it} - B_{it}^D & \text{if } D_{it} \geq B_{it}^D, \\ 0 & \text{otherwise} \end{cases}
$$

$$
I_{it}^{'D} = \begin{cases} I_{it}^D - B_{it}^D & \text{if } I_{it}^D \geq B_{it}^D. \\ 0 & \text{otherwise} \end{cases}
$$

We consider also changing of $I_{it}^N$, $N_{it}$ and $X_{it}^N$ when calculating the profit increase as

$$
I_{it}^{'N} = \begin{cases} I_{it}^N - B_{it}^D & \text{if } I_{it}^N \geq B_{it}^D, \\ 0 & \text{otherwise} \end{cases}
$$

$$
N_{it}^{'} = \begin{cases} N_{it} + D_{it} & \text{if } D_{it} \leq B_{it}^D, \\ N_{it} + B_{it}^D & \text{otherwise} \end{cases}
$$

$$
X_{it}^{'N} = \begin{cases}
X_{it}^N + B_{it}^D - I_{it}^D & \text{if } D_{it} \geq B_{it}^D, I_{it}^N < B_{it}^D \\
X_{it}^N - D_{it} - I_{it}^D & \text{if } D_{it} < B_{it}^D, I_{it}^N < D_{it} \\
X_{it}^N + B_{it}^D - D_{it} - I_{it}^D & \text{if } I_{it}^D < B_{it}^D, I_{it}^N < B_{it}^D - I_{it}^D
\end{cases}
$$

We finally select the choice with more profit increase, $\max \{\alpha_{it}, \beta_{it}\}$.

**Case 3 : Positive demand balance case** $B_{it}^S > 0$**.** To satisfy constraint (1), we consider decreasing the value of $N_{it}$ by $B_{it}^S$, i.e., $N_{it}' = N_{it} - B_{it}^S$ and this results in $-B_{it}^S p_i^N$ of the profit.

**Case 4: Positive new-part balance case** $B_{it}^N > 0$**.** To satisfy constraint (2), we consider increasing the value of $X_{it}^N$ up to $B_{it}^N$, and/or decreasing the value of $I_{it}^N$ and these results in profit increases, respectively as

$$\chi_{it} = -s_i^N \theta_{it} - c_i^N B_{it}^N,$$

$$\delta_{it} = \begin{cases} (|T|-t+1)h_i^N B_{it}^N & \\ (|T|-t+1)h_i^N I_{it}^N & \text{if } I_{it}^N \geq B_{it}^N \\ -s_i^N \theta_{it} - c_i^N (B_{it}^N - I_{it}^N) & \text{otherwise} \end{cases}$$

In each of the cases, the balance $B_{it}^D$, $B_{it}^S$, $B_{it}^N$ are updated in a similar manner as in Case 1 and the routine inside that case continues until the corresponding balance becomes zero.

Now, an overall procedure of the LP relaxation heuristic is presented below.

**Procedure 1 : LP relaxation heuristic.**

**Step 1.** (LP relaxation and solution initialization) Solve the linear programming model after relaxing the constraint (5), (6), (9) and (10) of integer program [P] and obtain the LP solution $N_{it}$, $D_{it}$, $I_{it}^N$, $I_{it}^D$, $I_{et}^R$, $X_{it}^N$, $X_{et}^D$ with real values. Obtain the rounded-down solution by applying the floor function to the LP solution.

**Step 2.** (Solution modification) From $t = 1$ to $|T|$, apply the following steps sequentially:

**2-1.** For all $e \in E$, calculate the balance $B_{et}$ using the rounded-down solution obtained in Step 1. Modify the solution using the solution modification method of Case 1 if $B_{et} \neq 0$.

**2-2.** For all $i \in P$, calculate the balance $B_{it}^D$ and disposal quality $W_{it}$, and modify the solution using the Case-2 method if $B_{it}^D \neq 0$. Next, calculate the balance $B_{it}^S$ and modify the solution using the Case-3 method if $B_{it}^S \neq 0$. Finally, calculate the balance $B_{it}^N$ and modify the solution using the Case-4 method if $B_{it}^N \neq 0$.

After obtaining the solution of the LP relaxation heuristic, the solution of binary variables $Y_{it}^N$ and $Y_{et}^D$ is obtained using the below equations.

$$Y_{it}^N = \begin{cases} 1 & \text{if } X_{it}^N > 0 \\ 0 & otherwise \end{cases} \text{ and } Y_{et}^D = \begin{cases} 1 & \text{if } X_{et}^D > 0 \\ 0 & otherwise \end{cases}$$

## 3.3 Neighborhood Solution Generation

To improve the initial solution (current solution), a neighborhood solution of the current solution is generated. The neighborhood solution is accepted if the solution is improved or with a specified probability as described earlier. A neighborhood solution is generated by three interchanging methods: single-period interchange, single-point interchange, and double-point interchange. The three interchanging methods apply to the setup decisions for new parts and EOLs. The best solution obtained by applying all combinations of the three interchange methods for all new parts and EOLs is selected as a neighborhood solution.

**Single-period interchanging method.** A period is randomly chosen and the setup of the chosen period is cancelled, i.e., $Y_{it}^N = 0$ if $Y_{it}^N = 1$ or $Y_{et}^D = 0$ if $Y_{et}^D = 1$ in the current solution, and vice versa. Then, the following LP model [NP] is directly solved based on modified setup variables and the LP solution of model [NP] is modified into the feasible solution of the original problem [P] using Procedure 1.

[**NP**] Maximize

$$\sum_{i \in P}\sum_{t \in T} p_i^N N_{it} + \sum_{i \in P}\sum_{t \in T} p_i^D D_{it} - \sum_{i \in P}\sum_{t \in T} c_i^N X_{it}^N - \sum_{e \in E}\sum_{t \in T} c_e^D X_{et}^D$$

$$- \sum_{i \in P}\sum_{t \in T} h_i^N I_{it}^N - \sum_{i \in P}\sum_{t \in T} h_i^D I_{it}^D - \sum_{e \in E}\sum_{t \in T} h_e^R I_{et}^R$$

$$- \sum_{i \in P \cup W}\sum_{t \in T} w_i W_{it} - \sum_{i \in P}\sum_{t \in T} s_i^N Y_{it}'^N - \sum_{e \in E}\sum_{t \in T} s_e^D Y_{et}'^D$$

subject to

$$X_{it}^N \leq M \cdot Y_{it}'^N \qquad\qquad \forall i \in P, t \in T$$
$$X_{et}^D \leq M \cdot Y_{et}'^D \qquad\qquad \forall e \in E, t \in T$$
$$X_{et}^D \geq 0 \qquad\qquad\qquad \forall e \in E, t \in T$$

and (1)-(4), (7), (8).

where $Y_{it}'^N$ and $Y_{rt}'^D$ are the setup solution modified by the interchanging method.

**Single-point interchanging method.** It randomly chooses a period and two items, i.e., two new parts, one new part and one EOL, or two EOLs. The setup values in later periods and the chosen period are interchanged between the two chosen items in the current solution. Then, the same method as in the single-period method is applied to obtain a feasible solution.

**Double-point interchanging method.** It randomly chooses two periods and two items, i.e., two new parts, one new part and one EOL, or two EOLs. The setup values in the periods between the chosen two periods are interchanged between the two chosen items in the current solution. Then, the same method as in the single-period method is applied to obtain a feasible solution.

# 4. Computational Experiment

To show the performance of the SA algorithm suggested in our paper, we performed computational tests using randomly generated problems. We used CPLEX 12.8.0, a commercial software package to solve problems for comparing our algorithm suggested in this paper. The algorithm and integer program were coded in C and the tests were conducted on a personal computer with Intel® Core™ i5-8400U, 8GB RAM, 2.80GHz.

We randomly generated 45 test instances, i.e., 5 instances for each of combination of three levels of the number of service parts (10, 20, 30) and three levels of the number of periods (10, 20, 30). The number of EOLs generated from $DU$ (5, 10) and the number of parent EOLs of part $i$ are generated from $DU$ ($|E|/2, |E|$), where $DU$ ($c, d$) is the discrete uniform distribution with [$c, d$]. The total number of units of parts and non-defectives taken from disassembling one unit of an EOL were generated from $DU$ (3, 5) and $DU$ (1, 3), respectively. For the new service part, the unit production cost and inventory holding cost were generated from $DU$ (150, 300) and $DU$ (20, 30), whereas for the disassembled part, the unit disassembly cost and the unit inventory holding cost were generated from $DU$ (75, 125) and

$DU$ (15, 20). The unit revenue for new and disassembled parts were generated from $DU$ (800, 1200) and $DU$ (600, 800). The unit inventory holding cost of EOLs was generated from $DU$ (5, 10) and the unit disposal cost for defective parts was generated from $DU$ (1, 3). The setup costs per the setup of new parts and EOLs were generated as $\bar{c}^N U$ (10, 15) and $\bar{c}^D U$ (1, 5) respectively where $\bar{c}^N$ and $\bar{c}^D$ are the averages of unit production cost of new service parts and unit disassembly cost of EOLs, respectively, and $U$ ($c, d$) is the uniform distribution with [$c, d$]. The demand for service parts was generated from $DU$ (50, 200) and the returned amount of EOLs was generated $DU$ (5, 15). In the SA algorithm, parameters $F_0$, $\alpha$, and $\beta$, $Q$ were set 0.64, 5, 0.99 and 200 respectively after a series of preliminary tests.

The performance of the SA algorithm is evaluated using two performance measures: the percentage gap from the optimal solution value (or lower bound if any optimal solution could not be obtained within the time limit 3600s) and CPU seconds. We set 3600s of time limit for each run of the CPLEX to avoid excessive computation time. The percentage gap was defined as $(O_{CPX} - O_{SA})/O_{CPX} \cdot 100$ where $O_{CPX}$ and $O_{SA}$ are the solution value of CPLEX and the SA algorithm, respectively. The results are summarized in <Table 1>. As can be seen in <Table 1>, the SA algorithm gives near-optimal solutions for all test problems, i.e., all percentage gaps are less than 0.3%. The computation time of the SA algorithm was significantly shorter than that of CPLEX, i.e., the maximum computation time of the SA algorithm and CPLEX are 26 sec. and 3600 sec., respectively. The percentage gap increases along with the number of parts as expected whereas there is no apparent relationship between the gap and the number of periods. The computation time increases along with the number of parts and periods.

<Table 1> Performance of the SA Algorithm

| Number of parts | Number of periods | Percentage gap(%) | | CPU seconds | | | |
| | | | | SA | | CPLEX | |
| | | Average | Std. dev. | Average | Maximum | Average | Maximum |
|---|---|---|---|---|---|---|---|
| 10 | 10 | 0.10 | 0.04 | 3.21 | 3.62 | 2181.65 | 3600 |
| | 20 | 0.10 | 0.05 | 6.34 | 8.74 | 2880.45 | 3600 |
| | 30 | 0.08 | 0.06 | 8.11 | 8.86 | 2176.40 | 3600 |
| 20 | 10 | 0.12 | 0.08 | 5.93 | 6.80 | 2888.95 | 3600 |
| | 20 | 0.11 | 0.05 | 11.01 | 12.19 | 3600 | 3600 |
| | 30 | 0.18 | 0.03 | 15.09 | 17.11 | 3600 | 3600 |
| 30 | 10 | 0.24 | 0.05 | 8.35 | 8.81 | 3600 | 3600 |
| | 20 | 0.21 | 0.06 | 16.04 | 18.81 | 3600 | 3600 |
| | 30 | 0.19 | 0.07 | 21.40 | 26.00 | 3600 | 3600 |

We further performed sensitivity analyses to analyze the effect of changes in the price and the amount of returned EOLs. The analyses were conducted based on the solution of the SA algorithm because the heuristic gave a near-optimal solutions within reasonable time whereas CPLEX required excessive computation time as discussed above. They are also based on the problem with maximum size in the test set in <Table 1> with 30 parts and periods.

Firstly, to analyze the effect of change in the price of the disassembled parts, their prices were varied from 65% to 100% of the new parts' price in increments of 5%, and 5 test problems were generated on the same percentage. Let *price rate* be for simplicity the price of the disassembled part over that of the new part. For this analysis, we obtained the sales share of the disassembled part out of all sales, the amount share of the non-disassembled EOLs out of all EOL returns, and the change rate in the total profit, which is calculated based on a price rate of 80%. The results are summarized in <Table 2>. The sales share of the disassembled part is zero and EOL is not disassembled at a price rate of 65%. Then, as the price rate increases, the sales share increases

and the share of the amount of non-disassembled EOLs decreases, but they are saturated after the price rate of 90%. These results imply that disassembling EOLs and selling disassembled parts are not economical at a very low price of disassembled parts. Although the sales share of disassembled parts increases along with the price rate in general, there is a certain price rate where the sales share of disassembled parts increases no longer and even decreases. Therefore, appropriate pricing of disassembled parts is important to make the disassembly process economical and sustainable. However, the total profit increases as the price rate becomes higher as expected.

Next, the effect of change in the amount of returned EOLs is analyzed by multiplying the current return amount used in the above experiments and multipliers ranging from 1 to 8. 5 test problems were generated on the same multiple. The price rate was set to 90% of the new part's price because sales share of the disassembled part and share of non-disassembled EOL are not changed significantly after that rate. The results are summarized in <Table 3>. The non-disassembled EOLs increase as the multiplier value increases

<Table 2> Effect of Change in the Price of the Disassembled Parts

| Price rate (%) | Sales of disassembled parts[a] (%) | Non-disassembled EOLs[b] (%) | Change in the total profit (%) |
|---|---|---|---|
| 65 | 0.00 | 100.00 | -1.63 |
| 70 | 0.10 | 99.67 | -1.63 |
| 75 | 11.57 | 61.58 | -1.44 |
| 80 | 28.62 | 10.80 | 0.00 |
| 85 | 28.09 | 9.67 | 2.12 |
| 90 | 31.60 | 2.16 | 4.26 |
| 95 | 31.12 | 3.47 | 6.44 |
| 100 | 31.95 | 1.54 | 8.62 |

[a]Sales share of the disassembled part out of all sales.
[b]Share of the amount of non-disassembled EOLs out of all EOL returns.

<Table 3> Effect of Change in the Returned Amount of EOLs

| Multiplier | Sales of disassembled parts[a] (%) | Non-disassembled EOLs[a] (%) | Change in the total profit (%) |
|---|---|---|---|
| 1 | 20.46 | 21.15 | 0.00 |
| 2 | 33.14 | 32.17 | 3.58 |
| 3 | 46.83 | 38.33 | 4.73 |
| 4 | 57.03 | 43.19 | 4.96 |
| 5 | 54.49 | 55.44 | 4.90 |
| 6 | 60.95 | 58.44 | 4.72 |
| 7 | 61.05 | 64.41 | 4.47 |
| 8 | 56.21 | 74.52 | 4.22 |

[a]See the footnote of <Table 2>.

as expected. However, the sales share of the disassembled parts and the total profit sharply increase. After multiplier value 5 until the multiplier value increases from 1 to 4, but they decrease slightly after multiplier value 5. The concavity may be because gathering EOLs a lot may incur excessive inventories. Therefore, like the effect of change in the price of the disassembled parts, deciding an appropriate amount of gathering EOLs is also important to make the disassembly process economical and sustainable.

## 5. Concluding Remarks

This study introduced a novel lot-sizing problem that has not been studied in the literature, named the service-parts lot-sizing with disassembly option. The disassembly option implies that the demands of the service parts can be fulfilled by newly manufactured parts but also by disassembled parts. We proved that the single-period version of the problem is NP-hard and suggested a heuristic by combining simulated annealing and linear-programming relaxation heuristics. Computational experiment results have shown that the heuristic is a time-efficient method of obtaining near-optimal solutions. Through sensitivity analyses, we observed that disassembling EOLs and selling disassembled parts were not economical at the very low price of disassembled service parts; there is a certain price of disassembled parts where no more disassembly is performed; and there is a certain amount of EOLs where gathering EOLs more is not economical anymore due to the inventory holding cost of EOLs. Those results of the sensitivity analyses imply that an appropriate pricing of disassembled parts and an appropriate gathering of EOLs are economical for sustainable service parts systems.

The service parts lot-sizing problem with disassembly option considered in this paper is expected to gain interest from practitioners in various industrial fields. For example, increasing returns of EOL electric vehicle battery has encouraged car manufacturers to retrieve valuable batteries from the EOL electric vehicle. Furthermore, the problem can also motivate electronic device manufacturers to find a cost-efficient after-sales supports, e.g., repairing and warranty service, with the disassembled and newly-procured service parts. Applications of the lot-sizing problem require practitioners to incorporate a variety of real-world operational issues into the problem.

In this sense, this paper can be extended in several different ways. First, we assumed that the quantity and return peri-od of EOLs, as well as the quantity of defective parts, are deterministic. These assumptions are by no means necessary for the presented model and solution algorithm. However, relaxing them results in the stochastic problem which is challenging from a modeling point of view, but meaningful from a practical point of view. Second, the problem can be extended to account for more general cost structures like the literatures for the lot-sizing with remanufacturing option. The problem with general concave costs is also challenging because it cannot be directly solved as an integer program. Third, we analyzed the effects of change in the price of disassembled parts in the experiments. Therefore, it is important and meaningful to determine the optimal price of disassembled parts from a practical point of view. The pricing problem is however challenging since it may be formulated as a non-linear program. Finally, it may be meaningful to address the problem of determining the optimal amount and timing of gathering EOLs by considering the activity costs of gathering EOLs. In this problem, it may be needed to consider the availability of EOLs, which is random in practice.

## Acknowledgement

## References

[1] Attila, O.N., Agra, A., Akartunalı, K., and Arulselvan, A., Robust Formulations for Economic Lot-Sizing Problem with Remanufacturing, *European Journal of Operational Research*, 2021, Vol. 288, No. 2, pp. 496-510.

[2] Baki, M.F., Chaouch, B.A., and Abdul-Kader, W., A Heuristic Solution Procedure for the Dynamic Lot Sizing Problem with Remanufacturing and Product Recovery, *Computers and Operations Research*, 2014, Vol. 43, pp. 225-236.

[3] Fleischmann, M., Van Nunen, J.A., and Grave, B., Integrating Closed-Loop Supply Chains and Spare-Parts Management at IBM, *Interfaces*, 2003, Vol. 33. No. 6, pp. 44-56.

[4] Garey, M.R. and Johnson, D.S., Computers and Intractability : A Guide to the Theory of NP-Completeness, W.H. Freeman & Company, New York, 1979.

[5] Godichaud, M. and Amodeo, L., Economic Order Quantity for Multistage Disassembly Systems, *International Journal of Production Economics*, 2018a, Vol. 199, pp. 16-25.

[6] Godichaud, M. and Amodeo, L., EOQ Inventory Models for Disassembly Systems with Disposal and Lost Sales, *International Journal of Production Research*, 2018b, Vol. 57, No. 18, pp. 5685-5704.

[7] Golany, B., Yang, J. and Gang, Y., Economic Lot-Sizing with Remanufacturing Options, *IIE Transactions*, 2001, Vol. 33, No. 11, pp. 995-1003.

[8] Gupta, S.M. and Taleb, K.N., Scheduling Disassembly, *International Journal of Production Research*, 1994, Vol. 32, No. 8, pp. 1857-1866.

[9] Jeon, H.-B., Kim, J.-G., Kim, H.-J., and Lee, D.-H., A Two-Stage Heuristic for Disassembly Scheduling with Capacity Constraints, *International Journal of Management Science*, 2006, Vol. 12, No. 1, pp. 95-112.

[10] Ji, X., Zhang, Z., Huang, S. and Li, L., Capacitated Disassembly Scheduling with Parts Commonality and Start-Up Cost and its Industrial Application, *International Journal of Production Research*, 2016, Vol. 54, No. 4, pp. 1225-1243.

[11] Kim, H.-J. and Xirouchakis, P., Capacitated Disassembly Scheduling with Random Demand, *International Journal of Production Research*, 2010, Vol. 48, No. 23, pp. 7177-7194.

[12] Kim, H.-J., Lee, D.-H., and Xirouchakis, P., A Lagrangean Heuristic Algorithm for Disassembly Scheduling with Capacity Constraints, *Journal of the Operational Research Society*, 2006b, Vol. 57, pp. 1231-1240.

[13] Kim, H.-J., Lee, D.-H., and Xirouchakis, P., An Exact Algorithm for Two-Level Disassembly Scheduling, *Journal of Korean Institute of Industrial Engineers*, 2008, Vol. 34, No. 4, pp. 414-424.

[14] Kim, H.-J., Lee, D.-H., and Xirouchakis, P., Disassembly scheduling : Literature Review and Future Research Directions, *International Journal of Production Research*, 2007, Vol. 45, pp. 4465-4484.

[15] Kim, H.-J., Lee, D.-H., and Xirouchakis, P., Two-Phase Heuristic for Disassembly Scheduling with Multiple Product Types and Parts Commonality, *International Journal of Production Research*, 2006a, Vol. 44, pp. 195-212.

[16] Kim, H.-J., Lee, D.-H., Xirouchakis, P., and Kwon, O.-K., A Branch and Bound Algorithm for Disassembly Scheduling with Assembly Product Structure, *Journal*

*of the Operational Research Society*, 2009, Vol. 60, No. 3, pp. 419-430.

[17] Kim, H.-J., Lee, D.-H., Xirouchakis, P., and Zust, R., Disassembly Scheduling with Multiple Product Types, *CIRP Annals-Manufacturing Technology*, 2003, Vol. 52, No. 1, pp. 403-406.

[18] Kim, J.-G., Jeon, H.-B., Kim, H.-J., Lee, D.-H., and Xirouchakis, P., Disassembly Scheduling with Capacity Constraints : Minimizing the Number of Products Disassembled, *Proceedings of the Institution of Mechanical Engineers*, 2006c, Vol. 220, pp. 1473-1481.

[19] Langella, I.M., Heuristics for Demand-Driven Disassembly Planning, *Computers and Operations Research*, 2007, Vol. 34, No. 2, pp. 552-577.

[20] Lee, D.-H. and Xirouchakis, P., A Two-Stage Heuristic for Disassembly Scheduling with Assembly Product Structure, *Journal of the Operational Research Society*, 2004, Vol. 55, No. 3, pp. 287-297.

[21] Lee, D.-H., Kim, H.-J., Choi, G., and Xirouchakis, P., Disassembly Scheduling : Integer Programming Models, Proceedings of the Institution of Mechanical Engineers, *Journal of Engineering Manufacture-Part B*, 2004, Vol. 218, No. 10, pp. 1357-1372.

[22] Lee, D.-H., Xirouchakis, P., and Zust, R., Disassembly Scheduling with Capacity Constraints, *CIRP Annals-Manufacturing Technology*, 2002, Vol. 51, No. 1, pp. 387-390.

[23] Neuendorf, K.-P., Lee, D.-H., Kiritsis, D., and Xirouchakis, P., Disassembly Scheduling with Parts Commonality Using Petri Nets with timestamps, *Fundamenta, Informaticae*, 2001, Vol. 47, pp. 295-306.

[24] Pan, Z., Tang, J., and Liu, O., Capacitated Dynamic Lot Sizing Problems in Closed-Loop Supply Chain, *European Journal of Operational Research*, 2009, Vol. 198, No. 3, pp. 810-821.

[25] Pineyro, P. and Viera, O., The Economic Lot-Sizing Problem with Remanufacturing and One-Way Substitution, *International Journal of Production Economics*, 2010, Vol. 124, No. 2, pp. 482-488.

[26] Sifaleras, A., Konstantaras, I. and Mladenovic, N., Variable Neighborhood Search for the Economic lot Sizing Problem with Product Returns and Recovery, *International Journal of Production Economics*, 2015, Vol. 160, pp. 133-143.

[27] Son, D.-H., An, S.-B., Kim, H.-J., and Jang, J.-M., Push and Pull Disassembly Quantity Models in a Reverse Supply Chain : The Case of an Automobile Disassem-

bly System in Korea, *International Journal of Logistics Research and Applications*, 2021, pp. 1-26.

[28] Taleb, K.N. and Gupta, S.M., Disassembly of Multiple Product Structures, *Computers & Industrial Engineering*, 1997, Vol. 32, pp. 949-961.

[29] Taleb, K.N., Gupta, S.M., and Brennan, L., Disassembly of Complex Product Structures with Parts and Materials Commonality, *Production Planning and Control*, 1997, Vol. 8, pp. 255-269.

[30] Teunter, R.H., Bayindir, Z.P., and van den Heuvel, W., Dynamic Lot Sizing with Product Returns and Remanu-facturing, *International Journal of Production Research*, 2006, Vol. 44, No. 20, pp. 4377-4400.

[31] Yang, J., Golany, B., and Yu, G., A Concave-Cost Pro-duction Planning Problem with Remanufacturing Options, *Naval Research Logistics*, 2005, Vol. 52, No. 5, pp. 443-458.

**ORCID**

Jin-Myeong Jang  | http://orcid.org/0000-0003-3166-1488
Hwa-Joong Kim  | http://orcid.org/0000-0003-4976-4813
Dong-Hoon Son  | http://orcid.org/0000-0003-1313-3315
Dong-Ho Lee  | http://orcid.org/0000-0002-7679-5167