Genetic algorithm and Monte Carlo simulation for a stochastic capacitated disassembly lot-sizing problem under random lead times

Ilhem Slama, Oussama Ben-Ammar, Alexandre Dolgui, Faouzi Masmoudi

Version of Record: https://www.sciencedirect.com/science/article/pii/S0360835221003727
Manuscript_f5e7c2d83079eb30917e4a054a5a1480

Genetic algorithm and Monte Carlo simulation for a stochastic capacitated disassembly lot-sizing problem under random lead times

Ilhem Slama[a,*], Oussama Ben-Ammar[b], Alexandre Dolgui[a], Faouzi Masmoudi[c]

[a] *IMT Atlantique, LS2N-CNRS, La Chantrerie, 4 rue Alfred Kastler - B.P. 20722, 44307 Nantes, France*
{ilhem.slama;alexandre.dolgui}@imt-atlantique.fr
[b] *Mines Saint-Étienne, Univ. Clermont Auvergne, UMR-CNRS 6158 LIMOS, CMP Department of Manufacturing Sciences and Logistics, 880 route de Mimet, F-13541 Gardanne, France*
oussama.ben-ammar@emse.fr
[c] *Engineering School of Sfax, Laboratory of Mechanic, Modeling and Production (LA2MP), University of Sfax, Tunisia*
masmoudi.fawzi@gmail.com

[*] Corresponding author
Email address: ilhemslm.fsegs@gmail.com (Ilhem Slama)

Genetic algorithm and Monte Carlo simulation for a stochastic capacitated disassembly lot-sizing problem under random lead times

Abstract

The purpose of this research is to propose several optimization methods for the stochastic multi-period disassembly lot-sizing problem. The case of one type of end-of-life product and a two-level disassembly system is studied. The disassembly lead times are discrete random variables with a known and bounded probability distribution. The objective is to optimize the expected value of the total cost, which is the sum of setup cost, overload cost, inventory holding cost and backlogging cost. Three approaches were developed to solve the studied problem: (i) a two-stage mixed-integer linear programming model based on all possible scenarios for small instances, (ii) a sample average approximation approach based on Monte Carlo simulation for medium-scale instances and, (iii) an optimization approach based on the Monte Carlo simulation and a genetic algorithm for large-scale instances. Experimental results show the effectiveness of the proposed models which can be used to support decision-making on replenishment and disassembly plans.

Keywords: Capacitated disassembly lot-sizing, Stochastic lead times, Monte Carlo Simulation, Sample average approximation, Genetic algorithm.

1. Introduction

A great deal of attention is now being paid to dealing with end-of-life (EoL) products for economic and environmental reasons such as incineration sites and shortages of landfill sites, legislative pressure, economic considerations etc. In particular, much legislation imposes an obligation on the manufacturing industry to collect and recover EoL products in more environmentally conscious ways (Kim et al., 2018b).

The disassembly operation is one of the important activities in recycling raw materials and separating reusable components. It can be defined as the process of separating EoL products into multiple components by sorting and inspection operations. Various disassembly optimization problems have been intensively studied in the literature such as disassembly line balancing problems (e.g., Liu et al., 2020; Xu et al., 2020; Ren et al., 2020; Jeunet et al., 2019; Li et al., 2019; Tseng et al., 2019, 2018; Li et al., 2018; Zhang & Zhang, 2010; Agrawal & Tiwari, 2008; Efendigil et al., 2008; Tiwari, 2005; Dini et al., 2001; Güngör & Gupta, 2001; Gungor & Gupta, 1998), the disassembly sequencing problems (e.g., Liu et al., 2020; Xu et al., 2020; Ren et al., 2020; Li et al., 2019; Jeunet et al., 2019; Tseng et al., 2019, 2018; Li et al., 2018; Han et al., 2013; Zhang & Zhang, 2010; Efendigil et al., 2008; Rai et al., 2002; Dini et al., 2001; Gungor & Gupta, 1998), the disassembly lot-sizing problem (e.g., Pour-Massahian-Tafti et al., 2020a,b; Slama et al., 2020; Habibi et al.,

2019; Tian & Zhang, 2019; Piewthongngam et al., 2019; Ji et al., 2016; Prakash et al., 2012; Kim et al., 2006a, 2003) and other interesting issues.

In this paper the demand for components must be satisfied from the disassembly of EoL products. The objective is to optimize the disassembly plan under capacity constraint and uncertainty of disassembly lead times (DLT). According to Ilgin & Gupta (2010), the studied problem enters the classification of the Disassembly Lot Sizing Problem (DLSP) that optimizes disassembly processes in order to satisfy demand of components over a given planning horizon.

Recently, a great deal of attention has been paid to studying disassembly lot-sizing problems. Many researchers have developed stochastic and deterministic approaches to optimize the disassembly plan for complex EoL products. Nevertheless, the majority of studied problems consider a deterministic environment and neglect uncertainty. In practice, managing disassembly operations is difficult due to technical problems (absenteeism, limited disassembly capacity, quality problem, etc.). When an EoL product is disassembled, it is fundamental to ensure that the right model is applied. Owing to the complexity of the manual disassembly operation, the workforce is often the major factor generating uncertainty regarding disassembly duration. Consequently, this uncertainty of disassembly lead times (DLT) could disturb the whole disassembly line and create additional costs related to inventory levels, tardiness penalties, lost sales, overload capacity, etc.

The aim of this paper is to optimize the basic two-level disassembly lot-sizing model under stochastic DLT. The demand and available capacity are fixed and known in advance. The disassembly lead times are discrete random variables that follow a known and bounded probability distribution. Here, we assume that the disassembly plan is frozen over a given planning horizon of several periods. In this case, even if the lead times are revealed as time goes by, the two-stage stochastic model remains valid. This is relevant in the capacity planning decision, as the schedule of the employees' weekly work schedules cannot be modified at short notice.

In this paper, we consider a predictive strategy and develop a two-stage model based on scenario approach. In our two-stage decision process, the disassembly lead times for the entire time horizon are revealed once the first stage decisions are made. These decisions correspond to the disassembly, ordering setup and disassembly overtime decisions for each time period. The second stage involves inventory and backlog decisions. The questions then become: At which point in time should we start the disassembly? and what quantity must be disassembled?

We summarize below the contributions and innovations of this study:

1. An original stochastic Capacitated Disassembly Lot Sizing (CDLS) model is formulated as a two-Stage Mixed Integer Linear Program (2S-MILP).
2. To alleviate the scalability issues, a sample average approximation approach (SAA) based on Monte Carlo simulation is proposed.
3. A sensitivity study is introduced to analyze the effect of disassembly lead time variability on the robustness of the proposed SAA algorithm.

4. To solve large scale problems, a model coupling MC simulation and a genetic algorithm (MC-GA) is proposed.
5. Useful managerial implications for industry practitioners are provided.

The rest of this paper is organized as follows. The next section outlines a brief overview of the stochastic DLSP literature. Section 3 gives the problem statement and model formulation. Section 4 presents the approach based on sample average approximation algorithm. In section 5, we introduce the model coupling MC simulation and a genetic algorithm. Section 6 reports numerical results. The paper ends with conclusions and avenues for future studies.

2. Literature review

In the literature, the majority of work focused on DLSP in a deterministic environment. Concerning this field, the existing literature can be split into two categories: (i) uncapacitated problems (see for example Pour-Massahian-Tafti et al., 2020a,b; Godichaud & Amodeo, 2019; Piewthongngam et al., 2019; Kim et al., 2018a; Godichaud & Amodeo, 2018; Kang et al., 2016; Gupta & Lambert, 2016; Prakash et al., 2012; Kim et al., 2009; Langella, 2007; Kim et al., 2006b; Lee et al., 2004; Lee & Xirouchakis, 2004; Kim et al., 2003; Neuendorf et al., 2001; Taleb et al., 1997a,b) and (ii) capacitated ones (see for example Slama et al., 2020; Ji et al., 2016; Ullerich & Buscher, 2013; Kim et al., 2006a, 2005; Lee et al., 2002). Table 1 gives a recapitulation of previous works which we have classified as deterministic *vs* stochastic approaches. For more detailed literature reviews, readers can refer to Slama et al. (2019) and Kim et al. (2007).

Table 1: Summary of relevant literature.

(a) Deterministic approaches.

Authors	Criteria	Resolution	**BOM**		Cap	PC
			T	**M**		
Gupta & Taleb (1994)	$Min(NP)$	R-MRP		√		
Taleb et al. (1997a)	$Min(NP)$	R-MRP		√		
Taleb et al. (1997b)	$Min(NP+D_c)$	heuristics		√		
Neuendorf et al. (2001)	$Min(NP)$	Petri nets		√		
Lee et al. (2002)	$Min(P_c+H_c+D_c)$	LP		√	√	
Kim et al. (2003)	$Min(S_c+H_c+D_c)$	LP		√	√	
Lee et al. (2004)	$Min(S_c+P_c+H_c+D_c)$	LP		√		√
Lee & Xirouchakis (2004)	$Min(S_c+P_c+H_c+D_c)$	Heuristic		√		√
Kim et al. (2005)	$Min(NP)$	DP		√	√	
Kim et al. (2006b)	$Min(S_c+H_c+D_c)$	Heuristic		√	√	
Kim et al. (2006a)	$Min(S_c+H_c+D_c)$	LH		√	√	
Kim et al. (2008)	$Min(S_c+H_c)$	DP	√			
Gao & Chen (2008)	$Min(S_c+H_c+D_c)$	GA		√		
Barba-Gutiérrez et al. (2008)	$Min(S_c+H_c+D_c)$	LS-RMRP		√		
Kim et al. (2009)	$Min(S_c+H_c)$	Branch and bound		√	√	
Kim & Lee (2011)	$Min(S_c+H_c+D_c)$	Heuristic		√		
Prakash et al. (2012)	$Min(H_c+D_c+P_c+S_c)$	CBSA		√		√
Ullerich & Buscher (2013)	$Min(H_c+D_c+P_c)$	Heuristic	√			√
Ji et al. (2016)	$Min(S_c+H_c+D_c+Z_c+C_c)$	LH	√			√
Godichaud et al. (2015)	$Min(S_c+H_c+L_c+O_c)$	GA	√		√	
Tian & Zhang (2019)	$Min(S_c+P_c+H_c+W_c)$	PSO	√			
Kim et al. (2018a)	$Min(S_c+H_c+D_c)$	Heuristic		√		
Godichaud & Amodeo (2018)	$Min(D_c+H_c+W_c+L_c)$	EOQ model	√			
Piewthongngam et al. (2019)	$Min(S_c+H_c+D_c+Z_c)$	MILP		√		√
Godichaud & Amodeo (2019)	$Min(D_c+H_c+W_c+L_c)$	EOQ model		√		
Slama et al. (2020)	Max profit	MIP		√	√	√
Pour-Massahian-Tafti et al. (2020a)	Max profit	EOQ model	√			
Pour-Massahian-Tafti et al. (2020b)	$Min(S_c+H_c+D_c)$	Heuristics	√			√

(b) Stochastic approaches.

Authors	Objective	Resolution	**Uncertainty**		
			Yield	**Demand**	**DLT**
Inderfurth & Langella (2006)	$Min(NP)$	Heuristic	√		
Barba-Gutiérrez & Adenso-Díaz (2009)	$Min(NP)$	F-RMRP		√	
Kim & Xirouchakis (2010)	$Min(S_c+H_c+L_c)$	LH		√	
Fang et al. (2017)	$Min(H_c+D_c+S_c+P_c)$	LH		√	
Liu & Zhang (2018)	$Min(S_c+H_c+P_c)$	Outer-approximation	√	√	
Current paper	$Min(H_c+S_c+A_c+O_c)$	GA			√

"T: Two-level, M: Multi-level, Cap : Capacity, PC: Parts commonalities, NP: Number of disassembled products, R-MRP: Reverse-material requirements planning, D_c: Disassembly operation cost, P_c: Purchase cost of root, H_c: Holding cost, S_c: Setup cost, LP: Linear programming, DP: Dynamic programming, LS-RMRP: Lot-sizing-RMRP, CBSA: Constraint-based simulated annealing, Z_c: Purchase cost of parts, W_c: Waste cost, C_c: Start-up cost, L_c: Lost sales cost, O_c: Overload cost, A_c: Backlogging cost, LH: Lagrangian heuristics, EOQ: Economic order quantity, DP: Dynamic programming".

In the last two decades, there has been a growing interest in developing methods and approaches to cope with uncertainties in supply and production planning. In the field of reverse logistics, the existing literature is very limited and has focused mainly on the uncertainty of demand and/or disassembly yield.

For the uncertainty of demand, Barba-Gutiérrez & Adenso-Díaz (2009) used trapezoidal distributions to characterize the problem. For solving the DLSP, they suggested a fuzzy Reverse Materials Requirement Planning (RMRP) algorithm. The authors determine the quantity of EoL products to be disassembled at each period. Their fuzzy RMRP approach (F-RMPR) seems to perform better than the traditional RMRP algorithm with a reduced inventory level. However, the proposed approach only seeks to satisfy component demands. The authors neglected the optimization of costs related to the disassembly process whereas these are an important element for companies. One year later, Kim & Xirouchakis (2010) studied the CDLSP with a multi-period planning, two-level bill-of-material (BOM) and a multi-type EoL product. To solve this problem, they developed a Lagrangian relaxation heuristic to minimize the sum of expected total cost which is equal to the sum of inventory, setup and penalty costs for the unfulfilled item demands. The work of Fang et al. (2017) studied the multi-period and multi-product stochastic problem for remanufacturing systems. In order to solve the studied problem, the authors introduced a scenario-based approach, a multi-stage stochastic MILP and a heuristic based on Lagrangian relaxation. The impact of the uncertainty of the demands on the solution is analyzed by a sensitivity study of several scenarios.

The uncertainty of yield occurs when the difference between the quantities of items released and those obtained after disassembly is random. Inderfurth & Langella (2006) considered a multi-type product with parts commonality and solve the single period disassembly-to-order problem under random disassembly yield (number of units of components obtained from disassembling one unit of parent item). To solve the considered problem, they developed a heuristic to reduce the expected disposal, purchasing and disassembly operation costs. In Inderfurth et al. (2015), the authors developed a mathematical model to illustrate the effect of stochastic disassembly yield on stochastically proportional and binomial models. The results indicate that presuming binomial yield is preferable to assuming proportional yield.

Few studies have been published on the uncertainty of demand and yield. Liu & Zhang (2018) formulated the stochastic DLSP with these uncertainties as a mixed integer nonlinear programming and developed an outer approximation-based solution algorithm to solve it. In that paper, the authors simultaneously use the maximum and minimum values of both a uniform (for demand) and normal (for stochastic disassembly yield) distribution to solve the DLSP.

As explained previously, variability in disassembly times is often observed during disassembly processes. The related uncertainty is due to technical problems (absenteeism, limited disassembly capacity, etc.) and economic conditions (availability of EoL products and variability of costs, etc.). As far as we know, there is not much literature on disassembly planning under uncertainty of durations. Most studies have only focused on deterministic environments. The aim of our research is to investigate stochastic CDLSP, and it outlines a new approach to optimize order and disassembling policies.

3. Problem statement and model formulation

3.1. Problem statement

Let us take a close look at the disassembly planning of a two level disassembly system in which the EoL product is disassembled into several components. A graphical illustration of the two-level BOM, N components and single type of EoL product is given in Fig. 1. The customer's component requirements are known over the planning horizon and should be delivered on predefined delivery dates. The number in brackets represents the disassembly yield or the number of components obtained by the disassembly operation of one unit of the EoL product. At the beginning of each period, a disassembly order for EoL product is made. In each period, we assume a limited disassembly time capacity. If this capacity is not sufficient to meet the component requirements in a given period t, an additional capacity can be added with a unit penalty cost u_t.

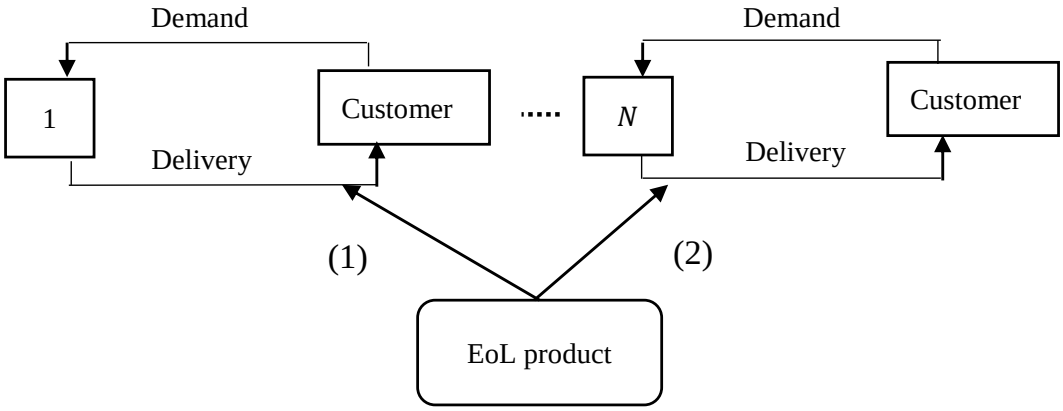


Figure 1: Two level disassembly system.

Disassembly systems are particularly vulnerable to EoL product disassembly lead times, since all the necessary components must be disassembled before the expected delivery dates. In fact, EoL product disassembly lead times, also named flow times, represent the time difference between releasing disassembly order and receiving the components from a disassembly order. To adhere more closely to industrial methods of planning, we assume a discrete temporal environment. The actual lead times are random discrete variables with known and finite possible values, and follow a known probability distribution. In fact, we assume that the workshop which disassembles the EoL products receives them from several disassembly lines. The workload of these workshops varies from period to period. In this case, each disassembly lead time depends on the quality of the EoL product and the workload of the refurbishment workshop at each period t. Thereafter, the disassembled items are available after a stochastic disassembly lead time (L_t).

Before modeling the considered problem, we detail the following assumptions frequently encountered for this type of problem (Kim et al., 2006b, 2009; Kim & Xirouchakis, 2010):

- The end-of-life products are available once they are requested;
- Dynamic demands for the components are known over the planning horizon;
- Demands are only satisfied by the disassembly system;
- All disassembled items are in good condition to satisfy the demand;

As EoL disassembly lead times are random variables over the planning horizon, we provide a scenario based stochastic optimization formulation. A scenario represents a possible realization of the disassembly lead time in each period. More precisely, the lead times L_t are random discrete variables varying between L^- and L^+. Thus, if any disassembly process starts in period t, all components are available at period $t+L_t$. Note that, if any disassembly order is released in period t, a setup cost s_t is occurred in that period. When a component i is received before its desired delivery date, an inventory holding cost h_i per period of time is incurred. Conversely, a backlogging cost b_i is assessed at the end of each period per unit backlogged. The objective is to optimize the disassembly quantities of the EoL product to minimise the sum of the expected holding and backlog costs of all disassembled components among all scenarios, as well as the setup and overload costs over a finite horizon.

3.2. Two-stage mixed integer linear programming (2S-MILP)

To solve the CDLSP with stochastic disassembly lead times, a stochastic 2S-MILP is proposed to minimize the expected total cost, denoted by $\mathbb{E}(TC)$, and to select the appropriate disassembly schedule plan i.e. the optimal quantity of EoL product to disassemble at each period over the planning horizon. The full list of notations used throughout this paper is given in Table 2.

Definition 3.1. *Let Ω_0 be the set of all possible scenarios which contains all combinations of the random disassembly lead times. Each scenario ω corresponds to the realisations of all lead times of periods $1,...,t$. Its probability value is $\mathbb{P}_\omega$. Knowing that all disassembly times are independent, so $\mathbb{P}_\omega = \prod_{t\in\mathcal{T}} \Pr(L_t = L_t^\omega)$ and $|\Omega_0| = (L^+ - L^- + 1)^{|\mathcal{T}|}$.*

Table 2: Notation.

Indexes	
t	index of periods, $\forall t = 1,...,T$
i	index of components, $\forall i = 1,...,N$
ω	index for scenarios, $\forall \omega = 1,...,\|\Omega_0\|$
Parameters	
$\mathcal{T}$	set of time periods of the planning horizon , $\|\mathcal{T}\| = T$
$\mathcal{N}$	set of components , $\|\mathcal{N}\| = N$
Ω_0	set of possible scenarios

R_i	disassembly yield of component i
$D_{i,t}$	external demand for component i in period t
I_{i0}	starting inventory of component i
L_t^ω	disassembly lead time in period t for scenario ω
h_i	unit time inventory holding cost of one unit of component i
s_t	setup cost in period t
b_i	unit time backlog cost of one unit of component i
G	disassembly operation time
C_t	available capacity in period t
u_t	cost of adding a unit of extra capacity in period t
K	Large number
Functions	
$\mathbb{E}(.)$	Expected value
$\mathbb{P}_\omega$	Probability value related to scenario ω, $\sum_{\omega\in\Omega}\mathbb{P}_\omega = 1$
First-stage decision variables	
Z_t	disassembly quantity ordered in period t
Y_t	binary indicator of disassembly in period t
O_t	disassembly overtime in period t
Second-stage decision variables	
H_{it}^ω	inventory level of component i at period t for scenario ω
B_{it}^ω	backlogging level of component i at period t for scenario ω
I_{it}^ω	inventory level at the end of period t. It is equal to $H_{it}^\omega - B_{it}^\omega$

The capacitated disassembly lot-sizing problem with stochastic disassembly lead times can be formulated using the following stochastic 2S-MILP model based on scenarios formulation.

$$\mathbb{E}(TC) = min \sum_{t\in\mathcal{T}}\Big(\sum_{\forall i\in\mathcal{N}}\sum_{\omega\in\Omega_0}\mathbb{P}_\omega\Big(h_i H_{i,t}^\omega + b_i B_{i,t}^\omega\Big) + s_t Y_t + u_t O_t\Big) \tag{1}$$

subject to constraints:

$$H_{i,t}^{\omega} - B_{i,t}^{\omega} = I_{i0} + \sum_{\tau \in \mathcal{T} \mid \tau + L_{\tau}^{\omega} \leq t} R_i Z_{\tau} - \sum_{\tau=1}^{t} D_{i,\tau} \qquad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \Omega_0 \tag{2}$$

$$Z_t \leq K.Y_t \qquad \forall t \in \mathcal{T} \tag{3}$$

$$G.Z_t \leq C_t + O_t \qquad \forall t \in \mathcal{T} \tag{4}$$

$$Z_t, O_t \geq 0 \qquad \forall t \in \mathcal{T} \tag{5}$$

$$Y_t \in \{0,1\} \qquad \forall t \in \mathcal{T} \tag{6}$$

$$H_{i,t}^{\omega}, B_{i,t}^{\omega} \geq 0 \qquad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \Omega_0 \tag{7}$$

The $\mathbb{E}(TC)$ can be calculated by considering all possible values of L_t^{ω} and the objective function, expressed in Constraint (1) minimizes, for the first stage of the problem, the sum of the setup and penalty of extra capacity costs and the expected inventory holding and backlog costs for the second stage. Constraints (2) define for the second stage problem, the inventory balance for each component i at the end of each period t of scenario ω. Constraints (3) guarantee that a setup cost is generated in a period t if any disassembly operation needs to be performed in that period. Constraints (4) represent the disassembling capacity constraint in each period t. Constraints (5-6) provide the conditions framing the decision variables.

Before moving on to the core topic of this research, we briefly examine the complexity of the studied problem. According to Florian et al. (1980) and Bitran & Yanasse (1982), the single item capacitated lot-sizing problem (SCLSP) is a special case of the CDLSP with a single disassembled item and L_t equals to 0 with probability 1. As the SCLSP is NP-hard, the CDLSP is NP-hard too.

4. Sample average approximation algorithm

Solving the 2S-MILP model with the set of all possible scenarios leads to the exact solution. However, the complexity of the problem may increase exponentially if a large set of Ω_0 is considered to represent the stochastic disassembly lead times and the proposed 2S-MILP becomes hard to solve. Therefore, in order to estimate the expected total cost (Kleywegt et al., 2002; Fishman, 1996), we propose the Sample Average Approximation (SAA) algorithm based on the Monte-Carlo (MC) sampling as proposed in Kim et al. (2018b).

The flowchart of the SAA solution algorithm developed in this study is presented in Fig. 2. In this figure, Ω_1, M and Ω_2 represent the current random sample size, the number of replications and the maximum sample size, respectively. The proposed algorithm includes three main steps at each replication: (1) generating Ω_1 random scenarios for disassembly lead times; (2) solving the SAA problem and (3) checking the stopping condition (i.e. the variance of the gap estimator (VGE) and the percentage optimality gap (POG) must be sufficiently small). It is important to note that the three main steps are repeatedly carried out after the current random sample size Ω_1 is increased by a specified quantity ρ until the number of the random scenarios reaches the maximum Ω_2 unless the stopping condition is satisfied.

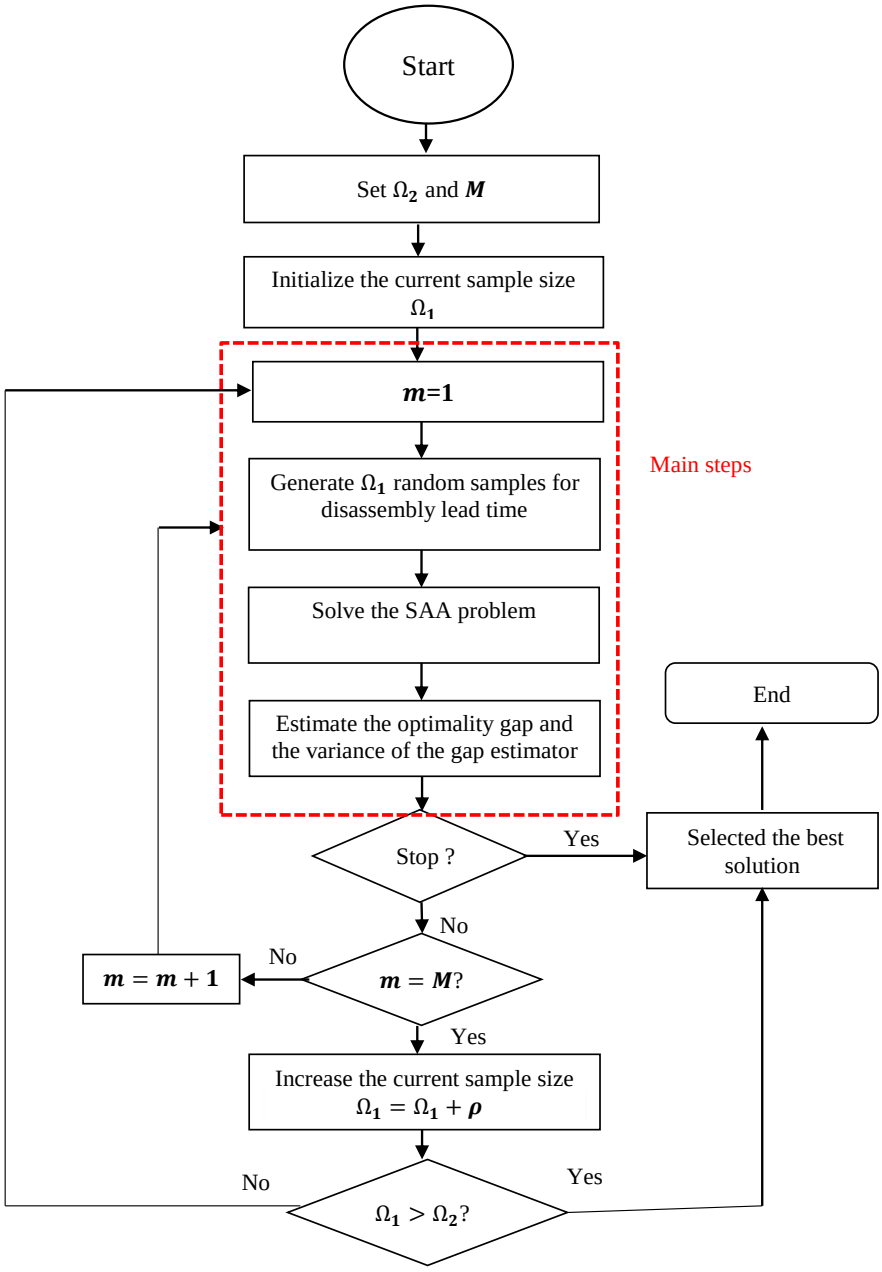


Figure 2: Sample average approximation-based algorithm (Kim et al., 2018b)

4.1. generating random samples

The average total cost, denoted by ATC, is obtained by sampling Ω_1 disassembly lead time scenarios. To build a scenario $\omega \in \Omega_1$, each disassembly lead time $L_t^\omega \in L_1^\omega, \ldots, L_{|\mathcal{T}|}^\omega$ is sampled randomly by following the probability distribution of L_t. Thus, the probability related to the scenario ω is $\mathbb{P}_\omega$.

Remark 4.1. *According to the law of large numbers, a large number of samples is needed to obtain a precise estimation of the exact total cost (Lamiri et al., 2008).*

4.2. Solving the sample average approximation problem

Corollary 4.2. *For a given set of random samples L_t^{ω} for $\omega = 1,..,|\Omega_1|$, the SAA problem can be formulated as follow:*

$$ATC = \ min \sum_{\forall t \in \mathcal{T}} \Big(\sum_{\forall i \in \mathcal{N}} \sum_{\omega \in \Omega_1} \frac{1}{|\Omega_1|} \Big(h_i H_{i,t}^{\omega} + b_i B_{i,t}^{\omega} \Big) + s_t Y_t + u_t O_t \Big) \tag{8}$$

subject to (3)-(6) and:

$$H_{i,t}^{\omega} - B_{i,t}^{\omega} = I_{i0} + \sum_{\tau \in \mathcal{T} \ | \tau + L_{\tau}^{\omega} \leq t} R_i . Z_{\tau} - \sum_{\tau=1}^{t} D_{i,\tau} \qquad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \Omega_1 \tag{9}$$

$$H_{i,t}^{\omega}, B_{i,t}^{\omega} \geq 0 \qquad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \Omega_1 \tag{10}$$

Expression (8) is an estimator of the expected total cost given in Expression (1). Constraints (9)-(10) provide the inventory level for each component i at the end of each period t for each scenario $\omega \in \Omega_1$.

4.3. Checking the stopping condition

The third main step of the proposed SAA is to check the stopping condition. The proposed SAA algorithm is stopped if the POG and the VGO are sufficiently small. These two parameters are estimated according to the method proposed by Kleywegt et al. (2002) and Kim et al. (2018b).

4.3.1. Estimation of the optimality gap

The estimated optimality gap is the difference between the lower and the upper bounds. The lower bound is calculated as the cumulative ATC from the first to the m^{th} replication as presented in Equation (11):

$$\overline{ATC}_{\Omega_1} = \frac{1}{m} \sum_{h=1}^{m} ATC_{\Omega_1}^{h} \tag{11}$$

where $ATC_{\Omega_1}^{h}$ represents the optimal ATC obtained at the h^{th} replication of the current sample size Ω_1 by solving the SAA problem. According to Mak et al. (1999); Norkin et al. (1998), $\overline{ATC}_{\Omega_1}$ gives a valid statistical estimation of the lower bound because $\mathbb{E}(\overline{ATC}_{\Omega_1}) < \mathbb{E}(TC)$.

According to Kim et al. (2018b), any ATC obtained under Ω_2 sample size is always greater than or equal to the $\mathbb{E}(TC)$. This is explained by the fact that ATC is an unbiased estimator of the objective function value. Recall that Ω_2 is the maximum sample size that must be much greater than Ω_1. Thus, an upper bound ATC_{Ω_2} for a random sample of size Ω_2 can be estimated as presented in Equation (12):

$$ATC_{\Omega_2} = min \sum_{\forall t \in \mathcal{T}} \Big(\sum_{\forall i \in \mathcal{N}} \sum_{\omega \in \Omega_2} \frac{1}{|\Omega_2|} \Big(h_i H_{i,t}^{\omega} + b_i B_{i,t}^{\omega} \Big) + s_t Y_t + u_t O_t \Big) \tag{12}$$

where $\{L_t^1, L_t^2, ..., L_t^{\Omega_2}\}$ is the set of independent random scenarios of the disassembly lead time.

The percentage optimality gap (POG) can be calculated as follows:

$$POG = \left(\frac{ATC_{\Omega_2}^{*} - \overline{ATC}_{\Omega_1}}{ATC_{\Omega_1}} \right) \times 100$$

where $ATC_{\Omega_2}^{*}$ is the minimum ATC obtained over all the replications under Ω_2 random samples.

4.3.2. Estimation of the variance gap estimator

The variance gap estimator depends on the variances of $\overline{ATC}_{\Omega_1}$ and $ATC^*_{\Omega_2}$. It can be formulated as follows:

$$\sigma^2_{ATC^*_{\Omega_2}-\overline{ATC}_{\Omega_1}} = \sigma^2_{ATC^*_{\Omega_2}} + \sigma^2_{\overline{ATC}_{\Omega_1}}$$

where each of the two terms can be estimated as follow:

$$\sigma^2_{ATC^*_{\Omega_2}} = \frac{1}{\Omega_2(\Omega_2-1)} \sum_{\omega=1}^{\Omega_2} \left(ATC_{\Omega_2} - ATC^*_{\Omega_2} \right)^2$$

and,

$$\sigma^2_{\overline{ATC}_{\Omega_1}} = \frac{1}{m(m-1)} \sum_{h=1}^{m} \left(ATC^h_{\Omega_1} - \overline{ATC}_{\Omega_1} \right)^2$$

Although the SAA approach based on the MC simulation technique is interesting for estimating the expected total cost, it can suffer from the impossibility of obtaining optimal solutions as the complexity of the problem increases. An approximate solution method is needed to solve large problems. To study disassembly systems with more components and periods, introducing another resolution approach seems to be necessary.

5. Genetic algorithm and Monte-Carlo simulation

Various meta-heuristics are proposed in the literature to solve optimization problems. The genetic algorithm (GA) has a proven track record for the two-level assembly systems under random lead times (e.g., Ben-Ammar et al., 2019, 2018; Guiras et al., 2019; Sakiani et al., 2012; Fallah-Jamshidi et al., 2011; Hnaien et al., 2010, 2009). For this reason, we decided to use the GA to optimize the two-level disassembly systems studied in this research. In fact, the representation of the solutions and the reproduction operators (crossover and mutation) can easily be set up for the considered problem. Moreover, insofar as there are no particular constraints, meta-heuristics based on a local search, such as taboo search or simulated annealing, would require the exploration of a large number of neighbors and thus a significant computation time.

A genetic algorithm coupled with the Monte Carlo sampling (MC-GA) approach is suggested to calculate the quantity of EoL product to be disassembled in each period while minimizing the average total cost given in Equation (8). This approach is modeled and inspired by the natural evolutionary process (Goren et al., 2010) which is based on a randomly generated population of individuals. Given the important literature existing on the topic, for the objective of the present paper, it suffices to recall few basic notions regarding the GA technique. Fig. 3 shows a flowchart of the proposed MC-GA. The different steps will be presented, in detail, in the following sub-sections.

5.1. Generation of the initial population of size n_{pop}

In the implementation of an MC-GA, the first step is to create the initial population of individuals. Each individual is coded by a chromosome. The variables in the SAA model are $H^{\omega}_{i,t}$, $B^{\omega}_{i,t}$, O_t among $Z_t > 0$ if Y_t=1 and Z_t=0 otherwise, $\forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \Omega_1$. In this study, we only encode Z_t the disassembly quantity

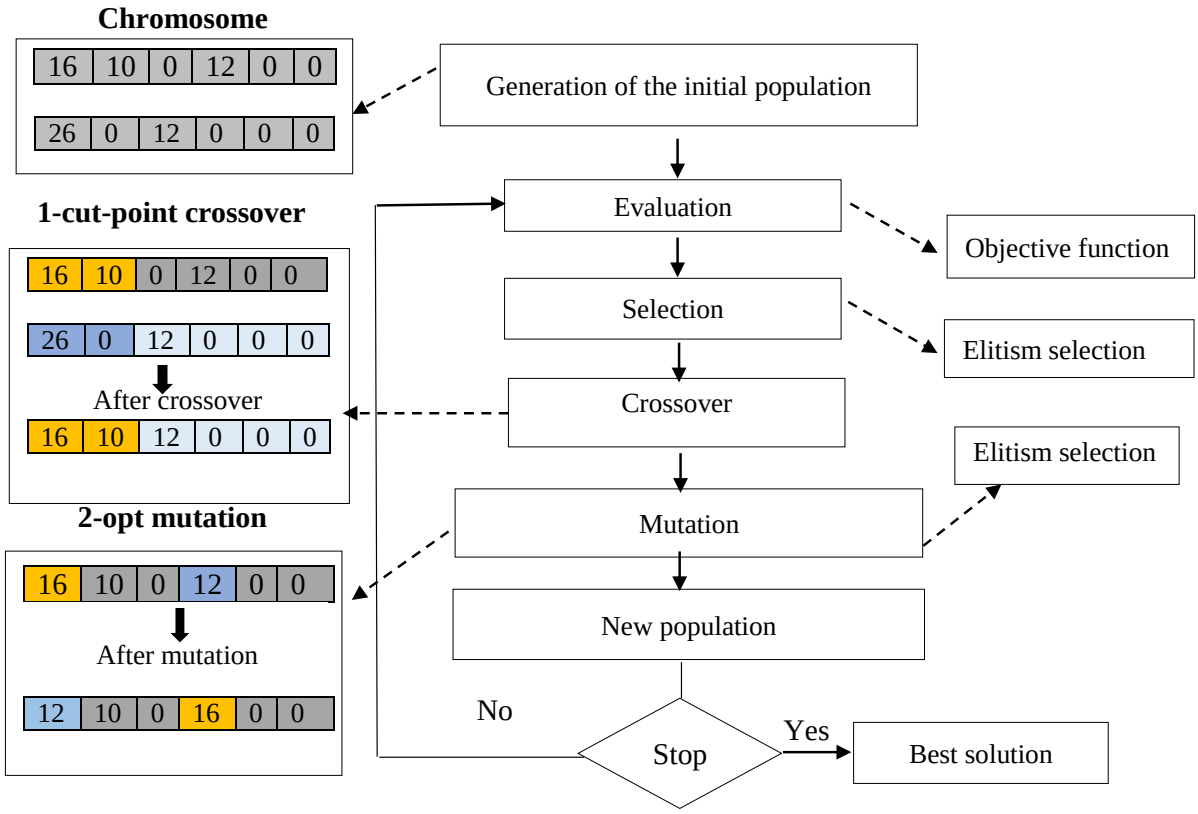


Figure 3: Flowchart of the proposed MC-GA.

of EoL product ordered in each period t. The other variables depend on Z_t and so they are not taken into account in the encoding step. Therefore, each gene in a chromosome represents a quantity of EoL product to be disassembled in each period t. The length of the chromosome is equal to the number of periods T. A Monte Carlo code must be run for each individual of the population throughout all the generations. Thus, each individual is built in many steps (see the procedure of generation of each individual of the population in Fig. 4).

Step 1: The first step is to generate the disassembly periods randomly ($Z_t > 0$ and $Y_t = 1$). Let $j < t$ to be the period such as $Z_j > 0$ and $Z_k = 0$, for $k = j+1,..,t$. This implies that the amount of EoL product is the quantity needed to satisfy the demands of all components from the period j to t, such as $j \leq k \leq t$. We can see that the problem can be decomposed into $t-$period sub-problems (one of the periods j to t , and the other periods $t+1$ to T). Let l_t be the last setup period in t-period sub-problem (i.e. t is the last setup period in t-period).

Step 2: The second step allows the calculation for each $t-$period, the disassembly quantity Z_j in period j. The exact value of Z_j is determined in order to satisfy the maximum demand among all the components

of the period j to t and all scenarios. We can determine Z_j using Equations (13) and take this lot-size plan as a final one. The main method of generating the GA is that the initial population is randomly generated. This method does not necessarily provide solutions that respect the disassembly capacity in each time period. Therefore, each overloading will be penalized.

$$Z_j = \max_{[\forall i, \forall \omega]} \left\lceil \frac{\sum_{i=1}^{N} \sum_{k=j}^{t} D_{i,k} - H^{\omega}_{i,j-1} + B^{\omega}_{i,j-1}}{R_i} \right\rceil \text{ and } \; Z_k = 0 \; \forall k = j+1, .., t \tag{13}$$

where $\lceil x \rceil$ is the smallest integer greater than or equal to x.

The relation between the ordered and received quantities of disassembled EoL product is defined by Equations (14). Let $X^{\omega}_{t'}$ be the quantity of EoL products received at period t' for the scenario ω. Therefore, depending on the disassembly lead time, the scenario is that an order placed at period j will be received at $t^{'} = j + L^{\omega}_{j}$. We underline that several orders made at different dates can be received at the same time. That explains why the received quantity is the sum of the orders having the same delivery date:

$$X^{\omega}_{t'} = \sum_{j \leq t'} Z_t, \qquad \text{if } j + L^{\omega}_{j} = t' \qquad \forall j, t' \in \mathcal{T}, \forall \omega \in \Omega_1 \tag{14}$$

Step 3: For each $t-$period, we can calculate the inventory and backlogging levels for each component i and scenario ω at the end of period k $(j \leq k \leq t)$ by using Equations (15) and (16), respectively:

$$H^{\omega}_{i,k} = (H^{\omega}_{i,k-1} + R_i X^{\omega}_{t'} - D_{i,k} + B^{\omega}_{i,k-1})^{+} \qquad \forall k = j, ..., t \in \mathcal{T}, \forall \omega \in \Omega_1, \forall i \in \mathcal{N} \tag{15}$$

$$B^{\omega}_{i,k} = (H^{\omega}_{i,k-1} + R_i X^{\omega}_{t'} - D_{i,k} + B^{\omega}_{i,k-1})^{-} \qquad \forall k = j, ..., t \in \mathcal{T}, \forall \omega \in \Omega_1, \forall i \in \mathcal{N} \tag{16}$$

where $(X)^{+} = \max\{0, X\}$ and $(X)^{-} = \max\{0, -X\}$.

Step 4: After obtaining all the quantities of the EoL product to be ordered in each period t, we determine the additional capacity O_t using Equation (17) and then penalize its violation.

$$O_t = \Big(G \times Z_t - C_t \Big)^{+} \qquad \forall t \in \mathcal{T} \tag{17}$$

5.2. Evaluation

We use Equation (8) to evaluate the fitness value of each individual.

5.3. Selection

There are many selection methods in the literature such as rank selection, Elitist selection, Boltzmann selection, roulette selection, etc.(Belkhamsa et al., 2018). Based on the significant amount of literature related to replenishment planning under uncertainty of lead times where the Elitist selection is widely used (e.g., Choi et al., 2009; Hnaien et al., 2009; Che & Chiang, 2010), we decide to use this type of selection in this study. Thus, an individual is selected according to his performance. In our case, there are two selection phases:

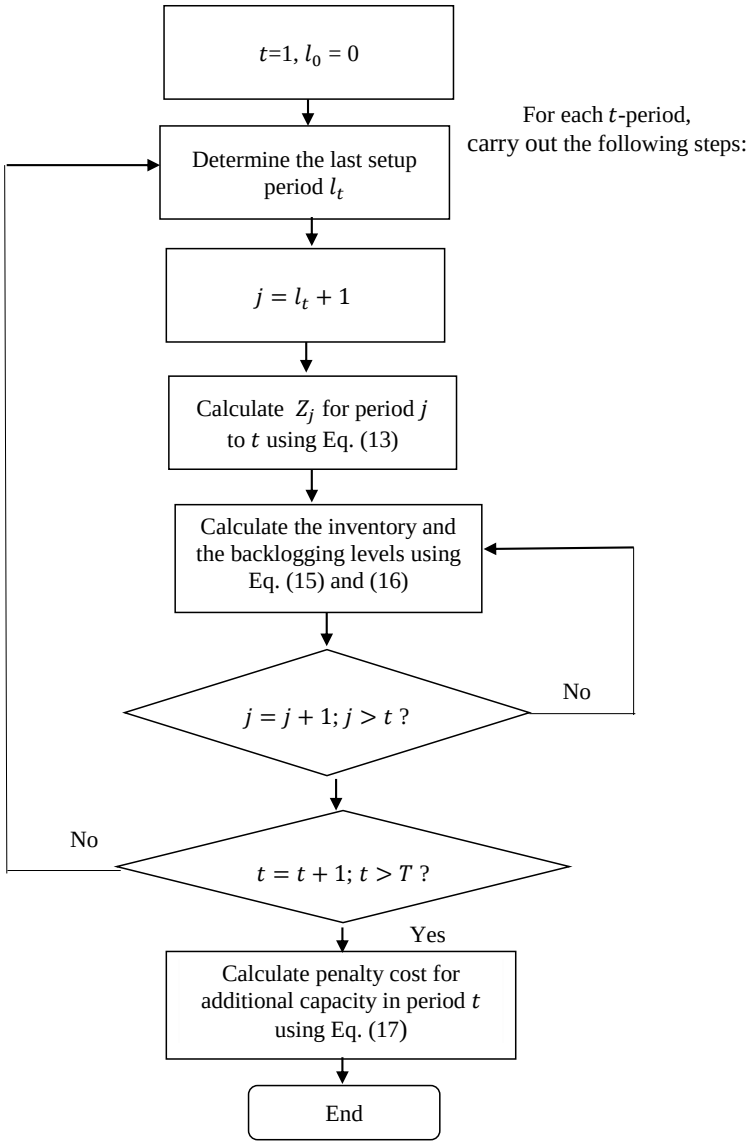


Figure 4: Procedure of generation of each individual.

- Reproductive selection: The $n_{pop}/2$ performed individuals in the population are chosen to undergo breeding operations (crossover and mutation). The selection is $n_{pop}/2$ Elitist.
- Replacement selection is the selection of each new population of at each generation. This new population is composed of the best individuals from the original generation and those obtained from mutation and crossover operations.

5.4. Crossover

The crossover operator permits us to produce a new individual called "offspring" from selected parents to obtain a "better" solution (Belkhamsa et al., 2018) (in our work, best quantity of the EoL product must be ordered in each period t). A standard 1-cut-point is considered. In this case, the first segment of the first

chromosome is followed by the second segment of the second chromosome (see 1-cut-point crossover in Fig. 3). The crossover operator is performed according to a probability P_c. The cut-point is randomly generated.

5.5. Mutation

The mutation operator permits the diversification to be kept and to escape the local optima. It is a matter of randomly modifying some individuals in our population by modifying one gene by another (see 2-opt mutation in Fig. 3). The mutation operator is performed according to a probability P_m.

5.6. New population

Once we have created new individuals by selection, crossover and mutation, we must select those who will constitute the new population.

5.7. The stopping criteria

The GA process is repeated until the stopping criteria is met. The calculation time is an indispensable factor in the decision making process. For this reason, as in Liu & Zhang (2018), we limit the computation time to 600 seconds for the execution of the proposed approach.

In the next section, the proposed approaches are evaluated and several tests are performed to examine the robustness of the solution approaches. We also examine the usefulness of the SAA and the MC-GA approaches.

6. Computational experiments and results

In order to show the efficiency of the different solution methods, several numerical experiments were performed. Three solution approaches are used to solve the studied problem. The first approach based on the 2S-MILP model, gives exact solutions using all possible combinations of the random disassembly lead times. The second approach based on the SAA algorithm, gives the average total cost based on random sampling of the random parameter. The third approach is based on MC-GA. All tests are performed on a PC with processor Intel (R) Core ™ i7-5500 CPU @ 2.4 GHz and 8 Go RAM under Windows 10 Professional. The two first approaches were solved by IBM CPLEX 12.6 within the fixed 3600 seconds time frame.

6.1. Numerical example

In this section we introduce a small numerical example. We considered a finite planning horizon with 7 periods and a two-level disassembly system with 3 components (1, 2 and 3) as shown in Fig. 5. The disassembly operation extracts one component of type 1, two components of type 2 and one component of type 3.

The demands for components over the planning horizon $D_{i,t}$ are listed in Table 4. The unit time inventory holding and backlogging costs are h_i=3 and b_i=100, respectively $\forall i \in \mathcal{N}$. The setup cost s_t is equal to 20,

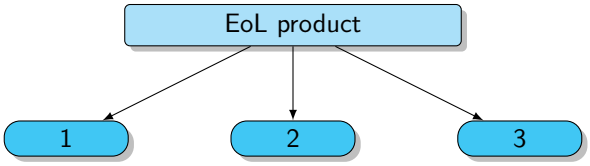


Figure 5: Two-level disassembly system

$\forall t \in \mathcal{T}$. Each disassembly lead time varies between 1 and 3 periods and follows the probability distribution given in Table 3. The disassembly operation time G is equal to 5. The available capacity C_t and the cost of adding a unit of extra capacity u_t are 80 and 10 $\forall t \in \mathcal{T}$, respectively. In this example, the number of all possible scenarios $|\Omega_0|$ is equal to 3^7 $((L^+ - L^- + 1)^{|\mathcal{T}|})$. In Table 4, we detail the optimal solution provided by the 2S-MILP model. More precisely, this table gives optimal values of variables in the two decision stages. In the first stage decision, we obtain the optimal quantity Z_t to be disassembled as well as the disassembly overtime O_t over the planning horizon. In the second stage decisions, the expected values of the inventory and the stockout of component i at each period t are calculated. The received quantities of component i at period t denoted by $\mathbb{E}(A_{i,t})$ are also presented in the same table. The optimal $\mathbb{E}(TC)$ takes the value 4752.43 and is obtained in 65.12 seconds.

Table 3: Disassembly lead time probability distribution

l	1	2	3
$Pr(L_t = l)$	0.245	0.49	0.265

In order to test the convergence provided by the MC optimization approach, the numerical example was tested for different values of $|\Omega_1|$: $[1, 30, 40, 50, 60, 70, 80, 90, 100, 200, \ldots, 1000]$. Here we only applied the first two steps of the SAA algorithm, i.e. without checking the stopping condition or evaluating the quality of the solution. The number of replications was set to 10, i.e. M = 10 under a given sample size Ω_1. Fig. 6 illustrates the convergence of the ATC (see Expression (8)) towards the exact value introduced in Expression (1) as well as the CPU needed to obtain this cost according to the different number of scenarios. On the one hand, this figure shows that the average total cost converges to the exact cost as the number of samples goes to infinity. In other words, a large number of samples is necessary to find a good approximation of the exact solution. On the other hand, for $\Omega_1 \geq 800$, the MC sampling provides a good approximation of optimal solution. Regarding the CPU, the same figure shows that the MC simulation-based approach can solve the problem in a reasonable time. Then, the application of the small test proved the effectiveness of the MC optimization approach that found a good compromise between the CPU time and the quality of the solution.

Fig. 6 describes how the optimal value ATC converges to $\mathbb{E}(TC)$ as the sample size Ω_1 increases. In the choice of sample size Ω_1, the trade-off between the quality of an optimal solution of the SAA problem, and the bounds on the percentage optimality and the variance of gap estimator should be taken into account.

Table 4: Solution of the numerical example using the 2S-MILP model

Period t	1	2	3	4	5	6	7
Z_t	30	50	16	4			
O_t	70	170					
$D_{1,t}$				10	70	20	
$\mathbb{E}(H_{1,t})$		7.35	34.29	60.66	12.73		
$\mathbb{E}(B_{1,t})$						1.05	
$\mathbb{E}(A_{1,t})$		7.35	26.95	36.37	22.07	6.20	1.06
$D_{2,t}$				60	50		
$\mathbb{E}(H_{2,t})$		14.70	68.59	81.33	75.47	87.87	90.00
$\mathbb{E}(B_{2,t})$							
$\mathbb{E}(A_{2,t})$		14.70	53.90	72.70	44.18	12.40	2.12
$D_{3,t}$			10	20	50	20	
$\mathbb{E}(H_{3,t})$		7.35	26.30	40.66	12.73		
$\mathbb{E}(B_{3,t})$		2.00				1.05	
$\mathbb{E}(A_{3,t})$		7.35	26.95	36.37	22.11	6.16	1.06

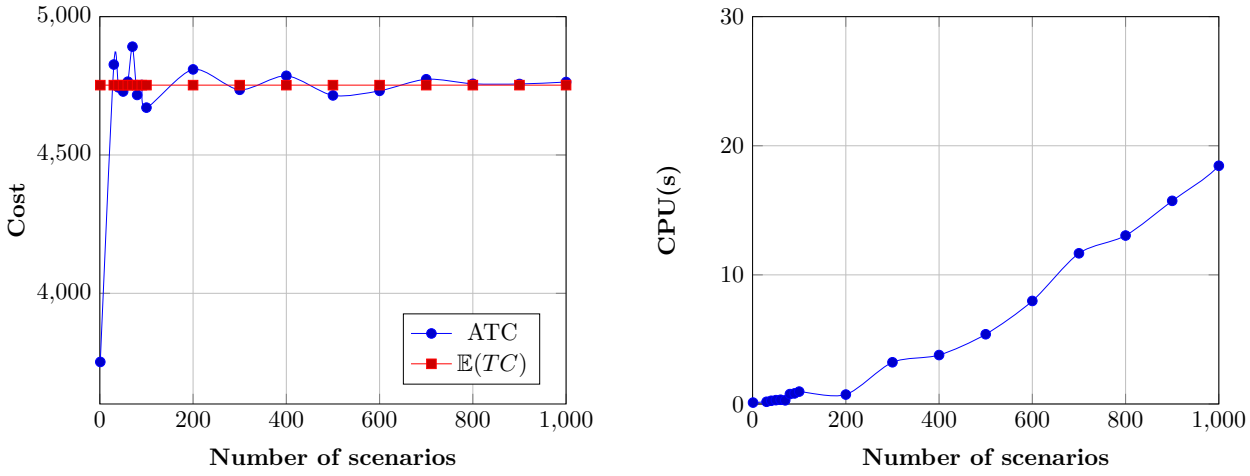


Figure 6: *ATC* convergence and CPU times.

In order to determine the random sample size that can be reached until a stopping condition is satisfied, a second test is performed and consists of applying the proposed SAA algorithm for different values of Ω_1: $[1, 30, 40, 50, 60, 70, 80, 90, 100, 200, \ldots, 1000]$. Recall that, if the percentage optimality and the variance of gap estimator are too large, the initial sample sizes Ω_1 must be increased until the sample size reaches the maximum Ω_2 unless the stopping condition is satisfied.

In our tests, the initial sample size is increased by 500 until the stopping condition is satisfied, i.e. the

percentage optimality gap and the variance of gap estimator for each replication are less than 5% and 10%, respectively. The maximum sample size $|\Omega_2|$ was set to 5,000. Fig. 7 shows the increased sample size as a function of the initial sample size. It can be seen from the figure that the increased sample size decreases as the initial sample size increases. In fact, the algorithm gave a robust solution with percentage optimality gap and variance of gap estimator less than 5% and 10%, respectively under the sample size of 1,000 without increasing the sample size.

The result of the SAA-based algorithm when the sample size Ω_1 was set to 1,000 is summarised in Table 5. It shows that the algorithm is stopped with the percentage optimality gap 4.02% and the variance of gap estimator 9.89% at the 5th replication. The resulting best quantities are $Z_1 = 30$, $Z_2 = 50$, $Z_3 = 20$, $Z_4 = 0$, $Z_5 = 0$, $Z_6 = 0$, $Z_7 = 0$. The related optimal ATC is equal to 4746 and obtained in only 19.92 seconds.

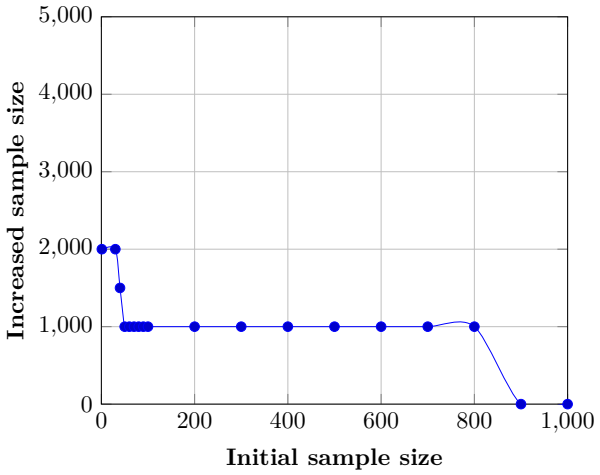


Figure 7: The increased random sample size as function of the initial random sample size.

6.2. Sensitivity analysis

In order to analyze the effect of disassembly lead times variability on the stability and robustness of the optimal solution found by the proposed SAA-based algorithm, the effect of variance (VAR) is treated. To do so, we generate 100 replications of the SAA problem under 1,000 random samples and we consider the same data instance presented in the previous sub-section. The probability distribution presented in Table 3 and denoted by VAR(a), is the reference case. Here, we vary the variance of the disassembly lead times between -75% and $+75\%$ as detailed in Table 6.

Fig. 8 illustrates the effect of the variation of the variance of disassembly lead times on the ATC. We firstly notice that, for a relatively low variation of the variance of disassembly lead times (VAR = -25% and $+25\%$), the variation of ATC increases slightly and remains at around $\pm 1\%$. For the large variationx ($\pm 75\%$), the variation of the ATC has also less significant impact on the average total cost and not exceed $\pm 3\%$. This proves that our approach remains robust even if the variance of lead times reaches $\pm$ 75%.

Table 5: SAA algorithm solution.

Replication (m)	Sample size ($\Omega_1 = 1,000$)	
	Percentage optimality gap	Variance of gap estimator (%)
1	4.12	-
2	6.23	19.24
3	4.54	11.75
4	5.77	09.15
5	4.02	09.89
6	-	-
7	-	-
8	-	-
9	-	-
10	-	-

Table 6: The changed distributions corresponding to different levels of VAR(a).

VAR	l	1	2	3
-75%	$\mathbb{P}(L_t = l)$	0.01	0.96	0.03
-50%	$\mathbb{P}(L_t = l)$	0.08	0.80	0.12
-25%	$\mathbb{P}(L_t = l)$	0.16	0.65	0.19
Var(a)	$\mathbb{P}(L_t = l)$	0.245	0.49	0.265
+25%	$\mathbb{P}(L_t = l)$	0.32	0.32	0.36
+50%	$\mathbb{P}(L_t = l)$	0.40	0.17	0.43
+75%	$\mathbb{P}(L_t = l)$	0.47	0.03	0.50

A second sensitivity analysis is provided to show the effects of capacity tightness, setup and backlog costs by solving the SAA problem. Let β be a factor that can take value in the set $\{0.1, 0.5, 1.0, 5, 10\}$. In this test, the capacity was generated from $\beta.C_t, \forall t \in \mathcal{T}$, the setup costs were generated from $\beta.s_t, \forall t \in \mathcal{T}$, and the backlog costs were generated from $\beta.b_i, \forall i \in \mathcal{N}$. All other parameters are generated in the same manner in section 6.1.

The test results are summarised in Table 7. The most remarkable result to emerge from the data is that the amounts of setup and backlog costs do not show a particular trend on the effect on the gap. The average percentage deviations were zero percent on average. Also, the amount of capacity and the setup costs affect the computation time while the importance of backlog costs cannot be stressed for the SAA approach.

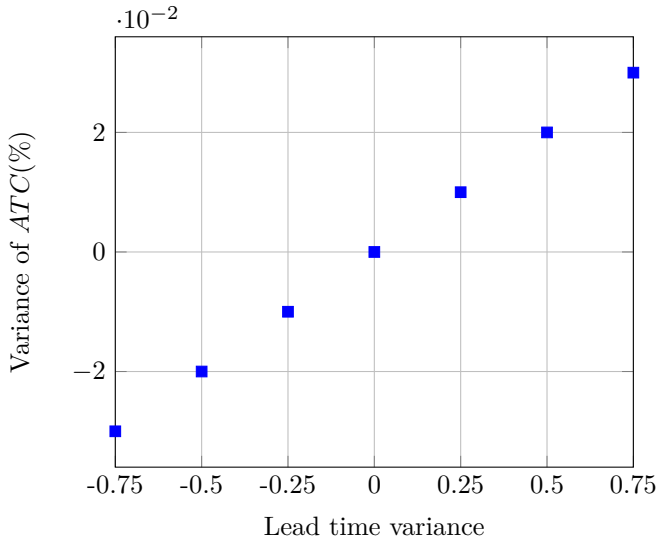


Figure 8: Effect of lead time variance on the expected total cost.

Table 7: Test results SAA approach.

	Capacity		Setup costs		Backlogging cost	
Factor (β)	Gap^*	CPU(s)	Gap^*	CPU(s)	Gap^*	CPU(s)
0.1	0.00	65.96	0.00	19.43	0.00	16.58
0.5	0.00	39.21	0.00	27.25	0.00	10.56
1	0.00	18.90	0.00	19.5	0.00	21.04
5	0.48	16.14	0.00	11.62	0.00	10.85
10	0.56	8.12	0.00	9.17	0.00	12.20
Mean	0.208	29.66	0.00	12.79	0.00	14.24

Gap^*: Percentage deviations from the lower bound or optimal solution value.

6.3. Random test instances

In order to compare and evaluate the 2S-MILP, SAA-based algorithm and MC-GA approaches more generally, the computational experiments involved different levels of problem size and complexity. The numerical examples are divided into two sets, where each set consists of several randomly generated problems.

The first set includes a disassembly system of 15 components. Since the problems in set 1 are small, they can be solved by all methods. Set 1 contains 3 different problems. The first one involves 10 time periods. The second one 20 time periods and the last one 30 time periods.

The second set consists of 5 problems of different sizes as presented in Table 8. Since, the problems in set 2 are rather large, they are solved only by the MC-GA. For each problem, the disassembly lead times were taken as independent random variables. These variables are bounded by a known interval whose upper and lower limits are randomly generated and follow a discrete uniform distribution (see Table 8).

For each problem (set 1 and 2), the disassembly system is randomly generated. Without loss of generality

the initial inventory for all items was set to 0. All the tests are run under the data listed in Table 9. Here, $D \sim U(a,b)$ means that the parameter follows the discrete uniform distribution characterized by the interval [a, b]. In the previous sub-section, we showed that, with $\Omega_1 = 1{,}000$, a SAA approach can guarantee an almost exact solution for the stochastic problem. Accordingly, SAA-based algorithm and MC-GA approaches are used with a number of samples Ω_1 equal to $1{,}000$.

Table 8: Characteristics of each problem in the second set.

Set	$[L^-, L^+]$	$\lvert\mathcal{N}\rvert$	$\lvert\mathcal{T}\rvert$
1	$D \sim U(1,20)$	20	30
2	$D \sim U(1,15)$	30	20
3	$D \sim U(1,20)$	30	30
4	$D \sim U(1,15)$	40	20
5	$D \sim U(1,20)$	40	30

Table 9: Characteristics of data set.

Parameter	Value
$D_{i,t}$	$D \sim U(10,100)$
h_i	$D \sim U(12,20)$
s_t	$D \sim U(0,1000)$
b_i	$2h_i$
u_t	$D \sim U(20,25)$
R_i	$D \sim U(1,5)$
C_t	$D \sim U(280,480)$
G	$D \sim U(5,15)$
$\lvert\Omega_1\rvert$	1,000

The good choice of the parameter values of the MC-GA makes a difference in the solutions' convergence. After preliminary tests, these parameters include the crossover rate (P_c =0.8), the mutation rate (P_m=0.1) and the population size (n_{pop}=200 chromosomes). The average MC-GA solution cost convergence process for each problem (100 run of the MC-GA) for the first set, is shown in Fig. 9. The speed of convergence is determined by the size of the chromosomes and the number of generations. More especially, for the population whose chromosomes are 10 genes, we prolong the number of generations to 200 on average, to make sure that a near optimal solution is found. The best generation occurs at the 52^{nd} generation as shown in Fig. 9(a). For chromosomes that contain 20 and 30 genes, better generation occurs on average at the 66^{th} and 97^{th} generations, respectively (see Fig. 9(b) and (c)). We prolong the end of generations to 250 and 400 for

problems 2 and 3 in set 1, respectively. For the following we define the gap between the best solution of the initial population ($bestSol_0$) and the best solution found after n generations ($bestSol_n$) using Equation (18).

$$gap = \frac{bestSol_0 - bestSol_n}{bestSol_n} \times 100 \tag{18}$$

The initial population seems to be of poor quality for each problem in set 1, which explains why the MC-GA gives an average improvement of 88 % during 200 generations on average for the first problem, 54.40 % during 250 generations on average for the second problem and 67.89 % during 400 generations on average for the third problem in set 1. For the problems in set 2, we stopped the algorithm after 500 generations. We observed that the research space is sufficiently explored and that the population is rapidly evolving.

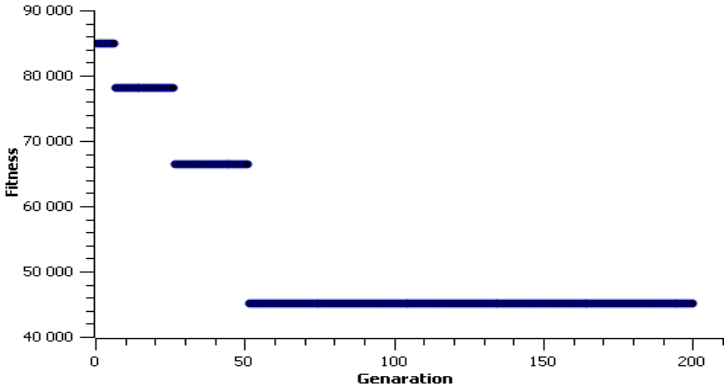
6.4. Effectiveness of the genetic algorithm approach

To validate the performance of MC-GA method, the test results are mentioned in this section. Recall that, we set the time limit as 3600(s) for the run of CPLEX in order to obtain the true optimal solution ($\mathbb{E}(CT)$) and the estimated optimal solution (ATC). For the GA execution, the computation time is limited to 600 seconds. In this case, we report the following performance measures throughout the numerical study:

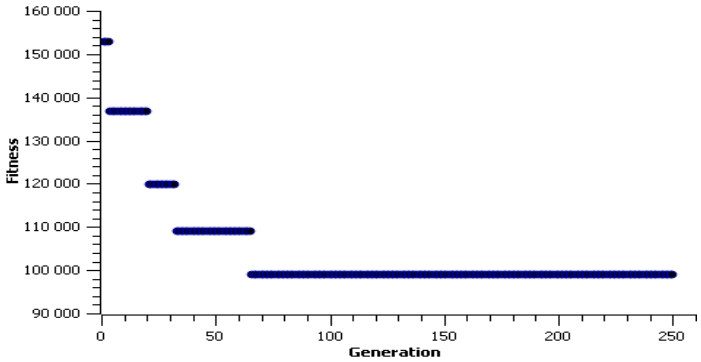
- the CPU time(s) needed to obtain optimal solutions;
- the integrality gap (G^*) given directly by CPLEX;
- the percentage deviation (G^+) between the average total cost (ATC) and the best solution given by the MC-GA ($bestSol_n$).
- the percentage deviation G^{++} between ATC/or $bestSol_n$ and the expected total cost ($\mathbb{E}(TC)$).

Table 10 presents the optimization results obtained by the 2S-MILP, SAA-based algorithm and MC-GA approaches, respectively for the problems of the first set (using the same data set). More precisely, this table gives the lead times range $[L^-, L^+]$, the computation time and the percentage deviations. In some cases, the results cannot be generated by the 2S-MILP model, which is indicated by a "-". Therefore, no comparison will be made, which is indicated by an "*".

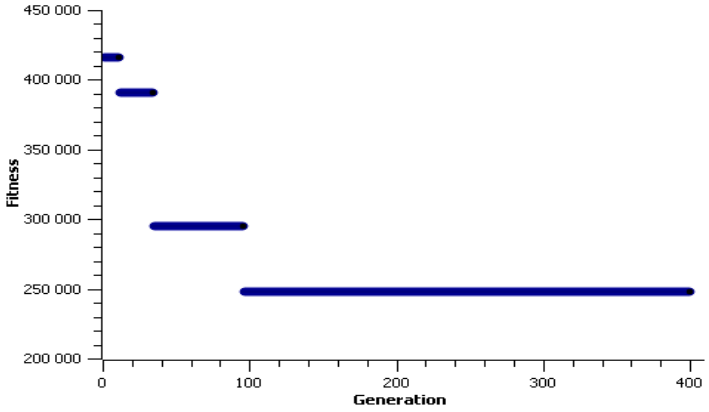
Generally, we can observe that the 2S-MILP can only solve the problem with 10 periods when the disassembly lead time range does not exceed 1. This is due to the exponential number of possible scenarios which depends strongly on the number of periods. On the one hand, the SAA algorithm solutions presented in Table 10(b), indicate that the optimal disassembly schedule can be obtained within 3600 seconds using CPLEX until a disassembly system reaches 15 items and 30 periods. This is due to the fact that the capacity restrictions on disassembly resources and an increase in the number of periods and components, increase the problem size. On the other hand, the proposed MC-GA can solve a large problems by providing a satisfactory outcome in a short computational time. Given small deviations G^+ and G^{++} (less than 0.3% and 1.10%,



(a) Problem 1



(b) Problem 2



(c) Problem 3

Figure 9: Average MC-GA convergence for each problem in set 1.

respectively for all problems in set 1), the MC-GA plays an important role in finding a solution very close to the optimal one in a very reasonable time. In summary, this research successfully formulated and solved the CDLSP with random disassembly lead times.

Table 10: Performances of the MC-GA.

(a) 2S-MILP solutions for problems in set 1

Problem	$\|\mathcal{N}\|$	$\|\mathcal{T}\|$	$\|\Omega\|$	$[L^-, L^+]$	CPU(s)	G^* (%)
1	15	10	2^{10}	[4,5]	675.22	0.00
2	15	20	15^{20}	[1,15]	-	-
3	15	30	20^{30}	[1,20]	-	-

(b) SAA approach solutions for problems in set 1

Problem	$\|\mathcal{N}\|$	$\|\mathcal{T}\|$	$\|\Omega_1\|$	$[L^-, L^+]$	CPU(s)	G^* (%)	G^{++} (%)
1	15	10	1,000	[4,5]	787.11	0.00	1.2
2	15	20	1,000	[1,15]	2615	0.00	*
3	15	30	1,000	[1,20]	3600	0.09	*

(c) MC-GA solutions for problems in set 1

Problem	$\|\mathcal{N}\|$	$\|\mathcal{T}\|$	$\|\Omega_1\|$	$[L^-, L^+]$	CPU(s)	G^+ (%)	G^{++} (%)
1	15	10	1,000	[4,5]	13.91	0.11	1.07
2	15	20	1,000	[1,15]	21.25	0.23	*
3	15	30	1,000	[1,20]	31.50	0.10	*

To show the effectiveness of the proposed MC-GA for large size tests, we deal with the second set of problems below. As explained previously, considering the random nature of the MC-GA, we carried out 100 independent runs of the same data set of each problem in the second set. For each problem, the best and worst known solutions are selected. Subsequently, the gap between these two solutions noted $Gap_{w.run}^{b.run}$ is obtained as shown in Fig. 10. This figure clearly shows that the MC-GA results are stable. In fact, the total gap from the best to the worst run does not exceed 0.6 % for all problems in set 2.

As discussed in section 5.7, the CPU times required to obtain the best solutions is limited to 600(s). Fig. 11, shows that the proposed approach can solve a disassembly system of up to 40 components and 30 periods. However, the number of periods has the most significant impact on the computation time.

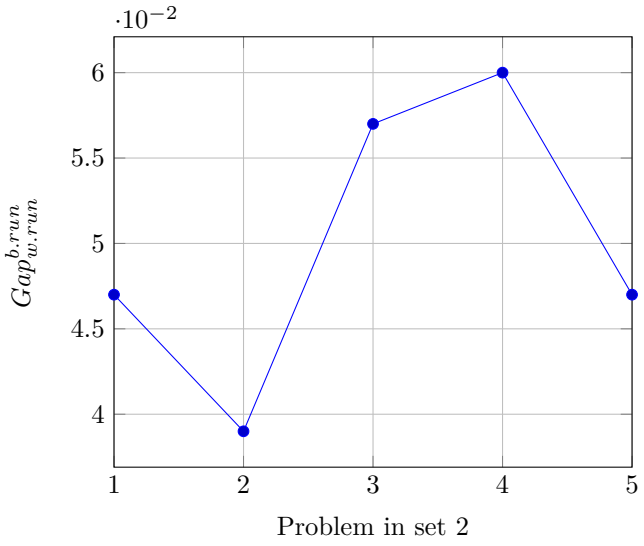


Figure 10: Average gap between the best and the worst run of each instance.

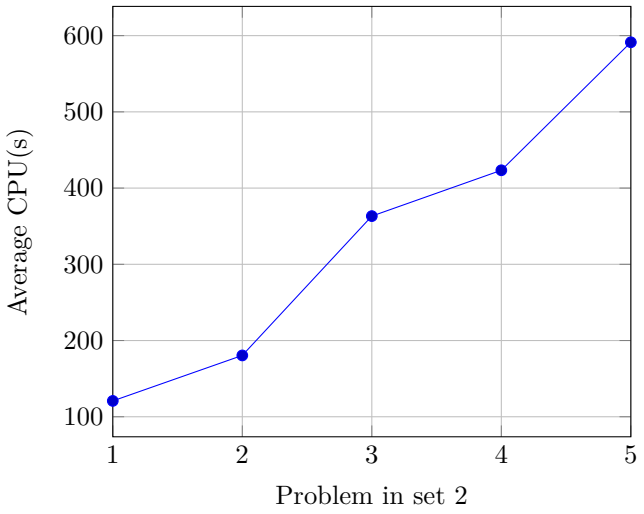


Figure 11: Average CPU(s) of each problem in set 2.

6.5. Managerial implications and insights

In reverse supply chain operation practice, disassembly lead time uncertainty is a common management issue. Guide Jr (2000) reported that the average disassembly and remanufacturing time of a typical recycled product can vary between 5.54 and 300 hours with coefficients of variation up to 5 hours. In that case, it is difficult to define the time required to complete the process of disassembling or obtaining the components. In this context, our study is unique in considering the uncertainty of disassembly lead time of the one type of end-of-life product and a two-level disassembly system in CDLSP.

The results presented in this paper reveal that a bad disassembly lead time management policy can be critical in disassembly systems. This type of uncertainty has a significant impact on the various costs of the disassembly process which will directly affect the overall operational performance of the reverse supply chain

especially in the case where the holding and the backlog costs are high.

In this study, we have used stochastic optimization models that take into account decisions to manage this type of uncertainty and allow better control of it. The proposed models and methods can be applied by decision makers to determine when and how many products should be disassembled to satisfy demand of all leaf items and to minimize the generated costs.

From a practitioner's point of view, the interest of our approach lies in the fact that it can be used in many industrial situations, as there are no assumptions on the cost functions and probability distributions of the disassembly lead time. Moreover, the proposed model assures practitioners that even in the presence of uncertainty in the DLT of an EoL product, the proposed approaches are tools that could help decision makers cope with lead time uncertainties and plan disassembly operations in the most cost effective, easy and efficient manner. However, with the availability of large amounts of data or dealing with large size issues, optimization of disassembly processes could be done in real time. In this case, these models could also be reinforced by techniques derived from Artificial Intelligence such as for example Machine learning, data mining which will allow the extraction of probability distributions to guarantee the performance of the models and methods proposed. In addition, a Digital twins concept and related approaches by collecting and storing large amounts of data in real-time and throughout disassembly processes could be studied.

7. Conclusion

This paper addresses the capacitated disassembly lot-sizing problem under given demand and uncertain disassembly lead times. The probability distribution of the disassembly lead time is assumed to be known and bounded. The planning problem identifies how much EoL product to disassemble during each period in order to minimize the expected total cost. Using a scenario-based approach to express stochastic disassembly lead times, the considered problem is modelled as a two-stage stochastic Mixed-Integer Linear Program. To alleviate the scalability issues, a sample average approximation-based solution algorithm is suggested. In addition, to solve large scale problems, we propose a basic genetic algorithm. Experimental results show the effectiveness of the proposed models and the convergence of the resulting Monte Carlo sampling. Finally, based on our analysis, we have generated an important managerial implications

This paper could be extended in several ways. As for the lot-sizing (LS) perspective, most works treat the problem under uncertainty on the two-level DLS problems. Since most of the real life, DLS are multi-levels. A promising future research area could be solving the multi-level and multiple type of EoL product disassembly system. For the GA perspective, the hybridization of the genetic algorithm with other meta-heuristics or heuristic optimization techniques has caught the attention of many researchers in LS literature.

References

Agrawal, S., & Tiwari, M. (2008). A collaborative ant colony algorithm to stochastic mixed-model u-shaped disassembly line balancing and sequencing problem. *International journal of production research*, *46*, 1405–1429.

Barba-Gutiérrez, Y., & Adenso-Díaz, B. (2009). Reverse mrp under uncertain and imprecise demand. *The International Journal of Advanced Manufacturing Technology*, *40*, 413–424.

Barba-Gutiérrez, Y., Adenso-Diaz, B., & Gupta, S. M. (2008). Lot sizing in reverse mrp for scheduling disassembly. *International Journal of Production Economics*, *111*, 741–751.

Belkhamsa, M., Jarboui, B., & Masmoudi, M. (2018). Two metaheuristics for solving no-wait operating room surgery scheduling problem under various resource constraints. *Computers & Industrial Engineering*, *126*, 494–506.

Ben-Ammar, O., Bettayeb, B., & Dolgui, A. (2019). Optimization of multi-period supply planning under stochastic lead times and a dynamic demand. *International Journal of Production Economics*, *218*, 106–117.

Ben-Ammar, O., Dolgui, A., & Wu, D. D. (2018). Planned lead times optimization for multi-level assembly systems under uncertainties. *Omega*, *78*, 39–56.

Bitran, G. R., & Yanasse, H. H. (1982). Computational complexity of the capacitated lot size problem. *Management Science*, *28*, 1174–1186.

Che, Z., & Chiang, C.-J. (2010). A modified pareto genetic algorithm for multi-objective build-to-order supply chain planning with product assembly. *Advances in Engineering Software*, *41*, 1011–1022.

Choi, Y.-K., Lee, D. M., & Cho, Y. B. (2009). An approach to multi-criteria assembly sequence planning using genetic algorithms. *The International Journal of Advanced Manufacturing Technology*, *42*, 180–188.

Dini, G., Failli, F., & Santochi, M. (2001). A disassembly planning software system for the optimization of recycling processes. *Production Planning & Control*, *12*, 2–12.

Efendigil, T., Önüt, S., & Kongar, E. (2008). A holistic approach for selecting a third-party reverse logistics provider in the presence of vagueness. *Computers & Industrial Engineering*, *54*, 269–287.

Fallah-Jamshidi, S., Karimi, N., & Zandieh, M. (2011). A hybrid multi-objective genetic algorithm for planning order release date in two-level assembly system with random lead times. *Expert Systems with Applications*, *38*, 13549–13554.

Fang, C., Liu, X., Pardalos, P. M., Long, J., Pei, J., & Zuo, C. (2017). A stochastic production planning problem in hybrid manufacturing and remanufacturing systems with resource capacity planning. *Journal of Global Optimization*, *68*, 851–878.

Fishman, G. (1996). Monte carlo: concepts, algorithms, and applications. *Science Business Media*, .

Florian, M., Lenstra, J. K., & Rinnooy Kan, A. (1980). Deterministic production planning: Algorithms and complexity. *Management science*, *26*, 669–679.

Gao, N., & Chen, W. (2008). A genetic algorithm for disassembly scheduling with assembly product structure. In *2008 IEEE International Conference on Service Operations and Logistics, and Informatics* (pp. 2238–2243). IEEE volume 2.

Godichaud, M., & Amodeo, L. (2018). Economic order quantity for multistage disassembly systems. *International Journal of Production Economics*, *199*, 16–25.

Godichaud, M., & Amodeo, L. (2019). Eoq inventory models for disassembly systems with disposal and lost sales. *International Journal of Production Research*, *57*, 5685–5704.

Godichaud, M., Amodeo, L., & Hrouga, M. (2015). Metaheuristic based optimization for capacitated disassembly lot sizing problem with lost sales. In *2015 International Conference on Industrial Engineering and Systems Management (IESM)* (pp. 1329–1335). IEEE.

Goren, H. G., Tunali, S., & Jans, R. (2010). A review of applications of genetic algorithms in lot sizing. *Journal of Intelligent Manufacturing*, *21*, 575–590.

Guide Jr, V. D. R. (2000). Production planning and control for remanufacturing: industry practice and research needs. *Journal of operations Management*, *18*, 467–483.

Guiras, Z., Turki, S., Rezg, N., & Dolgui, A. (2019). Optimal maintenance plan for two-level assembly system and risk study of machine failure. *International Journal of Production Research*, *57*, 2446–2463.

Gungor, A., & Gupta, S. M. (1998). Disassembly sequence planning for products with defective parts in product recovery. *Computers & Industrial Engineering*, *35*, 161–164.

Güngör, A., & Gupta, S. M. (2001). Disassembly sequence plan generation using a branch-and-bound algorithm. *International Journal of Production Research*, *39*, 481–509.

Gupta, S., & Taleb, K. (1994). Scheduling disassembly. *The International Journal of Production Research*, *32*, 1857–1866.

Gupta, S. M., & Lambert, A. F. (2016). *Disassembly modeling for assembly, maintenance, reuse and recycling*. CRC press.

Habibi, M. K. K., Battaïa, O., Cung, V.-D., Dolgui, A., & Tiwari, M. K. (2019). Sample average approximation for multi-vehicle collection–disassembly problem under uncertainty. *International Journal of Production Research*, *57*, 2409–2428.

Han, H.-J., Yu, J.-M., & Lee, D.-H. (2013). Mathematical model and solution algorithms for selective disassembly sequencing with multiple target components and sequence-dependent setups. *International Journal of Production Research*, *51*, 4997–5010.

Hnaien, F., Delorme, X., & Dolgui, A. (2009). Genetic algorithm for supply planning in two-level assembly systems with random lead times. *Engineering Applications of Artificial Intelligence*, *22*, 906–915.

Hnaien, F., Delorme, X., & Dolgui, A. (2010). Multi-objective optimization for inventory control in two-level assembly systems under uncertainty of lead times. *Computers & operations research*, *37*, 1835–1843.

Ilgin, M. A., & Gupta, S. M. (2010). Environmentally conscious manufacturing and product recovery (ecmpro): A review of the state of the art. *Journal of environmental management*, *91*, 563–591.

Inderfurth, K., & Langella, I. M. (2006). Heuristics for solving disassemble-to-order problems with stochastic yields. *OR Spectrum*, *28*, 73–99.

Inderfurth, K., Vogelgesang, S., & Langella, I. M. (2015). How yield process misspecification affects the solution of disassemble-to-order problems. *International Journal of Production Economics*, *169*, 56–67.

Jeunet, J., Della Croce, F., & Salassa, F. (2019). Heuristic solution methods for the selective disassembly sequencing problem under sequence-dependent costs. *IFAC-PapersOnLine*, *52*, 1908–1913.

Ji, X., Zhang, Z., Huang, S., & Li, L. (2016). Capacitated disassembly scheduling with parts commonality and start-up cost and its industrial application. *International Journal of Production Research*, *54*, 1225–1243.

Kang, K.-W., Doh, H.-H., Park, J.-H., & Lee, D.-H. (2016). Disassembly leveling and lot sizing for multiple product types: a basic model and its extension. *The International Journal of Advanced Manufacturing Technology*, *82*, 1463–1473.

Kim, D.-H., Doh, H.-H., & Lee, D.-H. (2018a). Multi-period disassembly levelling and lot-sizing for multiple product types with parts commonality. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *232*, 867–878.

Kim, D.-H., & Lee, D.-H. (2011). A heuristic for multi-period disassembly leveling and scheduling. In *2011 IEEE/SICE international symposium on system integration (SII)* (pp. 762–767). IEEE.

Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2006a). A lagrangean heuristic algorithm for disassembly scheduling with capacity constraints. *Journal of the Operational Research Society*, *57*, 1231–1240.

Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2006b). Two-phase heuristic for disassembly scheduling with multiple product types and parts commonality. *International Journal of Production Research*, *44*, 195–212.

552 Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2007). Disassembly scheduling: literature review and future
553 research directions. *International Journal of Production Research*, *45*, 4465–4484.

554 Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2008). An exact algorithm for two-level disassembly scheduling.
555 *Journal of the Korean Institute of Industrial Engineers*, *34*, 414–424.

556 Kim, H.-J., Lee, D.-H., Xirouchakis, P., & Kwon, O. (2009). A branch and bound algorithm for disassembly
557 scheduling with assembly product structure. *Journal of the Operational Research Society*, *60*, 419–430.

558 Kim, H.-J., Lee, D.-H., Xirouchakis, P., & Züst, R. (2003). Disassembly scheduling with multiple product
559 types. *CIRP Annals*, *52*, 403–406.

560 Kim, H.-J., & Xirouchakis, P. (2010). Capacitated disassembly scheduling with random demand. *International*
561 *Journal of Production Research*, *48*, 7177–7194.

562 Kim, H.-W., Park, C., & Lee, D.-H. (2018b). Selective disassembly sequencing with random operation times
563 in parallel disassembly environment. *International Journal of Production Research*, *56*, 7243–7257.

564 Kim, J.-G., Jeon, H.-B., Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2005). Capacitated disassembly schedul-
565 ing: minimizing the number of products disassembled. In *International Conference on Computational*
566 *Science and Its Applications* (pp. 538–547). Springer.

567 Kleywegt, A. J., Shapiro, A., & Homem-de Mello, T. (2002). The sample average approximation method for
568 stochastic discrete optimization. *SIAM Journal on Optimization*, *12*, 479–502.

569 Lamiri, M., Xie, X., Dolgui, A., & Grimaud, F. (2008). A stochastic model for operating room planning with
570 elective and emergency demand for surgery. *European Journal of Operational Research*, *185*, 1026–1037.

571 Langella, I. M. (2007). Heuristics for demand-driven disassembly planning. *Computers & Operations Re-*
572 *search*, *34*, 552–577.

573 Lee, D.-H., Kim, H., Choi, G., & Xirouchakis, P. (2004). Disassembly scheduling: integer programming mod-
574 els. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*,
575 *218*, 1357–1372.

576 Lee, D.-H., & Xirouchakis, P. (2004). A two-stage heuristic for disassembly scheduling with assembly product
577 structure. *Journal of the Operational Research Society*, *55*, 287–297.

578 Lee, D.-H., Xirouchakis, P., & Zust, R. (2002). Disassembly scheduling with capacity constraints. *CIRP*
579 *Annals*, *51*, 387–390.

580 Li, B., Ding, L., Rajai, M., Hu, D., & Zheng, S. (2018). Backtracking algorithm-based disassembly sequence
581 planning. *Procedia CIRP*, *69*, 932–937.

Li, K., Liu, Q., Xu, W., Liu, J., Zhou, Z., & Feng, H. (2019). Sequence planning considering human fatigue for human-robot collaboration in disassembly. *Procedia CIRP*, *83*, 95–104.

Liu, J., Zhou, Z., Pham, D. T., Xu, W., Ji, C., & Liu, Q. (2020). Collaborative optimization of robotic disassembly sequence planning and robotic disassembly line balancing problem using improved discrete bees algorithm in remanufacturing. *Robotics and Computer-Integrated Manufacturing*, *61*, 101829.

Liu, K., & Zhang, Z.-H. (2018). Capacitated disassembly scheduling under stochastic yield and demand. *European Journal of Operational Research*, *269*, 244–257.

Mak, W.-K., Morton, D. P., & Wood, R. K. (1999). Monte carlo bounding techniques for determining solution quality in stochastic programs. *Operations research letters*, *24*, 47–56.

Neuendorf, K.-P., Lee, D.-H., Kiritsis, D., & Xirouchakis, P. (2001). Disassembly scheduling with parts commonality using petri nets with timestamps. *Fundamenta Informaticae*, *47*, 295–306.

Norkin, V. I., Pflug, G. C., & Ruszczyński, A. (1998). A branch and bound method for stochastic global optimization. *Mathematical programming*, *83*, 425–450.

Piewthongngam, K., Chatavithee, P., & Apichottanakul, A. (2019). Disassembly scheduling for the meat processing industry with product perishability. *Journal of Advanced Manufacturing Systems*, *18*, 447–467.

Pour-Massahian-Tafti, M., Godichaud, M., & Amodeo, L. (2020a). Disassembly eoq models with price-sensitive demands. *Applied Mathematical Modelling*, .

Pour-Massahian-Tafti, M., Godichaud, M., & Amodeo, L. (2020b). New models and efficient methods for single-product disassembly lot-sizing problem with surplus inventory decisions. *International Journal of Production Research*, (pp. 1–21).

Prakash, P., Ceglarek, D., & Tiwari, M. (2012). Constraint-based simulated annealing (cbsa) approach to solve the disassembly scheduling problem. *The International Journal of Advanced Manufacturing Technology*, *60*, 1125–1137.

Rai, R., Rai, V., Tiwari, M., & Allada, V. (2002). Disassembly sequence generation: a petri net based heuristic approach. *International Journal of Production Research*, *40*, 3183–3198.

Ren, Y., Meng, L., Zhang, C., Zhao, F., Saif, U., Huang, A., Mendis, G. P., & Sutherland, J. W. (2020). An efficient metaheuristics for a sequence-dependent disassembly planning. *Journal of Cleaner Production*, *245*, 118644.

Sakiani, R., Ghomi, S. F., & Zandieh, M. (2012). Multi-objective supply planning for two-level assembly systems with stochastic lead times. *Computers & Operations Research*, *39*, 1325–1332.

Slama, I., Ben-Ammar, O., Dolgui, A., & Masmoudi, F. (2020). New mixed integer approach to solve a multi-level capacitated disassembly lot-sizing problem with defective items and backlogging. *Journal of Manufacturing Systems*, *56*, 50–57.

Slama, I., Ben-Ammar, O., Masmoudi, F., & Dolgui, A. (2019). Disassembly scheduling problem: literature review and future research directions. *IFAC-PapersOnLine*, *52*, 601–606.

Taleb, K. N., Gupta, S. M., & Brennan, L. (1997a). Disassembly of complex product structures with parts and materials commonality. *Production Planning & Control*, *8*, 255–269.

Taleb, K. N., Gupta, S. M., & Brennan, L. (1997b). Disassembly of complex product structures with parts and materials commonality. *Production Planning & Control*, *8*, 255–269.

Tian, X., & Zhang, Z.-H. (2019). Capacitated disassembly scheduling and pricing of returned products with price-dependent yield. *Omega*, *84*, 160–174.

Tiwari, M. (2005). Solving a dissassembly line balancing problem with task failure using a psycho-clonal algorithm. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (pp. 393–399). volume 47411.

Tseng, H.-E., Chang, C.-C., Lee, S.-C., & Huang, Y.-M. (2018). A block-based genetic algorithm for disassembly sequence planning. *Expert Systems with Applications*, *96*, 492–505.

Tseng, H.-E., Chang, C.-C., Lee, S.-C., & Huang, Y.-M. (2019). Hybrid bidirectional ant colony optimization (hybrid baco): An algorithm for disassembly sequence planning. *Engineering Applications of Artificial Intelligence*, *83*, 45–56.

Ullerich, C., & Buscher, U. (2013). Flexible disassembly planning considering product conditions. *International Journal of Production Research*, *51*, 6209–6228.

Xu, W., Tang, Q., Liu, J., Liu, Z., Zhou, Z., & Pham, D. T. (2020). Disassembly sequence planning using discrete bees algorithm for human-robot collaboration in remanufacturing. *Robotics and Computer-Integrated Manufacturing*, *62*, 101860.

Zhang, X. F., & Zhang, S. Y. (2010). Product cooperative disassembly sequence planning based on branch-and-bound algorithm. *The International Journal of Advanced Manufacturing Technology*, *51*, 1139–1147.