



HAL
open science

Stochastic program for disassembly lot-sizing under uncertain component refurbishing lead times

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

► **To cite this version:**

Ilhem Slama, Oussama Ben-Ammar, Simon Thevenin, Alexandre Dolgui, Faouzi Masmoudi. Stochastic program for disassembly lot-sizing under uncertain component refurbishing lead times. *European Journal of Operational Research*, 2022, 10.1016/j.ejor.2022.03.025 . hal-03614823

HAL Id: hal-03614823

<https://hal.science/hal-03614823>

Submitted on 1 Jun 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Stochastic program for disassembly lot-sizing under uncertain component refurbishing lead times

Ilhem Slama^a, Oussama Ben-Ammar^{b, *}, Simon Thevenin^c, Alexandre Dolgui^c, Faouzi Masmoudi^d

^a LINEACT CESI campus de Lyon, France

^b EuroMov Digital Health in Motion, Univ Montpellier, IMT Mines Ales, Ales, France

^c IMT Atlantique, LS2N-CNRS, La Chantrerie, 4 rue Alfred Kastler - B.P. 20722, Nantes 44307, France

^d Engineering School of Sfax, Laboratory of Mechanic, Modeling and Production, Univ Sfax, Tunisia

A B S T R A C T

Planning disassembly operations for a given demand in components is challenging in practice because the quality of recovered components is very uncertain, and thus the duration of refurbishing operations is unpredictable. In this paper, we address the capacitated disassembly lot-sizing problems under uncertain refurbishing durations. More precisely, we consider a two-level disassembly system with a single type of end-of-life product, a dynamic demand, and stochastic refurbishing lead times for all components. To deal with the static decision frameworks, this problem is modeled as a two-stage stochastic Mixed-Integer Linear Program (MILP), where the objective is to minimize the expected total cost. To alleviate the scalability issues, we propose a reformulation of the inventory constraint that significantly reduces the number of scenarios. In addition, to solve large scale problems, we couple this reformulation with Monte-Carlo sampling. We provide a rolling horizon approach to deal with the static decision framework, where disassembly decisions are updated when new information unfolds. Experimental results show the effectiveness of the proposed models and the convergence of the resulting Sample Average Approximation (SAA) estimator.

Keywords:

Combinatorial optimization
Capacitated disassembly lot-sizing
Stochastic refurbishing lead time
Monte-Carlo sampling
Rolling horizon strategy

1. Introduction

Governments around the world are imposing more and more environmental regulations on manufacturers and suppliers to minimize waste and recover resources. The directive on waste electrical and electronic equipment (directive 2002/96/EC) is a good example of the implementation of an environmental management policy (Wang & Huang, 2013). Members of the European Union were invited to set up different systems for returning products at the end of their life, reducing waste, and carrying out recovery and recycling processes. All of these actions aim to reduce electrical equipment wastes and their negative environmental impacts. According to Pini, Neri, & Ferrari (2018), Europe produces up to 3 billion tons of waste every year. In this context, recovery processes provide financial opportunities, and they allow manufacturers to engage in sustainable practices by extending the life of an end-of-life (EoL) product. In fact, the refurbishing activity is not only attractive from an environmental perspective, it can provide economic benefits by

obtaining valuable materials for reuse or energy production. Refurbished components, which typically undergo a thorough cleaning process are equivalent in performance to the new ones, yet they sell for only 40% of the original price, moreover the cycle of resale and refurbishing can be repeated several times (Pearce John, 2009). Currently, in the United States, refurbishing is \$50 billion industry with direct employment of about 480,000 in 73,000 firms' (Gutowski, Sahni, & Boustani, 2011).

The recovery process encompasses all activities related to the collection, disassembly, refurbishing, repair, recycling, disposal, of EoL products. Once collected, they are disassembled into components, and these components are then cleaned and tested. Based on their condition, these components undergo one of the two value recovery activities (refurbishing, or recycling) or are disposed of Langella (2007). As the reverse flow grows, many firms seek opportunities to reduce EoL product management costs. Among various tasks in the recovery process, disassembly and refurbishing are the most significant operations in terms of costs. In addition, these operations are difficult to manage due to the uncertainty of various refurbishing parameters such as quantity, quality, and timing.

* Corresponding author.

E-mail address: oussama.ben-ammam@mines-ales.fr (O. Ben-Ammar).

According to a review proposed by Jaehn (2016), the EoL product recovery process that considers random input is a promising area of research for sustainable operations in production. After disassembly, the components undergo several renovation processes (such as repair, cleaning, etc.) to meet the required quality level, and this renovation process can take several weeks. In a remanufacturing context, these refurbishing lead times (RLT) are difficult to predict, because the quality of the components is only known once the EoL item is disassembled. Managing the uncertainty of refurbishing lead times plays an important role in disassembly planning and inventory management since bad management can cause stock-outs and unnecessary stocks.

In this paper, we provide a methodology for disassembly planning of EoL products under stochastic refurbishing time for components, known component demand, and when the capacity of the disassembly line is limited. The goal is to decide the size of the lots to disassemble and to plan the amount of overtime required on the disassembly line. According to the classification proposed by Ji, Zhang, Huang, & Li (2016), the studied problem enters into the category of a Capacitated Disassembly Lot Sizing (CDLS) problem that considers the best use of disassembly resources to satisfy the demand on components over a given planning horizon. Note that we assume the disassembly duration is known, and it is shorter than a period, whereas the refurbishing duration may last several weeks and is unknown. In this setting, the disassembly duration corresponds to process duration, and it consumes the production capacity. In the production planning literature, the time difference between the start of processing of an operation and the availability of the resulting parts is called lead time. Therefore, the considered problem is a CDLS under uncertain refurbishing lead times.

As we aim to extend the current knowledge about DLS problem resolution by integrating the uncertainty of refurbishing lead times, we consider a single type of EoL product, a two-level disassembly system, and a multi-period planning. In addition, the customer's demand and the disassembly duration are known, whereas the refurbishing lead times are discrete random variables that follow known probability distributions. This tactical problem aims to determine the optimal quantity of EoL products to disassemble in each period over a given planning horizon which guarantees the best trade-off between costs related to disassembly operations and the sum of expected inventory and backlog costs. The contributions and innovations in this study are the following:

1. We formulate the considered problem as a stochastic mathematical program. The resulting MILP accounts for a set of possible scenarios that represent possible refurbishing lead times for components obtained by disassembly. The proposed model is flexible enough to account for the case where different disassembly lots cross in time due to the uncertain refurbishing lead times. In this situation, if the disassembly of a lot *A* occurs before the disassembly of a lot *B*, the components resulting from *B* may be received before the components of *A*. This is an important contribution since a large part of the literature on classical planning under lead time uncertainty requires the restrictive assumption that orders do not cross in time. To the best of our knowledge, we are the first to propose a method able to account for order crossover in the DLS problem.
2. Our experiments show that CPLEX cannot solve instances with more than 30 components and 10 periods in less than 3600 seconds of computation, because the number of scenarios becomes too large. To get around this obstacle, we developed a scenario aggregation technique. More precisely, we propose a new formulation of the inventory constraint that significantly reduces the number of scenarios. The resulting model significantly improves the scalability of the approach, and it can solve instances with a large number of components and peri-

ods when order crossover is forbidden, or if the refurbishing lead time range (i.e., the difference between the maximum and the minimum refurbishing lead time value for a component) is lower than 6 periods when order crossovers can occur.

3. Even though the scenario aggregation (SA) approach reduces the number of scenarios drastically, this number remains exponential in the refurbishing lead times range. As a consequence, the resulting method cannot solve instances with a large range of refurbishing lead times. To solve these large-scale problems, we couple the scenario aggregation technique with Monte-Carlo (MC) sampling.
4. To deal with the dynamic decision frameworks, where the disassembly plan is updated when new information unfolds, we employ a rolling horizon strategy.
5. Extensive computational experiments assess the effectiveness of the proposed approaches. In particular, the resulting sampling method provides a good approximation, with an average percentage deviation from the optimal solutions of less than 0.5%. In addition, our tests show the convergence of the Sample Average Approximation (SAA) estimator.

This paper is divided into six sections. Section 2 provides a brief overview of the literature review. Section 3 gives the problem description and formulation. The proposed approaches are described in Section 4. Section 5 gives numerical results. Finally, conclusions and some perspectives for future work are drawn in Section 6.

2. Literature review

Since the seminal work of Gupta & Taleb (1994), the DLS problems have attracted a lot of attention, and recent reviews are presented in Kim, Lee, & Xirouchakis (2007) and Slama, Ben-Ammar, Masmoudi, & Dolgui (2019a). Gupta & Lambert (2016) highlighted the importance of disassembly lot sizing problems. Compared to the assembly process, the disassembly process is characterized by more complex operational and physical properties (Gungor & Gupta, 1999; Slama, Ben-Ammar, Dolgui, & Masmoudi, 2020a). The most important difference is the number of demand sources. In an assembly system, the parts tend to converge to a single demand source related to the final product. Under a disassembly system, parts tend to diverge to multiple demand sources corresponding to each disassembled component. For more details on these divergences, the interested reader is referred to Gungor & Gupta (1999) and Kim et al. (2007).

Different variants of the DLS problem exist, and they differ according to the product structure (single or multi EoL products, two or multi-echelon bill of materials (BOM), with or without parts commonalities) and by the consideration of disassembly process resources (capacitated or uncapacitated). Most of the literature on disassembly systems concerns the deterministic uncapacitated problem (e.g., Godichaud & Amodeo, 2018; 2019; Gupta & Lambert, 2016; Kang, Doh, Park, & Lee, 2016; Kim, Doh, & Lee, 2018a; Kim, Lee, & Xirouchakis, 2006b; Kim, Lee, Xirouchakis, & Kwon, 2009; Langella, 2007; Lee & Xirouchakis, 2004; Neuendorf, Lee, Kiritsis, & Xirouchakis, 2001; Pour-Massahian-Tafti, Godichaud, & Amodeo, 2019a; 2019b; 2019c; 2020a; 2020b; Prakash, Ceglarek, & Tiwari, 2012; Taleb, Gupta, & Brennan, 1997), while a small number of studies focus on the deterministic capacitated version (e.g., Ji et al., 2016; Kim, Lee, & Xirouchakis, 2006a; Kim, Jeon, Kim, Lee, & Xirouchakis, 2005; Lee, Xirouchakis, & Zust, 2002; Slama et al., 2020a; Ullerich & Buscher, 2013). Table 1 summarizes the literature on the DLS problem, and indicates the resolution approach, the BOM structure, whether capacity is considered, and the type of uncertainties.

In practice, the product recovery process is sensitive to various sources of uncertainty, such as uncertain demand from cus-

Table 1
Summary of relevant literature.

Authors	Optimisation approach	#levels	Cap.	PC	Uncertainty
Gupta & Taleb (1994)	RMRP	Multi			
Taleb et al. (1997)	RMRP	Multi			-
Lee & Xirouchakis (2004)	Heuristic	Multi		✓	-
Inderfurth & Langella (2006)	Heuristic	Two			Yield
Kim et al. (2006b)	Heuristic	Multi	✓		-
Kim et al. (2006a)	LH	Multi	✓		-
Kim, Lee, & Xirouchakis (2008)	DP	Two			-
Kim et al. (2009)	Branch and bound	Multi	✓		-
Kim & Xirouchakis (2010)	LH	Two	✓		Demand
Kim & Lee (2011)	Heuristic	Multi			-
Prakash et al. (2012)	CBSA	Multi		✓	-
Ji et al. (2016)	LH	Two		✓	-
Godichaud, Amodeo, & Hrouga (2015)	GA	Multi	✓		-
Fang et al. (2017)	LH	Two			Demand
Kim et al. (2018a)	Heuristic	Multi			-
Godichaud & Amodeo (2018)	EOQ model	Two			-
Liu & Zhang (2018)	Outer-approximation	Two			Demand, yield
Tian & Zhang (2019)	PSO	Two			-
Slama et al. (2019b)	S-MIP	Two			DLT
Piewthongngam, Chatavithree, & Apichottanakul (2019)	MILP	Two		✓	-
Godichaud & Amodeo (2019)	EOQ model	Multi			-
Pour-Massahian-Tafti et al. (2019a)	AF, DF	Two			-
Pour-Massahian-Tafti, Godichaud, & Amodeo (2019b)	Heuristics	Two			-
Pour-Massahian-Tafti, Godichaud, & Amodeo (2019c)	Heuristics	Two			-
Slama et al. (2020a)	MIP	Multi	✓	✓	-
Pour-Massahian-Tafti, Godichaud, & Amodeo (2020a)	EOQ model	Two			-
Pour-Massahian-Tafti et al. (2020b)	Heuristics	Two		✓	-
Slama et al. (2020c)	SAA	Two	✓		DLT
Slama et al. (2021b)	GA	Two	✓		DLT
Slama, Ben-Ammar, & Dolgui (2021a)	Analytical model, recursive formula	Multi			DLT
Current paper	SA-MILP	Two	✓		RLT

Cap., Capacity; PC, Parts Commonalities; NP, Number of Disassembled products; RMRP, Reverse Material Requirements Planning; LP, Linear Programming; DP, Dynamic Programming; LS-RMRP, Lot-Sizing-RMRP; CBSA, Constraint-Based Simulated Annealing; LH, Lagrangian Heuristics; EOQ, Economic Order Quantity; AF, Aggregate Formulation; DF, Disaggregate Formulation; GA, Genetic Algorithm; SA-MILP, Scenario Aggregation-Mixed Integer Linear Programming.

tomers, recovery rates, disassembly lead times, etc. [Guide & Srivastava \(1997\)](#). As in supply planning for assembly systems, these uncertainties create disruptions in the disassembly plan, and leads to unmet customer demand. Many researchers have applied stochastic algorithms to the DLS problem (e.g., [Tarim, Dogru, Özen, & Rossi, 2011](#); [Tarim & Smith, 2008](#); [Vargas, 2009](#)), and they focus on demand, disassembly yield, and disassembly lead time (DLT) uncertainty ([Suzanne, Absi, & Borodin, 2020](#)). Therefore, to position our research in the existing literature, we successively review the workson stochastic DLS problems under demand, yield, and disassembly lead time uncertainty.

Demand uncertainty: [Barba-Gutiérrez & Adenso-Díaz \(2009\)](#) propose a fuzzy logic approach for the single EoL item, multi-period, multi-echelon DLS problem under stochastic demand. In this work, every unsatisfied demand in each period is back-ordered and postponed for future periods. [Kim & Xirouchakis \(2010\)](#) consider the capacitated multi-item DLS problem where each product has a two-echelon BOM structure. To solve the problem, the authors develop a stochastic inventory model, and they propose a Lagrangian relaxation-based heuristic that optimizes the sum of expected inventory holding, setup, and penalty costs. [Fang et al. \(2017\)](#); [Quezada, Gicquel, Kedad-Sidhoum, & Vu \(2020\)](#); [Wang & Huang \(2013\)](#) consider the multi-level, multi-product, multi-period stochastic lot-sizing problem in remanufacturing systems. [Wang & Huang \(2013\)](#) propose a two-stage stochastic programming model to find a compromise between the solution robustness and the expected cost. [Fang et al. \(2017\)](#) formulate a multi-stage stochastic programming approach, and they propose a Lagrangian relaxation algorithm to solve the problem. Recently, [Quezada et al. \(2020\)](#) focused on a multi-echelon system with disassembly, refurbishing, and reassembly operations. [Quezada et al. \(2020\)](#) extend the multi-

stage stochastic programming approach proposed by [Fang et al. \(2017\)](#) by considering the uncertainties related to the quantity and quality of returned products, the customer demand, and the production costs. To solve the problem, the authors develop an exact solution approach based on a branch-and-cut algorithm.

The yield uncertainty occurs when the number of components obtained by disassembling a unit of product is uncertain ([Inderfurth, Vogelgesang, & Langella, 2015](#)). To the best of our knowledge, only two papers consider the DLS problem under yield uncertainty. These two papers are restricted to the multi-products case with two-echelon BOMs in a one-period disassembly to-order (DTO) environment. [Inderfurth & Langella \(2006\)](#) give a heuristic to reduce the expected disposal, purchasing, and disassembly operation costs, whereas [Inderfurth et al. \(2015\)](#) propose a mathematical model for the specific case with three end-components and two sub-assemblies. The authors study the impact of process specification on the solution of the DTO problem. Finally, a statistical analysis shows that the yield uncertainty has a large impact in engine remanufacturing.

Uncertainty of demand and yield: to the best of our knowledge, [Liu & Zhang \(2018\)](#) is the only publication investigating simultaneously the uncertainty of component demand and disassembly yield. The yield uncertainty is modeled with a uniform distribution, whereas the demand follows a normal distribution. The authors provide a mixed integer nonlinear programming (MINLP) for the multi-period and two echelon disassembly system, and they solve it with an outer approximation-based solution algorithm.

Uncertainty of disassembly lead time (DLT): while yield and demand uncertainties impact the disassembly operation planning, remanufacturing facilities also face uncertain disassembly lead times. In fact, [Guide \(2000\)](#) indicates that the average time to re-

manufacture and disassemble a recycled EoL product varies from a low level of 5.54 hours to a high level of 300 hours, with coefficients of variation as high as 5.0. The literature on disassembly planning under random DLT is scarce. [Slama, Ben-Ammar, Masmoudi, & Dolgui \(2019b\)](#) propose a stochastic model for the multi-period, single product type, and two-level disassembly system. Recently, [Slama, Ben-Ammar, Dolgui, & Masmoudi \(2020b\)](#) proposed a generalization of the discrete Newsboy formulae to find the optimal release date when the time of disassembling the EoL product is a random variable. In this paper, the authors investigated the effect of uncertainty of DLT on the disassembly planning problem. Their analyses show that disassembly lead time variability has a significant effect on the system performance. The limitation of this work is that it considers a mono-period disassembly planning problem with unlimited capacity. [Slama, Ben-Ammar, Dolgui, & Masmoudi \(2020c\)](#) extends [Slama et al. \(2019b\)](#) by considering the capacitated case. Recently, [Slama, Ben-Ammar, Dolgui, & Masmoudi \(2021b\)](#) proposed a genetic algorithm to solve the multi-period, single EoL product, and two-level BOM. However, the authors assume the disassembly lead time is uncertain, where all the components of an end-item are received in the same period. On the contrary, the present work considers refurbishing lead time uncertainty, where the uncertainty affects each component independently.

As discussed in [Kim et al. \(2007\)](#), the disassembly process is very sensitive to the uncertainty of refurbishing lead time of each component. [Guide \(2000\)](#) states that this uncertainty can interrupt the delivery of disassembled parts, and it increases the inventory and backlog costs. Nevertheless, to the best of our knowledge, the uncertainty of the refurbishing lead time (i.e., the time required for component recovery) has not yet been investigated. In fact, after each disassembly operation, the part must be refurbished before being shipped to the customer. As the refurbishing lead time (RLT) depends on the quality of the disassembled items, it cannot be estimated accurately.

To fill this gap in the literature, the present paper investigates the DLS problem under component RLT uncertainty for the multi-period planning, single EoL item, with a two-echelon disassembly BOM. Here, the demand is considered dynamic and we consider order crossovers. As far as we know, order crossovers have not been investigated in the DLS problem.

In the field of assembly systems, several studies have focused on lead time uncertainty with order crossovers. These works can be split into three groups according to the number of levels in the BOM, the number of periods in the planning horizon, and the aspect of the demand: (i) one-level assembly systems, multi-period planning with a one demand (e.g., [Hnaïen & Afsar, 2017](#); [Jansen, Atan, Adan, & de Kok, 2019; 2018](#)), (ii) one-level assembly systems for a multi-period planning and a constant demand (e.g., [Ould-Louly & Dolgui, 2009; 2013](#); [Shojaie, Bahoosh, & Pourhassan, 2015](#)), and (iii) multi-level assembly systems for a mono-period planning and a one-demand (e.g., [Axsäter, 2006](#); [Ben-Ammar, Castagliola, Dolgui, & Hnaïen, 2020](#); [Ben-Ammar & Dolgui, 2018](#); [Sakiani, Ghomi, & Zandieh, 2012](#)). However, most of these studies are based on analytic models that require problem-specific assumptions, and they do not apply to our case. Interested readers can refer to the literature review proposed by [Díaz-Madroño, Mula, & Peidro \(2014\)](#); [Golmohammadi & Hassini \(2020\)](#).

Finally, as one of the main contributions of the present work is a reformulation of the stochastic program that leads to scenario aggregation, we mention that scenario aggregation techniques exist in stochastic optimization (e.g., [Dupačová, Gröwe-Kuska, & Römis, 2003](#); [Henrion, Küchler, & Römis, 2009](#); [Hewitt, Ortmann, & Rei, 2021](#); [Kuhn, 2008](#)). These approaches seek a good representation of the uncertain parameters with a restricted set of scenarios. More precisely, the problem of optimal scenario

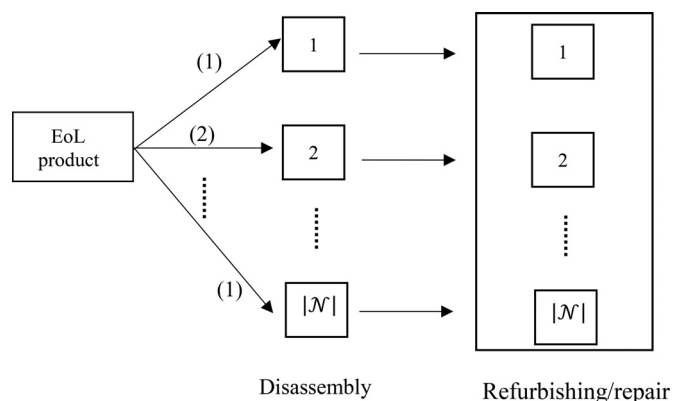


Fig. 1. Two level disassembly system.

reduction determines a scenario subset of prescribed cardinality (and a probability measure based on this set) that is the closest to the initial distribution. For instance, [Hewitt et al. \(2021\)](#) rely on clustering approaches to group scenarios that have a similar impact on the decision making. The resulting approach leads to high-quality upper and lower bounds for stochastic models. [Henrion et al. \(2009\)](#) rely on a discrepancy measure to build the scenarios set. The idea is to build a probability distribution represented by a few scenarios and that deviates at least as possible from the original distribution. [Kuhn \(2008\)](#) aggregate the scenarios based on upper and lower bounds of the optimal value of the original problem. [Dupačová et al. \(2003\)](#) have proposed a scenario reduction approach for a convex stochastic programming problem with a discrete initial probability distribution. However, the existing methods are different from the one proposed in the present work. Here, we rely on the properties of the considered problem to reformulate the inventory constraint and aggregate the scenarios.

3. Problem description and formulation

3.1. Problem description

In the present work, we study the capacitated disassembly lot-sizing problem in a two-level bill of materials. [Fig. 1](#) shows the disassembly and refurbishing process considered in this work. The EoL products (first level) are disassembled into $|\mathcal{N}|$ components (second level), where the number in parentheses represents the disassembly yield (i.e., the number of components obtained by the disassembly operation). Such a BOM structure is common in practice, for instance, in the disassembly of an EoL vehicle ([Pour-Massahian-Tafti, Godichaud, & Amodeo, 2020b](#)). Once the factory has disassembled the End of Life item, each lot of component is sent to a different refurbishing facility, where they are repaired, cleaned, and refurbished. The refurbishing and repair duration varies significantly from a lot of component to another because it depends on the quality of the components. Each component undergoes different refurbishing operations depending on its type and its quality.

In this context, planning the disassembly operation is complex, and it requires to account for the uncertain refurbish and repair time. More precisely, once disassembled, the components i are available after a stochastic refurbishing and repair lead time L_i . That is, if any disassembly operation is performed in period t , the components i are available at period $t + L_i$. For sake of simplicity, we use the term refurbishing lead times to refer to refurbishing and repair lead times in the rest of the paper. As the refurbishing and repair times are specific for each component, the refurbishing lead times are non-identically distributed. In addition, we consider

that the supports of the refurbishing lead time distributions are discrete and finite. More precisely, the refurbishing lead times L_i are random discrete variables characterized by known probability distributions and varying between L_i^- and L_i^+ , $\forall i \in \mathcal{N}$, $L_i^-, L_i^+ \in \mathbb{N}^*$. Here, \mathbb{N}^* is the set of non-negative integers except zero. Such an assumption is consistent with the literature on supply planning under uncertain lead times (see Ben-Ammar, Dolgui, & Wu, 2018, and related works). Note that the disassembly orders can cross in time; i.e., if a lot A is disassembled before lot B , the components resulting from the disassembly of B can arrive before the components resulting from A .

The problem is to decide the disassembly lot sizes Z_t in each period t , and the amount of extra capacity O_t required in the disassembly facility. In most remanufacturing facilities, the disassembly operation is done manually, and it is a bottleneck. Consequently, we consider that the disassembled quantity is limited by a given capacity in each period C_t . The capacity consumption corresponds to the disassembly time F which is known. As the disassembly line is the bottleneck, the disassembly lot-sizing plan must use its capacity correctly. Extra capacity can be added if a cost is paid. A classic solution is to use overtime, but it is also possible to bring in an extra worker for instance by using temporary workers. As in Voß & Woodruff (2006), we assume 'overtime' which encompasses all these capacity additions with a cost penalty u_t . Note that we consider that the disassembly time is known whereas the repair/refurbishing times are uncertain. This assumption is realistic since the refurbishing/repair time depends on the quality of the item. We assume that the time required to disassemble an item is known because the items are disassembled by batch, and the capacity consumption of each individual item averages out over the lot. Our objective is to manage the capacity on the disassembly line, whereas the refurbishing operations are performed in other parts of the shop floor.

The demand D_{it} for each disassembled component i is known in each period t of the planning horizon. The stochastic refurbishing lead times can lead to holding costs h_i and backlog costs b_i whenever the demand of a disassembled component i is not satisfied on time. The objective is to optimize the disassembly quantities at level 1 to minimize the sum of the expected holding and backlog costs of all components in level 2, as well as the setup and overtime costs over a finite horizon. Also, the release of a disassembly order in period t leads to a setup cost s_t in that period.

In this work, we consider the situation where the disassembly plan is decided over a planning horizon of several periods, and it is frozen. This decision process is required because the capacity planning decision (such as the schedule of the employees) cannot be modified in one period for the next one. Frozen horizons are commonly used in the industry to avoid nervousness. The academic literature (e.g., Sridharan & Berry, 1990; Sridharan, Berry, & Udayabhanu, 1987; Sridharan & LaForge, 1994; Xie, Zhao, & Lee, 2003; Zhao & Lee, 1993) shows that freezing the production planning has a notable impact on total cost, the service level of the system, and schedule instability. In addition, freezing the decisions is necessary when they must be made several months in advance. Such decisions include capacity adjustment since workers cannot be hired/fired at the last moment.

As the decisions are frozen, even though the refurbishing lead times are not revealed at once, the problem corresponds to a two-stage decision model. Nevertheless, the approaches proposed in this work can also be used in a dynamic decision framework, where the decisions are updated in each period after observing the quantities refurbished in each period. Solving the dynamic decision framework exactly requires a multi-stage model that would be too complex to solve. In fact, with uncertain refurbishing lead times, the setup decision should be adjustable to make good decisions. With the classical scenario tree representation, such an approach

would lead to an exponential number of binary variables, and the resulting model would not be solvable for realistically sized instances. In this context, the two-stage model provides a heuristic that can be used in a rolling horizon framework and can make the problem more tractable from the computational point of view. Such an approach leads to close to optimal solutions for lot-sizing under uncertain demand (Thevenin, Adulyasak, & Cordeau, 2021).

Before moving on to the core topic of this paper, we briefly investigate the complexity of the problem. The deterministic single item capacitated lot-sizing problem (SCLP) is a special case of CDLS with a single disassembled item i and L_i equals to 0 with probability 1. As the SCLP is NP-hard (Bitran & Yanasse, 1982; Florian, Lenstra, & Rinnooy Kan, 1980), the CDLS is NP-hard too.

3.2. Two-stage stochastic programming formulation

This section gives the two-stage stochastic programming model corresponding to the CDLS problem with stochastic refurbishing lead times. The first-stage decisions correspond to the disassembly plan and the disassembly capacity. These decisions are made before the uncertainty is revealed, and they are frozen for the entire planning horizon. The second-stage decisions (control decisions) correspond to the inventory and backlog level which are taken after the uncertainty is revealed. These second stage decisions are computed once the random refurbishing lead times are revealed.

Before introducing the solution approaches proposed to solve the studied problem, the full list of notations used throughout this paper is given in Table 2.

In this work, we provide a scenario based stochastic optimization formulation. A scenario represents a possible realization of the refurbishing lead time for each component in each period. Classically, in the literature, the set of scenarios contains all possible combination of the L_{it} 's value (e.g., Borodin, Dolgui, Hnaien, & Labadie, 2016), such as $|\Omega| = \prod_{i \in \mathcal{N}} \prod_{t \in \mathcal{T}} (L_i^+ - L_i^- + 1)$. Typically, in an assembly problem, an end-item is assembled when all its components arrive. Consequently, the inventory level of the end-items depends on the lead time of each of its components. Conversely, in a disassembly operation, the refurbishing lead time of a component i only affects the inventory and backlog level of component i . Consequently, the expected inventory level can be computed independently for each end-item i , and thus the scenarios can be generated independently for each component i .

Definition 3.1. Let Ω_i be the set of all possible scenarios for each component i . Each scenario ω corresponds to the realizations from periods 1 to t of the refurbishing lead times of component i . More precisely, if any disassembly operation starts in period t , the component i is available in period $t + L_{it}^\omega$ in scenario ω , and the probability of scenario ω is P_i^ω . If the refurbishing lead times in each period are independent, $P_i^\omega = \prod_{t \in \mathcal{T}} P_i(L_{it} = L_{it}^\omega)$.

Definition 3.2. Let \mathcal{A}_{it}^ω be the set that regroups the indices of quantities that arrive at the least before the end of period t , i.e., $\mathcal{A}_{it}^\omega := \{ \tau \in \mathcal{T} \mid \tau + L_{it}^\omega \leq t \}$.

Such an approach drastically reduces the total number of scenarios to $|\Omega_i| = \prod_{t \in \mathcal{T}} (L_i^+ - L_i^- + 1)$, $\forall i \in \mathcal{N}$.

The CDLS problem with stochastic refurbishing lead times can be expressed with the following two-stage stochastic MILP model.

$$\mathbb{E}(TC) = \min \sum_{t \in \mathcal{T}} \left(\sum_{i \in \mathcal{N}} \sum_{\omega \in \Omega_i} P_i^\omega \left(h_i H_{it}^\omega + b_i B_{it}^\omega \right) + s_t \cdot Y_t + u_t \cdot O_t \right) \quad (1)$$

Table 2
Notation.

Indexes	
t	index of period t , $\forall t \in \mathcal{T}$
i	index of component i , $\forall i \in \mathcal{N}$
ω	index for scenario ω for each component i , $\omega \in \Omega_i$
Parameters	
\mathcal{T}	set of time periods of the planning horizon
\mathcal{N}	set of components
Ω_i	set of possible scenarios for each component i
R_i	disassembly yield of component i
D_{it}	external demand for component i in period t
I_{i0}	starting inventory of component i
L_{it}^ω	refurbishing lead time for component i in period t under scenario ω
h_i	per-period inventory holding cost of one unit of component i
s_t	setup cost in period t
b_i	per-period backlog cost of one unit of component i
F	disassembly operation time
C_t	available capacity in period t
u_t	cost of adding a unit of extra capacity in period t
K	a large number
Functions	
$\mathbb{E}(\cdot)$	expected value
P_i^ω	probability of scenario ω for component i , with $\sum_{\omega \in \Omega_i} P_i^\omega = 1$, $\forall i \in \mathcal{N}$, $\forall \omega \in \Omega_i$
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	
I_{it}^ω	inventory level at the end of period t
H_{it}^ω	positive inventory level of component i at the end of period t under scenario ω . H_{it}^ω is equal to I_{it}^ω if $I_{it}^\omega > 0$ and zero otherwise
B_{it}^ω	negative inventory level of component i at the end of period t under scenario ω . B_{it}^ω is equal to $-I_{it}^\omega$ if $I_{it}^\omega < 0$ and zero otherwise

subject to :

$$H_{it}^\omega - B_{it}^\omega = I_{i0} + \sum_{\tau \in \mathcal{A}_{it}^\omega} R_i Z_\tau - \sum_{\tau=1}^t D_{it} \quad (2)$$

$$i \in \mathcal{N}, t \in \mathcal{T}, \omega \in \Omega_i$$

$$Z_t \leq K \cdot Y_t \quad t \in \mathcal{T} \quad (3)$$

$$F \cdot Z_t \leq C_t + O_t \quad t \in \mathcal{T} \quad (4)$$

$$Z_t, O_t \geq 0 \quad t \in \mathcal{T} \quad (5)$$

$$Y_t \in \{0, 1\} \quad t \in \mathcal{T} \quad (6)$$

$$H_{it}^\omega, B_{it}^\omega \geq 0 \quad i \in \mathcal{N}, t \in \mathcal{T}, \omega \in \Omega_i \quad (7)$$

where

$$K = \max_{i \in \mathcal{N}} \left(\frac{\sum_{i \in \mathcal{N}} \sum_{t \in \mathcal{T}} D_{it}}{R_i} \right) \quad (8)$$

The objective function (1) minimizes the expected value of the total cost over the planning horizon. Constraints (2) define the inventory balance for each component i at the end of period t under scenario ω . Constraints (3) guarantee that a setup cost is generated in period t if any disassembly operation needs to be performed in that period, where the big K is equal to the maximum cumulative demand of all components over the planning horizon (see Eq. (8)). Constraints (4) give the disassembling capacity constraints in each period t . Finally, constraints (5)–(7) provide the domain of the decision variables.

Note that the model (1)–(7) is flexible enough to represent the situation where order crossover is allowed or forbidden. In the case

without order crossover, no part of a component disassembled in period t can be available before the parts of a component disassembled in period t' with $t' < t$.

To better explain the notion of non-crossing orders, we present a small example (with a single component and a planning horizon of 4 time periods). The lead times realisation $L_{1t} = (3, 1, 1, 1)$ correspond to a case with order crossover since the components disassembled in period $t' = 1$ are received after the reception period of the component disassembled in period $t = 2$. On the contrary, if the value of refurbishing lead times L_{1t} are (2,1,1,1), there is no order crossover since the components disassembled in period $t' = 1$ are received at the same time as the component disassembled in period $t = 2$. To account for the situation where order crossover does not occur, the set of scenarios must be generated such the reception period ($t + L_{it}^\omega$) of any order passed in period t is later than the reception period ($t' + L_{it'}^\omega$) of any order passed in a period t' earlier than t .

4. Solution approaches

The stochastic model describes the uncertain refurbishing lead time through all possible scenarios. However, the size of the scenario set increases exponentially with the number of periods, and the resulting MILP requires too much memory and time resources to solve. Consequently, we investigate two techniques to reduce the number of scenarios, namely Monte–Carlo (MC) sampling and the scenario aggregation (SA).

4.1. Monte–Carlo simulation and mixed integer linear programming (MC–MILP)

Model (1)–(7) is difficult to solve because the size $|\Omega_i|$ of the scenario set increases exponentially with the number of periods. Therefore, we introduce the well-known Monte–Carlo (MC) sampling to estimate the expected total cost (Fishman, 1996). MC relies on discrete samples to approximate a probability distribution that cannot be calculated analytically.

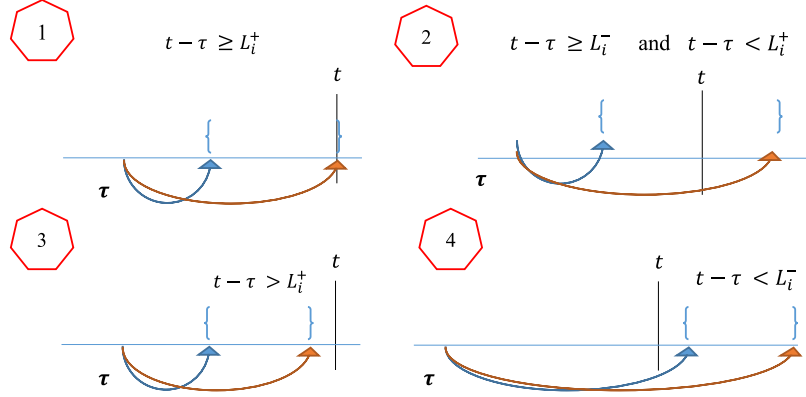


Fig. 2. Aggregation technique.

The estimated total cost noted by \widehat{TC} is obtained by sampling $|\vartheta_i|$ refurbishing lead time scenarios for each component i . To construct a scenario ω of ϑ_i , each refurbishing lead time $L_{it}^\omega \in \{L_{i1}^\omega, \dots, L_{i|\mathcal{T}|}^\omega\}$ is sampled randomly by following the probability distribution of L_{it} . In the resulting sample, all the scenarios have the same probability $\frac{1}{|\vartheta_i|}$. The law of large numbers indicates that \widehat{TC} converges with probability 1 to $\mathbb{E}(TC)$ as ϑ_i increases. That is, for any small value ϵ : $\lim_{\vartheta_i \rightarrow +\infty} P(|\widehat{TC} - \mathbb{E}(TC)| \geq \epsilon) = 0$. Therefore, the model (1)–(7) can be converted to the following MC-MILP model:

$$\widehat{TC} = \min \sum_{\forall t \in \mathcal{T}} \left(\sum_{\forall i \in \mathcal{N}} \sum_{\omega \in \vartheta_i} P_i^\omega \left(h_i H_{it}^\omega + b_i B_{it}^\omega \right) + s_t Y_t + u_t O_t \right) \quad (9)$$

subject to (3)–(6) and:

$$H_{it}^\omega - B_{it}^\omega = I_{i0} + \sum_{\tau \in \mathcal{A}_{it}^\omega} R_i Z_\tau - \sum_{\tau=1}^t D_{i\tau} \quad i \in \mathcal{N}, t \in \mathcal{T}, \omega \in \vartheta_i \quad (10)$$

$$H_{it}^\omega, B_{it}^\omega \geq 0 \quad i \in \mathcal{N}, t \in \mathcal{T}, \omega \in \vartheta_i \quad (11)$$

The objective in (9) is now an estimation of the exact value in (1). Constraints (10) and (11) provide an estimation of the inventory level for each component i at the end of period t based on the random samples.

4.2. Scenario aggregation (SA) approach

This section describes an aggregation strategy that reduces the number of scenarios as much as possible. The SA approach creates a reduced scenario set that keeps the stochastic properties of the original. This approach provides an efficient scenario reduction technique for stochastic programming problems involving both discrete and continuous uncertain parameters, and it can assist researchers to solve the two-level and/or multi-product disassembly/assembly systems under uncertain lead times. The basic idea is to analyze the characteristics of the scenarios, to identify the essential features that affect the decisions, and to remove redundant scenarios.

After releasing the disassembly operation of EoL product in a period τ (with $t > \tau$), each component is received after a random refurbishment lead time $L_{i\tau}$. As the support of the refurbishing lead time distribution is finite, the probability of receiving an order passed before period $t - L_i^+$ is one, and the probability of receiving an order passed after period $t - L_i^-$ is zero. Consequently, the distribution of the backlog and inventory level depends only on the refurbishing lead times of orders passed within a specific time interval. Fig. 2 represents all possible cases of component reception.

In cases 1 and 3, the disassembly occurs in period $\tau | \tau \leq t - L_i^+$, and component i is received before the end of period t . In case 4, the disassembly occurs after period $t - L_i^-$, and the resulting components i are not available in period t . In case 2, as the disassembly occurs in the interval $[t - L_i^+ + 1, t - L_i^-]$, the components can arrive in the time interval $[\tau + L_i^-, \tau + L_i^+]$, and the inventory and backlog levels depend on the realization of lead-times.

Definition 4.1. Let \mathcal{R}_t be the set of disassembly periods that may be involved in the calculation of the backlog or inventory level in period t . A period $\tau \in \mathcal{R}_t$ verifies:

$$\mathcal{R}_t := \{ \tau \in \mathcal{T} \mid t - L_i^+ + 1 \leq \tau \leq t - L_i^-, \forall i \in \mathcal{N} \}.$$

Remark 4.2. The cardinality of \mathcal{R}_t is constant, and it does not depend on the value of $|\mathcal{T}|$, i.e.:

$$|\mathcal{R}_t| = (L_i^+ - L_i^-), \forall i \in \mathcal{N}, t \in \mathcal{T}.$$

Definition 4.3. Let $\alpha_{i\tau t}$ be a stochastic parameter indicating if the quantity of component i resulting from a disassembly in period τ arrives before period t . That is,

$$\alpha_{i\tau t} = \begin{cases} 1 & \text{if } L_{i\tau} \leq t - \tau \\ 0 & \text{if } L_{i\tau} > t - \tau. \end{cases}$$

Proposition 4.4. Eq. (2) are equivalent to the following inventory equations for all components i , periods t , and scenarios ω :

$$H_{it}^\omega - B_{it}^\omega = I_{i0} + \sum_{\tau=1}^{t-L_i^+} R_i Z_\tau + \sum_{\tau=t-L_i^++1}^{t-L_i^-} \alpha_{i\tau t}^\omega R_i Z_\tau - \sum_{\tau=1}^t D_{i\tau}. \quad (12)$$

The parameter $\alpha_{i\tau t}^\omega$ (see Definition 4.3) defines the quantities of EoL to be disassembled at period τ and received before the end of period t under scenario ω . Thus, $\forall i \in \mathcal{N}, \forall t \in \mathcal{T}$ and $\forall \omega \in \Omega_i$, we have:

$$\mathcal{A}_{it}^\omega := \{ \tau \in \mathcal{T} \mid \tau + L_{i\tau}^\omega \leq t \} := \{ \tau \in \mathcal{T} \mid \alpha_{i\tau t}^\omega > 0 \}.$$

The inventory equation shown in constraints (2) can be replaced by:

$$H_{it}^\omega - B_{it}^\omega = I_{i0} + \sum_{\tau=1}^t \alpha_{i\tau t}^\omega R_i Z_\tau - \sum_{\tau=1}^t D_{i\tau} \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, \omega \in \Omega_i. \quad (13)$$

The quantities (Z_τ) disassembled in period $\tau | \tau \leq t - L_i^+$ are refurbished in period t with probability 1 (see cases 1 and 3 in Fig. 2). Thus, for any scenario ω , component i , and period t :

$$\alpha_{i\tau t}^\omega = 1 \quad \forall \tau \leq t - L_i^+. \quad (14)$$

Similarly, the quantities (Z_τ) disassembled in period $\tau | \tau > t - L_i^-$ are not available in period t with probability 1 (see the cases 2 and 4 in Fig. 2). Thus, for any scenario ω , component i , and period t :

$$\alpha_{i\tau t}^\omega = 0 \quad \forall \tau > t - L_i^- \quad (15)$$

Finally, the components of an EoL disassembled in period $\tau | \geq t - L_i^+ + 1$ can be received before or after t (see case 2 in Fig. 2). For these periods, $\alpha_{i\tau t}^\omega$ may take value 1 or 0. More precisely, for any scenario ω , component i , and period t :

$$\alpha_{i\tau t}^\omega \in \{0, 1\} \quad \forall \tau \geq t - L_i^+ + 1. \quad (16)$$

We can easily deduce expression (12) by setting $\alpha_{i\tau t}^\omega$ in Eq. (13) to the values given in (14)–(16).

We explain below how Proposition 4.4 helps to identify the scenario that leads to the same amount of inventory or backlog for a component in a period. Here, the idea is to propose an aggregation strategy in order to reduce the number of possible scenarios by regrouping similar scenarios to form only one. In fact, the inventory and backlog levels of component i in period t are equal for any two scenarios ω_1 and ω_2 such that $\alpha_{i\tau t}^{\omega_1} = \alpha_{i\tau t}^{\omega_2}$, $\forall \tau \in \mathcal{R}_t$. Therefore, for each component i and period t , such scenarios ω_1 and ω_2 can be aggregated.

The resulting set Φ_{it} of aggregated scenarios contains all possible values for the $\alpha_{i\tau t}$'s with $\tau \in \mathcal{R}_t$. Clearly, $\mathbb{P}(\alpha_{i\tau t} = 1) = \mathbb{P}(L_{i\tau} \leq t - \tau)$, and $\mathbb{P}(\alpha_{i\tau t} = 0) = \mathbb{P}(L_{i\tau} > t - \tau)$. Therefore, the occurrence probability P_{it}^ϕ of scenario $\phi \in \Phi_{it}$ corresponds to the joint probability of these refurbishing lead times. In particular, if the refurbishing lead times in each period are independent, the occurrence probability P_{it}^ϕ of scenario $\phi \in \Phi_{it}$ is:

$$P_{it}^\phi = \prod_{\tau \in \mathcal{R}_t} \left(\alpha_{i\tau t}^\phi \times \mathbb{P}(L_{i\tau} \leq t - \tau) + (1 - \alpha_{i\tau t}^\phi) \times \mathbb{P}(L_{i\tau} > t - \tau) \right)$$

When the probability distribution is more complex to express (e.g., the case where order crossover is forbidden), the distribution can be described with the set Ω , and the joint probability distribution can be computed with $P_{it}^\phi = \sum_{\omega \in \Omega^\phi} P^\omega$, where P^ω is the probability of scenario ω , and Ω^ϕ is the set of scenarios in Ω with the correct value for the indicator $\alpha_{i\tau t}$ ($\alpha_{i\tau t}^\omega = \alpha_{i\tau t}^\phi$):

$$\Omega^\phi := \{ \omega \in \Omega \mid L_{i\tau}^\omega \leq t - \tau \quad \forall \tau \text{ with } \alpha_{i\tau t}^\phi = 1 \text{ and } L_{i\tau}^\omega > t - \tau \quad \forall \tau \text{ with } \alpha_{i\tau t}^\phi = 0 \}$$

Definition 4.5. The cardinality of the set of the aggregated scenarios (Φ_{it}) is $|\Phi_{it}| = 2^{|\mathcal{R}_t|}$, $\forall i \in \mathcal{N}$, $t \in \mathcal{T}$. More precisely, Φ_{it} contains all possible values for the vector $[\alpha_{i(t-L_i^++1)t}, \dots, \alpha_{i(t-L_i^-)t}]$.

Proposition 4.6. The cardinality of the set of the aggregated scenarios (Φ_{it}), without considering the crossover of orders:

$$|\Phi_{it}| = L_i^+ - L_i^- + 1 \quad \forall i \in \mathcal{N}, t \in \mathcal{T}$$

Proof. If orders cannot cross, $\alpha_{i\tau-1t} \geq \alpha_{i\tau t}$. That is, if the order passed in period τ arrived in period t ($\alpha_{i\tau t} = 1$), then the order passed in period $\tau - 1$ arrived ($\alpha_{i\tau-1t} = 1$). As a consequence, the vector $[\alpha_{i(t-L_i^++1)t}, \dots, \alpha_{i(t-L_i^-)t}]$ contains only 1 at the beginning followed by only 0:

$$[\alpha_{i(t-L_i^++1)t}, \dots, \alpha_{i(t-L_i^-)t}] = [1, \dots, 1, 0, \dots, 0]. \quad (17)$$

There is only $L_i^+ - L_i^- + 1$ such vectors. \square

Based on the scenario aggregation (SA) technique and Proposition (4.4), a model denoted as SA-MILP is proposed to solve the studied problem with and without orders crossover. Therefore, the

model (1)–(7) is equivalent to the following SA-MILP model:

$$\mathbb{E}(TC) = \min \sum_{t \in \mathcal{T}} \left(\sum_{i \in \mathcal{N}} \sum_{\phi \in \Phi_{it}} P_{it}^\phi \left(h_i H_{it}^\phi + b_i B_{it}^\phi \right) + s_t Y_t + u_t O_t \right) \quad (18)$$

subject to (3)–(6) and:

$$H_{it}^\phi - B_{it}^\phi = \sum_{\tau=1}^{t-L_i^+} R_i Z_\tau + \sum_{\tau \in \mathcal{R}_t} \alpha_{i\tau t}^\phi R_i Z_\tau + I_{i0} - \sum_{\tau=1}^t D_{i\tau} \quad i \in \mathcal{N}, t \in \mathcal{T}, \phi \in \Phi_{it} \quad (19)$$

$$H_{it}^\phi - B_{it}^\phi > 0 \quad i \in \mathcal{N}, t \in \mathcal{T}, \phi \in \Phi_{it} \quad (20)$$

Although the SA approach reduces significantly the number of scenarios, this number remains exponential in the refurbishing lead time range when the orders can cross in time. In the next section, we show that SA-MILP can be coupled with MC sampling to solve large instances. Note that in the case where order crossover is forbidden, the number of aggregated scenarios generated by the SA technique is small enough to directly solve large instances (without the need to sample with MC).

4.3. Monte-Carlo simulation and scenario aggregation (MC-SAMILP)

A straightforward approach to couple the scenario aggregation with MC sampling is to sample a large number of refurbishing lead time scenarios (as presented in Section 4.1), before aggregating these scenarios (as presented in Section 4.2). However, such an approach prevents the control of the number of resulting scenarios, and thus the computation time would be unpredictable. In this section, we propose to generate a scenario set Γ_{it} by sampling directly from the zero-one matrix $(\alpha_{i\tau t})$ and to determine the production plan with the minimal estimated total cost \widehat{TC} . The steps of the MC-SAMILP approach are given below.

- Step 1: Generate random values for $\alpha_{i\tau t}^\gamma$ by following the distribution of $\alpha_{i\tau t}$. More precisely, generate a random value $X_{i\tau t}^\gamma$ following a uniform distribution in $[0; 1]$ for each component i , each period t and τ , and each scenario γ . Then, set $\alpha_{i\tau t}^\gamma$ to 1 or 0 according to the distribution of $\alpha_{i\tau t}$, as follows:

$$\alpha_{i\tau t}^\gamma = \begin{cases} 1 & \text{if } X_{i\tau t}^\gamma \leq \mathbb{P}(\alpha_{i\tau t} = 1) \\ 0 & \text{otherwise} \end{cases}$$

- Step 2: Set the probability of each scenario $\gamma \in \Gamma_{it}$ as follows: $P_i^\gamma = \frac{1}{|\Gamma_{it}|} \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \tau \in \mathcal{R}_t$.
- Step 3: Estimate the expected total cost expressed in (18) using the following MC-SAMILP model:

$$\widehat{TC} = \min \sum_{t \in \mathcal{T}} \left(\sum_{i \in \mathcal{N}} \sum_{\gamma \in \Gamma_{it}} P_{it}^\gamma \left(h_i H_{it}^\gamma + b_i B_{it}^\gamma \right) + s_t Y_t + u_t O_t \right) \quad (21)$$

subject to (3)–(6) and:

$$H_{it}^\gamma - B_{it}^\gamma = \sum_{\tau=1}^{t-L_i^+} R_i Z_\tau + \sum_{\tau \in \mathcal{R}_t} \alpha_{i\tau t}^\gamma R_i Z_\tau + I_{i0} - \sum_{\tau=1}^t D_{i\tau} \quad i \in \mathcal{N}, t \in \mathcal{T}, \gamma \in \Gamma_{it} \quad (22)$$

$$H_{it}^\gamma, B_{it}^\gamma \geq 0 \quad i \in \mathcal{N}, t \in \mathcal{T}, \gamma \in \Gamma_{it} \quad (23)$$

4.4. Sample average approximation (SAA) approach

The sample average approximation algorithm, which is based on the MC simulation technique, is a method that gives a robust solution for a stochastic optimization problem by solving the average sample approximated problems under a large random sample of the stochastic parameters (Kim, Park, & Lee, 2018b; Kleywegt, Shapiro, & Homem-de Mello, 2002).

The stochastic problem can be approximated by a deterministic optimization problem expressed by MC-MILP (respectively MC-SAMILP). However, the costs resulting from the optimization over a sample of scenarios in MC-MILP and MC-SAMILP are biased because the decisions are tailored for these specific scenarios. Therefore, we use the SAA framework developed in Lamiri, Xie, Dolgui, & Grimaud (2008) to compute an unbiased approximation of the expected total costs (see expressions (1) and (18)). The basic steps of the SAA algorithm based on MC simulation are given in Algorithm 1.

Algorithm 1: The SAA algorithm for MC-MILP (respectively MC-SAMILP).

- 1 **Input:** The model MC-MILP (respectively MC-SAMILP) and the related parameter ϑ_i (respectively Γ_{it}).
 - 2 **Step 1:** Solve the MC-MILP (respectively MC-SAMILP) model;
 - 3 **Step 2:** Let X^* and \widehat{TC}^* be the optimal solution and the 'estimated' optimal cost obtained by each model, respectively;
 - 4 **Step 3:** Generate \mathcal{B}_i large set of independent random samples of $L_{it}^1, \dots, L_{it}^{B_i}$ (respectively \mathcal{B}_{it} of $\alpha_{it\tau}^1, \dots, \alpha_{it\tau}^{B_i}$);
 - 5 **Step 4:** Evaluate the exact cost of the optimal solution $\widehat{TC}_{\mathcal{B}}$ for the MC-MILP using (24) (respectively (25), for the MC-SAMILP).
 - 6 **Output:** The approximate cost $\widehat{TC}_{\mathcal{B}}$ for each model.
-

Algorithm 2: Rolling horizon (R-H) sampling method.

- 1 Generate the sets of μ_{it} optimization scenarios with Monte Carlo: $\alpha_{it\tau}^1, \dots, \alpha_{it\tau}^{\mu_{it}}, \forall i \in \mathcal{N}, \forall t \in \mathcal{T}$
 - 2 Create the MC-SAMILP model based on these scenarios
 - 3 Define $t:=1$
 - 4 **while** ($t \leq \mathcal{T}$) **do**
 - 5 Solve MC-SAMILP and implement the resulting disassembly decisions Z_t^o, Y_t^o, O_t^o of period t .
 - 6 Observe the arrival of components sent to repair. That is, the value $\alpha_{it\tau}^o$ unfolds $\forall i \in \mathcal{N}, \tau \leq t$
 - 7 Modify the MC-SAMILP model to set $Z_t = Z_t^o, Y_t = Y_t^o, O_t = O_t^o$, and $\alpha_{it\tau}^\omega = \alpha_{it\tau}^o \forall i \in \mathcal{N}, \tau \leq t, 1 \leq \omega \leq \mu_{it}$
 - 8 Solve MC-SAMILP model
 - 9 $t = t + 1$
 - 10 **end while**
 - 11 Calculate the \widehat{TC} over the planning horizon.
-

Corollary 4.7. The expected total cost expressed in (1) can be approximated by the following SAA model:

$$\widehat{TC}_{\mathcal{B}} = \sum_{t \in \mathcal{T}} \left(\sum_{i \in \mathcal{N}} \sum_{\omega \in \mathcal{B}_i} P_{it}^\omega \left(h_i H_{it}^\omega + b_i B_{it}^\omega \right) + s_t Y_t + u_t O_t \right) \quad (24)$$

subject to (4), (5) and (10), (11), where \mathcal{B}_i is a large set of independent random samples of $L_{it}^1, \dots, L_{it}^{B_i}$ such that $\mathcal{B}_i \subset \Omega_i, \forall i \in \mathcal{N}$.

Corollary 4.8. The expected total cost in Eq. (18) can be approximated by the following SAA model:

$$\widehat{TC}_{\mathcal{B}} = \sum_{t \in \mathcal{T}} \left(\sum_{i \in \mathcal{N}} \sum_{\gamma \in \mathcal{B}_{it}} P_{it}^\gamma \left(h_i H_{it}^\gamma + b_i B_{it}^\gamma \right) + s_t Y_t + u_t O_t \right) \quad (25)$$

subject to (4), (5) and (19), (20), where \mathcal{B}_{it} is a large set of independent random samples of $\alpha_{it\tau}^1, \dots, \alpha_{it\tau}^{B_{it}}, \forall i \in \mathcal{N}$.

We note that the models presented in Corollaries 4.7 and 4.8 do not contain the equations related to the binary indicator of disassembly variables formulated in (3) and (6). Here, we consider an evaluation of the disassembly plans provided by the MC-MILP and MC-SAMILP models.

4.5. Rolling horizon framework

As mentioned earlier, the proposed MILP, MC-MILP, SA-MILP, and MC-SAMILP models are developed to deal with the static decision framework. In the static decision framework, the disassembly decisions are made in period 1, and they are frozen. In this section, we develop a rolling horizon (R-H) strategy to deal with the dynamic decision framework. In the dynamic decision framework, the disassembly decisions are updated in each period t of the horizon after observing the value of the refurbishing lead time up to period t .

We extend the SA-MILP approach to revise the decisions at each period. More precisely, the simulation model (MC-SAMILP) optimizes the disassembly plan in period 0 by considering the first \mathcal{H} period, and the decisions of period 0 are implemented. Then, the value of RLТ in period 0 is revealed, and the inventory levels and on-hand inventory can be updated. The model is updated to account for the initial and on-order inventory, and the disassembly plan is re-optimized on the horizon from period 1 until the last period.

Note that the \widehat{TC} considered in this simulation is not the objective function of the optimization model, but the sum over time of the true cost incurred by the resolution of the MC-SAMILP model over a true scenario. In the next section, all the proposed approaches are evaluated in order to verify their effectiveness.

5. Numerical results

This section presents the results of experiments conducted to study the behavior of the proposed optimization approaches. The proposed models are implemented in C with Concert Technology and solved with IBM CPLEX 12.5 on a PC with processor Intel (R) Core™ i7-5500 CPU @ 2.4GHz and 8 Go RAM under Windows 10 Professional.

5.1. Instances generation

The instance generation methods extends the one used by Kim & Xirouchakis (2010) to account for the specificity of the CDLS with stochastic refurbishing lead times. Table 3 shows the generation method for each parameter, where $D \sim U(a, b)$ indicates that the parameter is randomly generated following a discrete uniform distribution with support $\{a, \dots, b\}$.

The numerical experiments are performed with two test beds. The first contains small and medium size instance to compare the SA approach with the classical stochastic MILP model, and the second contains large instances to show the scalability and the limits of the proposed methods. The *small and medium size test bed* consists of eight sets of instances. Each set contains one randomly-generated problem. Table 4 gives the refurbishing lead time range ($L_i^+ - L_i^-$), the number of components and the number of periods of each instance set. The *large test beds* consists of 50 problems for

Table 3
Characteristics of data sets (Kim & Xirouchakis, 2010).

Parameters	h_i	s_t	l_{i0}	b_i	D_{it}
Values	$D \sim U(5, 10)$	$D \sim U(500, 1000)$	$D \sim U(20, 100)$	$D \sim U(100, 200)$	$D \sim U(50, 200)$
Parameters	R_i	F	C_t	u_t	L_{it}
Values	$D \sim U(1, 4)$	$D \sim U(1, 4)$	$D \sim U(280, 480)$	$D \sim U(20, 40)$	$D \sim U(L_i^-, L_i^+)$

Table 4
Set combinations.

Set	L_{it}	$ \mathcal{N} $	$ \mathcal{T} $
1	$D \sim U(1, 4)$	10	5
2	$D \sim U(1, 4)$	20	5
3	$D \sim U(1, 3)$	10	7
4	$D \sim U(1, 3)$	20	7
5	$D \sim U(6, 10)$	10	15
6	$D \sim U(6, 10)$	20	15
7	$D \sim U(1, 6)$	10	20
8	$D \sim U(1, 6)$	20	20

each combination of 3 levels of the number of components (10, 20 and 30) and 3 levels of the number of periods (10, 20, and 30). For each level of the number of periods, 4 sizes of refurbishing lead time range are considered (1, 6, 9 and $|\mathcal{T}|$).

The size of the random samples ϑ_i and Γ_{it} takes value in [2, 10, 30, 50, 150, 200]. For each ϑ_i and Γ_{it} , we perform 10 independent runs to provide 10 Monte-Carlo optimum solutions X^* and their 'estimated' optimal costs \widehat{TC}^* . The SAA evaluation of X^* uses 10^4 random independent samples. However, to keep the computational time reasonable, the instances with 30 components and 30 periods are evaluated with 10^3 random samples. The time limit is set to one hour, and we report the following performance measures throughout the numerical study:

- I^* : the number of optimal solutions (out of 50 instances) obtained by CPLEX solver in one hour;
- CPU (s): the average CPU times in second;
- G^* : the average integrality gap given by CPLEX;
- G^+ : the average percentage deviation between the estimated total cost (\widehat{TC}) and the expected total cost ($\mathbb{E}(TC)$). More precisely, G^+ is calculated as follows:

$$G^+ = \frac{\widehat{TC} - \mathbb{E}(TC)}{\mathbb{E}(TC)} \times 100 \quad (26)$$

5.2. Effectiveness of the scenario aggregation approach

We start our experiments by showing the good computational performance of the proposed SA-MILP model compared to the MILP model. Table 5 reports the performance of SA-MILP and MILP on the small and medium-size testbed, whether order crossover is considered or not. More precisely, for each instance set, Table 5 reports the number of scenarios, and the CPU times. For some instances, the MILP model is too large to be generated, and it is indicated with a “-” in such cases.

As the number of scenarios in the stochastic MILP is exponential in the number of periods, the MILP consumes too much memory for instances with more than 10 periods. On the contrary, the number of scenarios remains small using our SA technique. As a consequence, the SA method requires significantly less computation time. While MILP requires several minutes to solve instances with less than 10 periods, SA-MILP solves these instances in less than a second. Besides, the special case where orders do not cross in time is even easier to solve with SA-MILP since CPLEX requires less than 0.50 seconds for this special case versus 9.50 seconds for the general case. Note that, for this test bed, the optimality gap G^* returned by CPLEX is null for all instances.

Tables 6 and 7 report the performance of MILP and SA-MILP on the large testbed, when the orders can and cannot cross in time, respectively. As previously stated, the MILP returns an optimal solution only for small-size instances with 10 periods when the refurbishing lead time range does not exceed 1. Table 6 provides further evidence of the effectiveness of the scenario aggregation technique for the case without order crossover since SA-MILP solves all instances in few seconds. For instances with order crossover, Table 7 shows that the performance of SA-MILP depends on the size of the support of the refurbishing lead times probability distribution, the number of periods, and the number of components. Among these parameters, the lead-time range has the most significant impact on the computation time, and once the size of the interval of uncertainty passes a certain value (i.e., $L_i^+ - L_i^- \geq 9$), SA-MILP cannot solve any instance with 30 periods and/or components.

5.3. Effectiveness of scenario aggregation combined with scenario sampling (case with order crossovers)

The numerical results reported in Section 5.2 show that SA-MILP outperforms MILP. Besides, SA-MILP solves large instances in a few seconds when order crossover is forbidden. However, SA-MILP cannot solve instances with large refurbishing lead time ranges when orders can cross. Therefore, the present section investigates the performance of SA-MILP for the case with crossover, when the total costs are approximated with Monte-Carlo Sampling, as presented in Eqs. (1) and (18). Figs. 3–5 illustrate the average convergence of the estimated values towards the exact values under different number of samples. On the one hand, Figs. 3–5 show that for ϑ_i (respectively Γ_{it}) equal to 200, the MC-MILP approach (respectively MC-SAMILP) can generate a good approximation of the stochastic problem for all tests on average. In addition, the results show that the proposed MC-SAMILP model results in a good approximation with less scenarios than the MC-MILP. For example, for problems with 10 periods, the curves in Fig. 3 shows that the MC-SAMILP provides a good approximation with 50 scenarios, whereas MC-MILP requires at least 100 scenarios.

The optimization results of the MC-MILP and MC-SAMILP obtained under 200 random samples are detailed in Table 8. This table indicates that the computation time of the MC-MILP approach decreases with the size of the refurbishing lead time range. Regarding the method combining Monte-Carlo simulation and scenario aggregation approach (MC-SAMILP), the same table clearly shows that this approach can solve all problems within a reasonable time frame.

To investigate further the quality of the approximation obtained with optimization, in Table 9, we analyze the percentage deviations G^+ between the optimal cost obtained with SA-MILP, and the solution obtained with the sampling approach MC-MILP (respectively, MC-SAMILP). For instances with a large number of components, a large number of periods, and a large refurbishing lead times range, the model based on scenario aggregation (SA-MILP) becomes too large to be generated. In such a case, SA-MILP cannot provide a solution, the gap cannot be computed, and this case is indicated by a “*”. The most remarkable observation is that the percentage deviation between MC-MILP and SA-MILP solutions does not exceed 1.76% on average for all tests, while that of MC-SAMILP

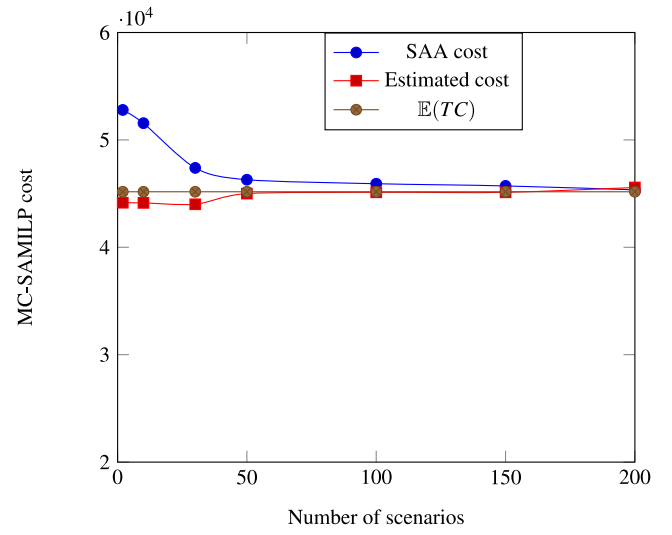
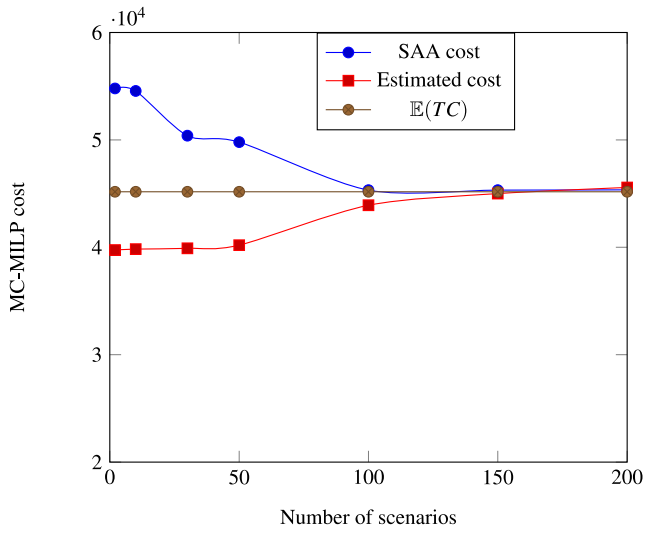


Fig. 3. Convergence of problems with 10 periods on average.

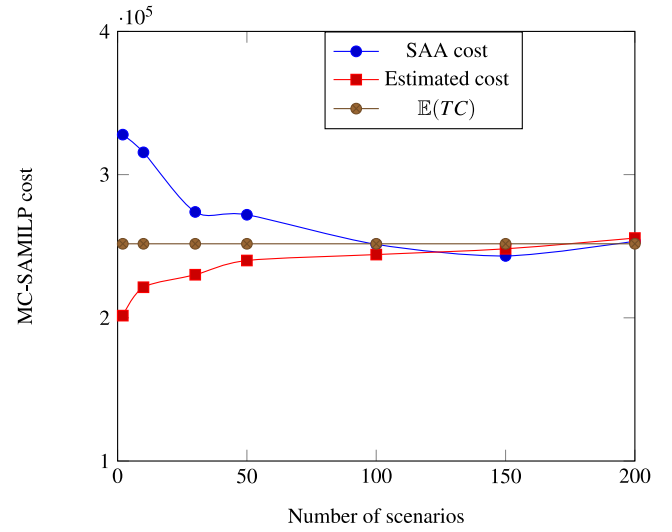
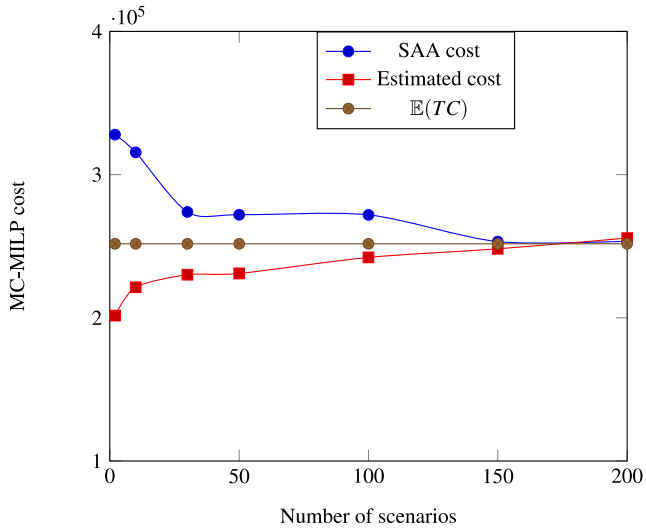


Fig. 4. Convergence of problems with 20 periods on average.

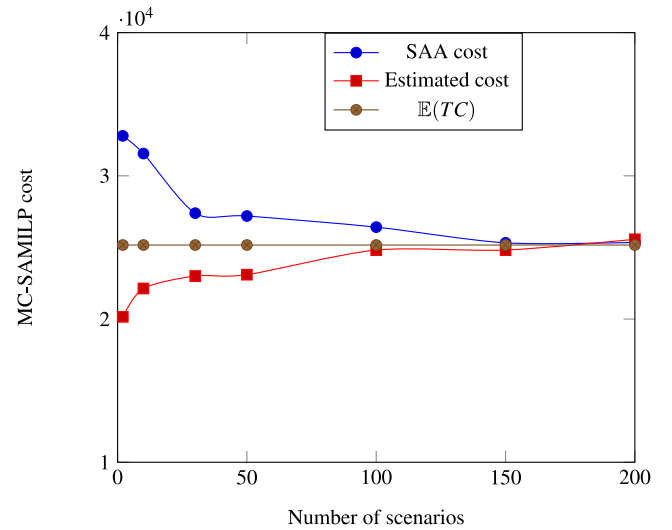
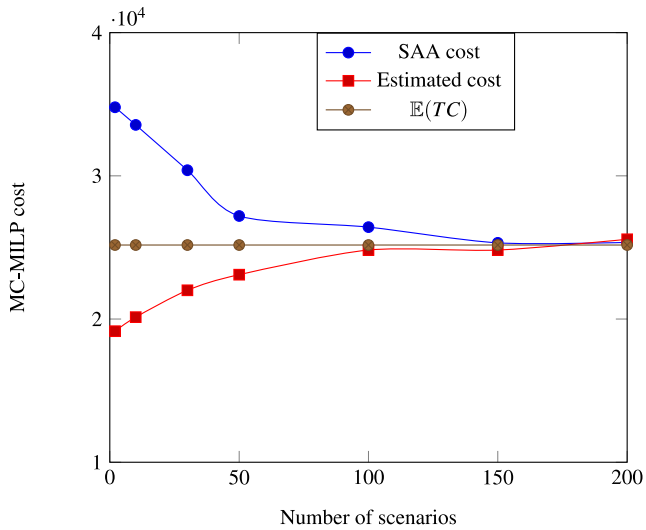


Fig. 5. Convergence of problems with 30 periods on average.

Table 5
SA performance.

L_{it}	$ \mathcal{N} $	$ \mathcal{T} $	Without order crossovers				With order crossovers			
			MILP		SA-MILP		MILP		SA-MILP	
			$ \Omega_i $	CPU(s)	$ \Phi_{it} $	CPU(s)	$ \Omega_i $	CPU(s)	$ \Phi_{it} $	CPU(s)
[1, 4]	10	5	1024	9.17	4	0.04	1,024	30.28	8	0.10
[1, 4]	20	5	1024	22.59	4	0.09	1,024	98.65	8	0.15
[1, 3]	10	7	2,187	699.56	3	0.07	2,187	457.03	4	0.17
[1, 3]	20	7	2187	1,790.62	3	0.10	2,187	2095.56	4	0.24
[6, 10]	10	15	5 ¹⁵	-	5	0.12	5 ¹⁵	-	16	0.35
[6, 10]	20	15	5 ¹⁵	-	5	0.18	5 ¹⁵	-	16	5.28
[1, 6]	10	20	6 ²⁰	-	6	0.21	6 ²⁰	-	32	1.42
[1, 6]	20	20	6 ²⁰	-	6	0.38	6 ²⁰	-	32	9.32

Table 6
Performances of the SA-MILP: without orders crossover.

#components	MILP						SA-MILP					
	10		20		30		10		20		30	
$L_i^+ - L_i^-$	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU
(a) Problem with 10 periods.												
1	50	59.7	50	210.8	50	552.1	50	0.09	50	0.38	50	0.61
6	0	-	0	-	0	-	50	0.14	50	0.83	50	1.28
9	0	-	0	-	0	-	50	0.17	50	0.31	50	1.5
10	0	-	0	-	0	-	50	0.33	50	1.5	50	1.73
Mean	12.5	14.9	12.5	52.7	12.5	138	50	0.2	50	0.8	50	1.3
(b) Problem with 20 periods.												
1	0	-	0	-	0	-	50	0.31	50	0.34	50	0.44
6	0	-	0	-	0	-	50	0.83	50	1.08	50	1.98
9	0	-	0	-	0	-	50	1.19	50	1.63	50	2.63
20	0	-	0	-	0	-	50	2.16	50	5.17	50	6.19
Mean	0	-	0	-	0	-	50	1.12	50	2.1	50	2.9
(c) Problem with 30 periods.												
1	0	-	0	-	0	-	50	0.55	50	1.03	50	1.88
6	0	-	0	-	0	-	50	0.75	50	1.59	50	2.20
9	0	-	0	-	0	-	50	1.23	50	2.97	50	3.42
30	0	-	0	-	0	-	50	6.47	50	13.39	50	29.02
Mean	0	-	0	-	0	-	50	2.25	50	4.74	50	9.13

Table 7
Performances of the SA-MILP: with orders crossover.

#components	MILP						SA-MILP					
	10		20		30		10		20		30	
$L_i^+ - L_i^-$	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU
(a) Problem with 10 periods.												
1	50	510.8	50	829.9	50	2336.5	50	0.3	50	0.07	50	0.4
6	0	-	0	-	0	-	50	2.7	50	8.9	50	35.1
9	0	-	0	-	0	-	50	4.9	50	99.6	50	341.2
10	0	-	0	-	0	-	50	9.5	50	1620.2	0	-
Mean	12.5	127.7	12.5	207.5	12.5	584.2	50	4.3	50	432.2	37.5	94.2
(b) Problem with 20 periods.												
1	0	-	0	-	0	-	50	0.4	50	0.5	50	0.7
6	0	-	0	-	0	-	50	8.4	50	71.3	50	89.3
9	0	-	0	-	0	-	50	622.3	0	-	0	-
20	0	-	0	-	0	-	0	-	0	-	0	-
Mean	0	-	0	-	0	-	37.5	157.7	25	17.9	25	22.5
(c) Problem with 30 periods.												
1	0	-	0	-	0	-	50	1.2	50	1.9	50	2.6
6	0	-	0	-	0	-	50	7.8	50	81.7	50	135.4
9	0	-	0	-	0	-	0	-	0	-	0	-
30	0	-	0	-	0	-	0	-	0	-	0	-
Mean	0	-	0	-	0	-	25	2.5	25	20.9	25	34.5

does not exceed 1.05% on average. This observation confirms that MC-SAMILP provides better approximations than MC-MILP. Note that, the optimality gap G^* returned by CPLEX is null for all instances.

To investigate the computation time and the average optimality deviation obtained by solving the SAA approach for both

MC-MILP and MC-SAMILP, the sampling results are presented in Table 10. Note that the costs for MC-MILP (respectively, MC-SAMILP) are unbiased since they are evaluated through a simulation over $10^4/10^3$ scenarios different from the scenarios used for optimization. Table 10 shows that the SAA can guarantee the convergence of the exact solution with a G^+ and a resolution time that

Table 8
Performances of the MC simulation.

#components	MC-MILP						MC-SAMILP					
	10		20		30		10		20		30	
$L_i^+ - L_i^-$	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU	I^*	CPU
(a) Problem with 10 periods.												
1	50	16.68	50	38.60	50	90.31	50	7.48	50	23.25	50	41.53
6	50	9.89	50	27.20	50	54.39	50	4.23	50	17.73	50	36.15
9	50	6.68	50	25.71	50	51.06	50	4.92	50	17.17	50	64.21
10	50	5.62	50	14.75	50	49.95	50	5.23	50	13.09	50	34.35
Mean	50	9.71	50	26.56	50	61.42	50	5.46	50	17.81	50	44.06
(b) Problem with 20 periods.												
1	50	74.3	50	290.1	50	482.1	50	22.8	50	88.2	50	260.2
6	50	49.9	50	93.4	50	191.9	50	20.7	50	98.2	50	200.3
9	50	31.4	50	98.8	50	210.1	50	22.2	50	75.6	50	190.9
20	50	26.7	50	84.6	50	98.7	50	19.6	50	93.8	50	277.1
Mean	50	45.6	50	141.7	50	245.7	50	21.3	50	88.9	50	232.1
(c) Problem with 30 periods.												
1	50	160.9	50	430.1	50	1757.5	50	50.9	50	261.2	50	632.4
6	50	113.5	50	302.7	50	600.9	50	61.5	50	292.6	50	883.5
9	50	129.4	50	264.5	50	574.2	50	67.6	50	275.7	50	952.4
30	50	84.9	50	204.9	50	647.5	50	59.0	50	255.9	50	1255.3
Mean	50	122.1	50	300.5	50	895	50	194.7	50	271.35	50	931

Table 9
Percentage deviation from MC simulation.

#components	MC-MILP			MC-SAMILP		
	G^+	G^+	G^+	G^+	G^+	G^+
(a) Problem with 10 periods.						
1	2.01	1.08	1.26	0.74	1.65	0.95
6	1.65	0.35	0.75	0.45	0.99	1.32
9	2.33	1.67	1.07	1.06	0.57	0.72
10	1.06	1.87	*	1.35	1.01	*
Mean	1.76	1.24	0.77	0.9	1.05	0.74
(b) Problem with 20 periods.						
1	0.87	1.36	1.94	0.41	1.02	0.81
6	1.96	0.74	1.98	0.09	0.94	1.06
9	0.98	*	*	1.12	*	*
20	*	*	*	*	*	*
Mean	0.95	0.52	0.98	0.4	0.49	0.46
(c) Problem with 30 periods.						
1	0.06	1.33	1.27	0.06	0.98	1.02
6	1.92	1.47	0.65	1.43	0.57	1.18
9	*	*	*	*	*	*
30	*	*	*	*	*	*
Mean	0.49	0.7	0.48	0.37	0.38	0.55

Table 11
RH performance.

L_{it}	$ \mathcal{N} $	$ \mathcal{T} $	SA-MILP		RH		GAP (%)
			$ \Phi_{it} $	$\mathbb{E}(TC)$	$ \Delta_{it} $	\widehat{TC}	
[1, 4]	10	5	8	45,168	200	33789	-25.19
[1, 4]	20	5	8	246,381	200	202794	-17.69
[1, 3]	10	7	4	958,800	200	812578	-15.25
[1, 3]	20	7	4	97,100	200	77630	-20.05
[6, 10]	10	15	16	568,932	200	427853	-24.80
[6, 10]	20	15	16	768,547	200	607469	-20.96
[1, 6]	10	20	32	12,535	200	10368	-17.29
[1, 6]	20	20	32	336,412	200	213576	-31.51

5.4. Effectiveness of R-H method (case with order crossovers)

We continue our experiments by highlighting the good computational performance of the proposed R-H method compared to SA-MILP model. Table 11 reports the performance of these two approaches on the small and medium-size testbed for the case with order crossovers. More precisely, for each instance set, Table 11 reports the number of scenarios, the expected total costs $\mathbb{E}(TC)$, the estimated total costs \widehat{TC} and the percentage deviation between them. Results from Table 11 suggest that the disassembly plan obtained by the R-H strategy is significantly better than the disassembly plan provided by a two-stage stochastic programming model. We note that cost reduction mainly comes from the decrease in the backlog costs obtained when the R-H approach makes dynamic and corrective decisions.

do not exceed 0.5% and 110 seconds on average for the two models, respectively.

From all these analyses, we conclude that the model that combines scenario aggregation and Monte-Carlo simulation (MC-SAMILP) is quite efficient and can provide a good approximation with 200 independent random samples. Therefore, we can state that our study provides additional support for solving the stochastic CDLS problem and validates the usefulness of the SA, MC and SAA approaches.

Table 10
Performances of the SAA on average.

#components	MC-MILP						MC-SAMILP					
	10		20		30		10		20		30	
periods	CPU	G^+	CPU	G^+	CPU	G^+	CPU	G^+	CPU	G^+	CPU	G^+
10	24.1	0.62	48.2	0.17	83.1	0.33	22.3	0.29	53.9	0.12	85.4	0.64
20	53.9	0.43	117.3	0.51	211.7	0.09	69.6	0.35	81.2	0.05	120.3	0.07
30	88.4	0.04	157.3	0.52	27.2	0.74	77.7	0.64	136.3	0.26	22.9	0.17
Mean	55.5	0.36	107.6	0.4	107.3	0.38	56.5	0.42	90.46	0.14	76.3	0.29

Table 12
Characteristics of data sets ($TBO \geq 2$).

Parameters	h_i	s_t	l_{i0}	b_i	D_{it}
Values	$D \sim U(0.3, 0.5)$	$D \sim U(3500, 4500)$	$D \sim U(20, 100)$	$D \sim U(0.6, 1)$	$D \sim U(0, 160)$
Parameters	R_i	F	C_t	u_t	L_{it}
Values	$D \sim U(1, 3)$	$D \sim U(1, 4)$	$D \sim U(280, 480)$	$D \sim U(150, 200)$	$D \sim U(L_i^-, L_i^+)$

Table 13
Performances of the SA-MILP according to TBO.

#components periods	TBO = 1			TBO ≥ 2		
	10 CPU	20 CPU	30 CPU	10 CPU	20 CPU	30 CPU
10	0.15	0.18	0.34	0.10	0.21	0.26
20	0.54	0.76	1.75	0.46	0.578	1.56
30	22.0	25.65	42.69	5.17	10.79	23.89
Mean	7.57	8.87	14.93	1.92	3.86	8.57

5.5. Impact of the time between order

The time Between Order (TBO) plays an important role on the computational complexity of lot-sizing problems. In Section 5.1, numerical results are presented based on the benchmarks of Kim & Xirouchakis (2010) that contains instances with a TBO equal to 1 (an order at each period). In this section, these benchmarks are modified to obtain a TBO greater than or equal to 2. We modify the ratio related to setup cost, inventory holding costs and backlog costs. Eq. (27) presents the TBO formula adapted from Godichaud & Amodeo (2018, 2019); Pour-Massahian-Tafti et al. (2020b). We define \mathcal{N} as the set of components and $\bar{s}, \bar{R}, \bar{h}, \bar{b}, \bar{D}$ as the mean of the discrete uniform distribution used to randomly generate setup, yield, holding inventory, backlog and demand, respectively. In addition, the instance of Kim & Xirouchakis (2010) consider rather small values for u_t (cost of outsourcing) and rather large values for the backlog cost b_i . In this section, we investigate the impact of instances with values more in line with the lot-sizing literature.

The parameters given in Table 12 are generated in such a way that $TBO \geq 2$. The time needed to obtain optimal solutions presented in Table 13 is significantly reduced. This is explained by the fact that increasing TBO reduces the number of orders whom the number of scenario depends.

$$\sqrt{\frac{2\bar{s}}{\mathcal{N} \cdot \bar{R} \cdot \bar{h} \cdot \bar{D}}} \leq TBO \leq \sqrt{\frac{2\bar{s}}{\mathcal{N} \cdot \bar{R} \cdot \bar{h} \cdot \bar{D}}} \sqrt{\frac{\bar{h} + \bar{b}_i}{\bar{b}_i}} \quad (27)$$

5.6. Benefit of stochastic modelling

In this subsection, we seek to assess the practical performance of the stochastic programming model (SA-MILP) by comparing it with three deterministic models which completely ignores uncertainty. These deterministic versions consider the minimum, average, and maximum lead time, respectively. More precisely, in the

first version, the value of L_i corresponds to L_i^- ($L_i = L_i^-$). In the second version, the value of L_i is the expected refurbishing lead time ($L_i = \mathbb{E}(L_i)$). Finally, in the most conservative model, the value L_i is set to the maximum lead time L_i^+ ($L_i = L_i^+$).

The parameters given in Table 12 and the MILP model (1–8) are used to evaluate the “exact” costs of the solutions obtained with this deterministic model. Results from Table 14 show that the solution provided by the stochastic optimization approach significantly outperforms the deterministic methods, since the stochastic approach reduces the costs by up to 31.14%

6. Conclusion

In this paper we have proposed a stochastic model for the capacitated disassembly lot sizing problem (CDLS) that can allow the orders to cross in time. The time necessary to refurbish the components is considered as a stochastic variable under the interval representation of uncertainty. The planning problem identifies how much end-of-life (EoL) product to disassemble during each period in order to minimize the expected total cost. Our contribution can be summarized as follows: first, a formulation of the stochastic CDLS problem by a two-phase stochastic mathematical program is proposed. The first-stage decisions correspond to the quantity of EoL product that has to be disassembled as well as the extra capacity to be added in each period. The second-stage decisions correspond to the inventory and the backlog levels. This model describes the random parameter through the set of all possible scenarios. Second, due to the large number of scenarios, a scenario aggregation (reduction) technique is developed. The effectiveness of the SA approach has been successfully tested and proved, especially for the case when order crossover is forbidden. Third, an approach combining a Monte-Carlo simulation and mixed integer programming (respectively scenario aggregation) is proposed in order to study the convergence of the expected cost. Fourth, an almost optimal solution is obtained from a SAA based on the Monte-Carlo optimization method. Fifth, to deal with the dynamic decision frameworks, we introduce a rolling horizon approach.

Our investigations into this area are still ongoing. First, most studies treat the problem under uncertainty on two level DLS problems. Since in real life, most DLS are multi-levels, a promising future research area could be solving the multi-level CDLS problem. Second, studying the uncertainty of demand and/or yield is also an important consideration. Finally, most existing research focuses only on developing effective and efficient approaches for solving stochastic models. However, it is important to evaluate the

Table 14
Performances of the stochastic approach.

#components periods	$L_i = L_i^-$			$L_i = \mathbb{E}(L_i)$			$L_i = L_i^+$		
	10 G (%)	20 G (%)	30 G (%)	10 G (%)	20 G (%)	30 G (%)	10 G (%)	20 G (%)	30 G (%)
10	30.06	20.23	28.61	43.63	37.51	24.54	15.16	15.93	19.41
20	31.37	14.69	27.56	34.05	32.06	35.69	4.92	8.35	9.37
30	32.01	18.68	21.83	43.93	24.08	27.61	8.44	3.45	1.16
Mean	31.14	17.86	26.01	40.54	31.22	29.28	9.51	9.24	10.98

G: Percentage deviation from the stochastic approach.

effectiveness and the robustness of the solution methods. Performance indicators for evaluation should be introduced in future research.

References

- Axsäter, S. (2006). *Planning order releases for an assembly system with random operation times*. In G. Liberopoulos, C. T. Papadopoulos, B. Tan, J. M. Smith, & S. B. Gershwin (Eds.) (pp. 333–344). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Barba-Gutiérrez, Y., & Adenso-Díaz, B. (2009). Reverse MRP under uncertain and imprecise demand. *The International Journal of Advanced Manufacturing Technology*, 40(3–4), 413–424.
- Ben-Ammar, O., Castagliola, P., Dolgui, A., & Hnaien, F. (2020). A hybrid genetic algorithm for a multilevel assembly replenishment planning problem with stochastic lead times. *Computers and Industrial Engineering*, 149, 106794.
- Ben-Ammar, O., & Dolgui, A. (2018). Optimal order release dates for two-level assembly systems with stochastic lead times at each level. *International Journal of Production Research*, 56(12), 4226–4242.
- 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(10), 1174–1186.
- Borodin, V., Dolgui, A., Hnaien, F., & Labadie, N. (2016). Component replenishment planning for a single-level assembly system under random lead times: A chance constrained programming approach. *International Journal of Production Economics*, 181, 79–86.
- Dupačová, J., Gröwe-Kuska, N., & Römisich, W. (2003). Scenario reduction in stochastic programming. *Mathematical Programming*, 95(3), 493–511.
- Díaz-Madroño, M., Mula, J., & Peidro, D. (2014). A review of discrete-time optimization models for tactical production planning. *International Journal of Production Research*, 52(17), 5171–5205.
- 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(4), 851–878.
- Fishman, G. (1996). *Monte Carlo: Concepts, algorithms, and applications*. Springer.
- Florian, M., Lenstra, J. K., & Rinnooy Kan, A. (1980). Deterministic production planning: Algorithms and complexity. *Management Science*, 26(7), 669–679.
- 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(18), 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.
- Golmohammadi, A., & Hassini, E. (2020). Review of supplier diversification and pricing strategies under random supply and demand. *International Journal of Production Research*, 58(11), 3455–3487.
- Guide, V. D. R. (2000). Production planning and control for remanufacturing: industry practice and research needs. *Journal of Operations Management*, 18(4), 467–483.
- Guide, V. D. R., & Srivastava, R. (1997). An evaluation of order release strategies in a remanufacturing environment. *Computers and Operations Research*, 24(1), 37–47.
- Gungor, A., & Gupta, S. M. (1999). Issues in environmentally conscious manufacturing and product recovery: A survey. *Computers and Industrial Engineering*, 36(4), 811–853.
- Gupta, S. M., & Lambert, A. F. (2016). *Disassembly modeling for assembly, maintenance, reuse and recycling*. CRC Press.
- Gupta, S. M., & Taleb, K. N. (1994). Scheduling disassembly. *The International Journal of Production Research*, 32(8), 1857–1866.
- Gutwiski, T. G., Sahni, S., & Boustani, A. (2011). Remanufacturing and energy savings. *Environmental Science and Technology*, 45(10), 4540–4547.
- Henrion, R., Küchler, C., & Römisich, W. (2009). Scenario reduction in stochastic programming with respect to discrepancy distances. *Computational Optimization and Applications*, 43(1), 67–93.
- Hewitt, M., Ortmann, J., & Rei, W. (2021). Decision-based scenario clustering for decision-making under uncertainty. *Annals of Operations Research*, 1–25.
- Hnaien, F., & Afsar, H.-M. (2017). Robust single-item lot-sizing problems with discrete-scenario lead time. *International Journal of Production Economics*, 185, 223–229.
- Inderfurth, K., & Langella, I. M. (2006). Heuristics for solving disassemble-to-order problems with stochastic yields. *OR Spectrum*, 28(1), 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.
- Jaehn, F. (2016). Sustainable operations. *European Journal of Operational Research*, 253(2), 243–264.
- Jansen, S., Atan, Z., Adan, I., & de Kok, T. (2019). Setting optimal planned lead-times in configure-to-order assembly systems. *European Journal of Operational Research*, 273(2), 585–595.
- Jansen, S. W. F., Atan, Z., Adan, I. J. B. F., & de Kok, A. G. (2018). Newsvendor equations for production networks. *Operations Research Letters*, 46(6), 599–604.
- 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(4), 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(9–12), 1463–1473.
- Kim, D.-H., Doh, H.-H., & Lee, D.-H. (2018a). Multi-period disassembly leveling and lot-sizing for multiple product types with parts commonality. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 232(5), 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(10), 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(1), 195–212.
- Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2007). Disassembly scheduling: Literature review and future research directions. *International Journal of Production Research*, 45(18–19), 4465–4484.
- Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2008). An exact algorithm for two-level disassembly scheduling. *Journal of the Korean Institute of Industrial Engineers*, 34(4), 414–424.
- Kim, H.-J., Lee, D.-H., Xirouchakis, P., & Kwon, O. K. (2009). A branch and bound algorithm for disassembly scheduling with assembly product structure. *Journal of the Operational Research Society*, 60(3), 419–430.
- Kim, H.-J., & Xirouchakis, P. (2010). Capacitated disassembly scheduling with random demand. *International Journal of Production Research*, 48(23), 7177–7194.
- Kim, H.-W., Park, C., & Lee, D.-H. (2018b). Selective disassembly sequencing with random operation times in parallel disassembly environment. *International Journal of Production Research*, 56(24), 7243–7257.
- Kim, J.-G., Jeon, H.-B., Kim, H.-J., Lee, D.-H., & Xirouchakis, P. (2005). Capacitated disassembly scheduling: Minimizing the number of products disassembled. In *International conference on computational science and its applications* (pp. 538–547). Springer.
- Kleywegt, A. J., Shapiro, A., & Homem-de Mello, T. (2002). The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12(2), 479–502.
- Kuhn, D. (2008). Aggregation and discretization in multistage stochastic programming. *Mathematical Programming*, 113(1), 61–94.
- Lamiri, M., Xie, X., Dolgui, A., & Grimaud, F. (2008). A stochastic model for operating room planning with elective and emergency demand for surgery. *European Journal of Operational Research*, 185(3), 1026–1037.
- Langella, I. M. (2007). Heuristics for demand-driven disassembly planning. *Computers and Operations Research*, 34(2), 552–577.
- Lee, D.-H., & Xirouchakis, P. (2004). A two-stage heuristic for disassembly scheduling with assembly product structure. *Journal of the Operational Research Society*, 55(3), 287–297.
- Lee, D.-H., Xirouchakis, P., & Züst, R. (2002). Disassembly scheduling with capacity constraints. *CIRP Annals*, 51(1), 387–390.
- Liu, K., & Zhang, Z.-H. (2018). Capacitated disassembly scheduling under stochastic yield and demand. *European Journal of Operational Research*, 269(1), 244–257.
- Neuendorf, K.-P., Lee, D.-H., Kiritsis, D., & Xirouchakis, P. (2001). Disassembly scheduling with parts commonality using petri nets with timestamps. *Fundamenta Informaticae*, 47(3–4), 295–306.
- Ould-Louly, M.-A., & Dolgui, A. (2009). Calculating safety stocks for assembly systems with random component procurement lead times: A branch and bound algorithm. *European Journal of Operational Research*, 199(3), 723–731.
- Ould-Louly, M.-A., & Dolgui, A. (2013). Optimal MRP parameters for a single item inventory with random replenishment lead time, POQ policy and service level constraint. *International Journal of Production Economics*, 143(1), 35–40.
- Pearce John, A. (2009). The profit-making allure of product reconstruction-MIT SMR store. *MIT Sloan Management Review*, 50, 59–63.
- Piewthongam, K., Chatavithae, P., & Apichottanakul, A. (2019). Disassembly scheduling for the meat processing industry with product perishability. *Journal of Advanced Manufacturing Systems*, 18(03), 447–467.
- Pini, M., Neri, P., & Ferrari, A. M. (2018). Environmental performance of waste management in an Italian region: How LCI modelling framework could influence the results. *Procedia CIRP*, 69, 956–961.
- Pour-Massahian-Tafti, M., Godichaud, M., & Amodeo, L. (2019a). Models for disassembly lot sizing problem with decisions on surplus inventory. In *Advances in optimization and decision science for society, services and enterprises* (pp. 423–432). Springer.
- Pour-Massahian-Tafti, M., Godichaud, M., & Amodeo, L. (2019b). Models for the single product disassembly lot sizing problem with disposal. *IFAC-PapersOnLine*, 52(13), 547–552.
- Pour-Massahian-Tafti, M., Godichaud, M., & Amodeo, L. (2019c). Single product disassembly lot sizing problem with disposal. In *2019 IEEE 6th international conference on industrial engineering and applications (ICIEA)* (pp. 135–140). IEEE.
- Pour-Massahian-Tafti, M., Godichaud, M., & Amodeo, L. (2020a). Disassembly EOQ models with price-sensitive demands. *Applied Mathematical Modelling*, 88, 810–826.
- 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*, 59(22), 1–21.

- Prakash, P., Ceglarek, D., & Tiwari, M. K. (2012). Constraint-based simulated annealing (CBSA) approach to solve the disassembly scheduling problem. *The International Journal of Advanced Manufacturing Technology*, 60(9–12), 1125–1137.
- Quezada, F., Gicquel, C., Kedad-Sidhoum, S., & Vu, D. Q. (2020). A multi-stage stochastic integer programming approach for a multi-echelon lot-sizing problem with returns and lost sales. *Computers and Operations Research*, 116, 104865.
- Sakiani, R., Ghomi, S. M. T. F., & Zandieh, M. (2012). Multi-objective supply planning for two-level assembly systems with stochastic lead times. *Computers and Operations Research*, 39(7), 1325–1332.
- Shojaie, S. H., Bahoosh, A., & Pourhassan, M. (2015). A study on MRP with using leads time, order quality and service level over a single inventory. *Journal UMP Social Sciences and Technology Management*, 3(1)(1), 235–239.
- Slama, I., Ben-Ammar, O., & Dolgui, A. (2021a). A model for a multi-level disassembly system under random disassembly lead times. In *Iifip international conference on advances in production management systems* (pp. 39–47). Springer.
- Slama, I., Ben-Ammar, O., Dolgui, A., & Masmoudi, F. (2020a). 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., Dolgui, A., & Masmoudi, F. (2020b). A newsboy formulae to optimize planned lead times for two-level disassembly systems. *IFAC-PapersOnLine*, 53(2), 10816–10821.
- Slama, I., Ben-Ammar, O., Dolgui, A., & Masmoudi, F. (2020c). A stochastic model for a two-level disassembly lot-sizing problem under random lead time. In *Iifip international conference on advances in production management systems* (pp. 275–283). Springer.
- Slama, I., Ben-Ammar, O., Dolgui, A., & Masmoudi, F. (2021b). Genetic algorithm and Monte Carlo simulation for a stochastic capacitated disassembly lot-sizing problem under random lead times. *Computers and Industrial Engineering*, 159, 107468.
- Slama, I., Ben-Ammar, O., Masmoudi, F., & Dolgui, A. (2019a). Disassembly scheduling problem: Literature review and future research directions. *IFAC-PapersOnLine*, 52(13), 601–606.
- Slama, I., Ben-Ammar, O., Masmoudi, F., & Dolgui, A. (2019b). Scenario-based stochastic linear programming model for multi-period disassembly lot-sizing problems under random lead time. *IFAC-PapersOnLine*, 52(13), 595–600.
- Sridharan, V., & Berry, W. L. (1990). Freezing the master production schedule under demand uncertainty. *Decision Sciences*, 21(1), 97–120.
- Sridharan, V., Berry, W. L., & Udayabhanu, V. (1987). Freezing the master production schedule under rolling planning horizons. *Management Science*, 33(9), 1137–1149.
- Sridharan, V., & LaForge, R. L. (1994). Freezing the master production schedule: Implications for fill rate. *Decision Sciences*, 25(3), 461–469.
- Suzanne, E., Absi, N., & Borodin, V. (2020). Towards circular economy in production planning: Challenges and opportunities. *European Journal of Operational Research*, 287(1), 168–190.
- Taleb, K. N., Gupta, S. M., & Brennan, L. (1997). Disassembly of complex product structures with parts and materials commonality. *Production Planning and Control*, 8(3), 255–269.
- Tarim, S. A., Dogru, M. K., Özen, U., & Rossi, R. (2011). An efficient computational method for a stochastic dynamic lot-sizing problem under service-level constraints. *European Journal of Operational Research*, 215(3), 563–571.
- Tarim, S. A., & Smith, B. M. (2008). Constraint programming for computing non-stationary (R,S) inventory policies. *European Journal of Operational Research*, 189(3), 1004–1021.
- Thevenin, S., Adulyasak, Y., & Cordeau, J.-F. (2021). Material requirements planning under demand uncertainty using stochastic optimization. *Production and Operations Management*, 30(2), 475–493.
- Tian, X., & Zhang, Z.-H. (2019). Capacitated disassembly scheduling and pricing of returned products with price-dependent yield. *Omega*, 84, 160–174.
- Ullerich, C., & Buscher, U. (2013). Flexible disassembly planning considering product conditions. *International Journal of Production Research*, 51(20), 6209–6228.
- Vargas, V. (2009). An optimal solution for the stochastic version of the Wagner-Whitin dynamic lot-size model. *European Journal of Operational Research*, 198(2), 447–451.
- Voß, S., & Woodruff, D. L. (2006). *Introduction to computational optimization models for production planning in a supply chain*: vol. 240. Springer Science & Business Media.
- Wang, H.-F., & Huang, Y.-S. (2013). A two-stage robust programming approach to demand-driven disassembly planning for a closed-loop supply chain system. *International Journal of Production Research*, 51(8), 2414–2432.
- Xie, J., Zhao, X., & Lee, T. S. (2003). Freezing the master production schedule under single resource constraint and demand uncertainty. *International Journal of Production Economics*, 83(1), 65–84.
- Zhao, X., & Lee, T. S. (1993). Freezing the master production schedule for material requirements planning systems under demand uncertainty. *Journal of operations management*, 11(2), 185–205.