

# Approches d'optimisation pour un problème de planification de désassemblage sous incertitude des délais de désassemblage

## Optimization approaches for a stochastic disassembly lot-sizing problem under random lead times

Ilhem Slama<sup>1</sup>, Oussama Ben-Ammar<sup>2</sup>, Alexandre Dolgui<sup>1</sup>, Faouzi Masmoudi<sup>3</sup>

<sup>1</sup>IMT Atlantique, LS2N-CNRS, La Chantrerie, 4 rue Alfred Kastler - B.P. 20722, 44307 Nantes, France  
ilhem.slama@imt-atlantique.fr

<sup>2</sup>Mines Saint-Étienne, Univ Clermont Auvergne, UMR-CNRS 6158 LIMOS, CMP Department of Manufacturing Sciences and Logistics, 880 route de Mimet, F-13541 Gardanne, France

<sup>3</sup>Engineering School of Sfax, Laboratory of Mechanic, Modeling and Production (LA2MP), University of Sfax, Tunisia

**RÉSUMÉ.** Dans un contexte de logistique inverse, cette étude traite un problème de planification de désassemblage multi-période sous incertitude de délai d'obtention des composants. Ce délai est le nombre de périodes écoulées entre la passation d'un ordre de désassemblage et la réception des composants désassemblés. À notre connaissance, cette incertitude est étudiée pour la première fois dans le problème de planification des opérations de désassemblage. Un système de désassemblage avec une nomenclature à deux-niveau et un seul type de produit est considéré. Les délais de désassemblage sont des variables aléatoires discrètes avec un nombre fini de valeurs possibles. Le problème est formulé en tant que modèle de Programmation Linéaire en Nombres Entiers (PLNE) stochastique à deux étapes à travers tous les scénarios possibles afin de minimiser l'espérance mathématique du coût total. En raison du grand nombre de scénarios, le modèle PLNE est intraitable. Pour le rendre traitable, nous proposons une approche d'optimisation qui combine la simulation Monte Carlo (MC) et le modèle PLNE. Cependant, cette approche est inefficace pour les problèmes de grande taille. Pour résoudre des problèmes à grande échelle, nous proposons un algorithme génétique classique. Les résultats expérimentaux basés sur des données générées de façon aléatoire montrent l'efficacité de l'algorithme proposé.

**ABSTRACT.** In a reverse logistics context, this study addresses a multi-period disassembly lot-sizing problem under uncertainty disassembly lead times. The disassembly lead time comprises the time elapsed between releasing the disassembly order and receiving the disassembled item. To the best of our knowledge, this uncertainty is studied for the first time in the disassembly lot-sizing problem. A disassembly system with a two-level BOM and a single end-of-life product type is considered. Disassembly lead times are discrete random variables with a finite number of possible values. The problem is formulated as a two-stage stochastic Integer Linear Programming (ILP) model through all possible scenarios in order to minimize the expected total cost. Because of the large number of scenarios, ILP is intractable. To make it tractable, we propose an optimization approach that combines Monte Carlo (MC) simulation and ILP. Besides, to solve large scale problems, we propose a basic genetic algorithm. Experimental results based on randomly generated data show the effectiveness of the proposed.

**MOTS-CLÉS.** Planification de désassemblage, délai stochastique, programmation stochastique, loi de grands nombres, simulation Monte Carlo.

**KEYWORDS.** Disassembly lot-sizing, stochastic lead time, stochastic programming, law of Large Numbers, Monte Carlo simulation.

## 1. Introduction

Avec l'accumulation de la pression environnementale sur les activités industrielles, le flux inverse est devenu indispensable. Le processus de reconditionnement est une étape cruciale dans l'économie circulaire, et il concerne toutes les tâches associées à la collecte, désassemblage, la remise à neuf, la réparation, le recyclage et le rejet des Produits en Fin de Vie (PFV) (Kim *et al.*, 2018). Après avoir reçu les PFV, chaque produit est envoyé à sa cellule de désassemblage où l'opération de désassemblage est

effectuée. Les composants désassemblés doivent être nettoyés, remis à neuf afin d'atteindre le niveau de qualité souhaité.

Dans ce document, nous examinons la situation dans laquelle la demande en composants doit être satisfaite par le désassemblage de PFV. Selon Ilgin et Gupta (2010), le problème étudié entre dans la classification du problème de Lot-Sizing de Désassemblage (LSD) qui considère la meilleure utilisation des ressources de désassemblage afin de satisfaire les demandes en composants sur un horizon de planification donné.

De nombreuses sources d'incertitudes liées à la demande, les temps opératoires, la quantité et la qualité des composants désassemblés peuvent perturber le processus de désassemblage. Ce processus implique également un degré d'incertitude sur les délais de désassemblage. Par définition, ce délai est une fenêtre de temps entre le lancement d'un ordre de désassemblage et la réception des composants démontés (Kim *et al.*, 2006a). En général, ces délais sont exprimés en nombre des unités de temps (jours, mois, semaines, etc.). Par conséquent, ces incertitudes troublent le niveau de stock et peut créer de la rupture. Guide Jr (2000) a indiqué que le temps moyen de désassemblage et de remise à neuf d'un produit recyclé typique peut varier entre 5,54 et 300 heures avec des coefficients de variation pouvant atteindre 5 heures. Dans ce cas, il est difficile de définir les délais nécessaires pour effectuer le processus de désassemblage ou d'obtention des composants. Nous notons aussi que pour la majorité des industriels, un effort considérable est nécessaire pour paramétrer les délais planifiés afin de satisfaire la demande des clients avec les moindre coûts. A notre connaissance, il n'existe pas des travaux dans la littérature qui aident les décideurs à déterminer les dates et les quantités optimales de désassemblage sous incertitude des délais de désassemblage.

L'objectif de cette recherche est de construire un modèle de programmation stochastique à deux étapes pour un problème LSD multi-période, une nomenclature à deux-niveau avec une capacité de désassemblage limitée. Pour représenter l'incertitude de délais de désassemblage, nous utilisons une formulation d'optimisation stochastique basée sur l'ensemble de tous les scénarios possibles. Nous notons que dans cette étude, les délais de désassemblage sont des variables aléatoires discrètes qui suivent une distribution de probabilité connue et limitée.

Dans notre processus de décision en deux étapes, les délais de désassemblage pour l'ensemble de l'horizon de planification sont révélés une fois que les décisions de la première étape sont prises. Ces décisions correspondent aux décisions de désassemblage, de reconfiguration et de temps supplémentaire de désassemblage à chaque période. La deuxième étape concerne les décisions relatives aux niveaux des stocks.

Une simulation Monte Carlo (MC) basée sur un nombre important d'échantillons générés aléatoirement est suggérée pour résoudre les instances de taille moyenne. Cependant, ces échantillons peuvent entraîner une intractabilité ou un temps de calcul élevé. Pour cette raison, nous proposons une méthode approchée basée sur un algorithme génétique (AG). L'objectif est de résoudre des problèmes plus grands garantissant un bon compromis entre la qualité des solutions obtenues et le temps de calcul.

Le reste de ce papier est organisé comme suit. La section suivante présente un bref aperçu de l'analyse de la littérature sous incertitude. La section 3. présente la description et la formulation du problème LSD. La section 4. propose un algorithme génétique pour résoudre le modèle que nous proposons. La section 5. présente les résultats numériques. La section 6. présente la conclusion avec des pistes pour de futures

études.

## 2. État de l'art

Les problèmes LSD sont formulés en modèles déterministes ou stochastiques en intégrant des facteurs aléatoires. Dans la littérature scientifique, la plupart des travaux se focalisent sur le développement de méthodes efficaces et efficientes pour résoudre des problèmes déterministes. Ceci est expliqué par le fait que l'incertitude est complexe. Les lecteurs intéressés peuvent consulter ces études Kim *et al.* (2007); Slama *et al.* (2019).

Une grande partie de la littérature concerne le problème déterministe (voir par exemple : Guide Jr et Srivastava (1997); Gupta et Taleb (1994); Taleb *et al.* (1997); Neuendorf *et al.* (2001); Lee *et al.* (2004); Kim *et al.* (2006b); Langella (2007); Kang *et al.* (2016); Prakash *et al.* (2012); Kim *et al.* (2018); Godichaud et Amodeo (2018, 2019) et très peu d'études portent sur la version avec une capacité sur la quantité de produits à désassembler (voir par exemple : Kim *et al.* (2006a); Ullrich et Buscher (2013); Ji *et al.* (2016); Slama *et al.* (2020); Lee *et al.* (2002); Kim *et al.* (2003).

En pratique, le processus de récupération des produits est sensible à diverses sources d'incertitude, telles que la demande incertaine des clients en composants, les taux de récupération, etc. (Guide Jr et Srivastava, 1997). Ces incertitudes créent des perturbations sur le plan de désassemblage qui entraîne par la suite une demande non satisfaite des clients et des difficultés à gérer le stock (Brach et Brusset, 2014). La littérature scientifique sur le problème de LSD s'est concentré principalement sur l'incertitude des demandes et/ou du rendement du désassemblage (Suzanne *et al.*, 2020). Par conséquent, pour positionner notre recherche dans la littérature existante, nous passons en revue les articles pertinents sur le problème stochastique sous incertitude de la demande et/ou du rendement de désassemblage.

*Incertain de demande* Barba-Gutiérrez et Adenso-Díaz (2009) proposent l'approche de la logique floue pour résoudre le problème LSD multi-période avec un système de désassemblage à plusieurs niveaux et un seul type de PFV. Dans ce travail, chaque demande non satisfaite dans chaque période sera reportée pour les périodes futures en tant que besoins déterministes. Dans la planification du désassemblage, cette approche est utilisée pour déterminer la date et la quantité de PFV à désassembler sans prendre en compte l'optimisation des différents coûts liés au processus de désassemblage. Cependant, l'optimisation de ce dernier est un élément important dans une entreprise. Kim et Xirouchakis (2010) considère le cas de plusieurs types de PFV à désassembler, mais chaque produit a une nomenclature à deux niveaux. En outre, le plan de désassemblage doit respecter une capacité de désassemblage limitée. Pour résoudre le problème, les auteurs développent un modèle stochastique, et une heuristique basée sur la relaxation lagrangienne a été proposée pour optimiser la somme des coûts de stockage, de reconfiguration et de pénalité de rupture de stock. Wang et Huang (2013); Fang *et al.* (2017); Quezada *et al.* (2020) considèrent le problème LSD stochastique multi-période à plusieurs niveaux et plusieurs produits dans les systèmes de reconditionnement. Wang et Huang (2013) propose un modèle de programmation stochastique à deux étapes pour trouver un compromis entre la robustesse de la solution et l'espérance du coût. Fang *et al.* (2017) formulent une approche de programmation stochastique à plusieurs étapes, et proposent un algorithme de relaxation lagrangienne pour résoudre le problème. Récemment, Quezada *et al.* (2020) s'est concentré sur un système multi-niveau qui comprend trois opérations telles que le désassemblage, la remise à neuf et l'opération de réassemblage. Cet article étend l'approche de program-

mation stochastique multi-étapes proposée par Fang *et al.* (2017) en considérant la demande stochastique, la quantité et la qualité de retour ainsi que les coûts de production. Pour résoudre le problème, les auteurs ont développé une approche de résolution exacte basée sur l'algorithme de branchement.

*Incertitude du rendement* : le rendement de désassemblage est le nombre de composants obtenus par l'opération de désassemblage d'une unité de produit. Le rendement aléatoire se produit lorsque la différence entre les quantités d'articles lancées et celles obtenues après le désassemblage n'est pas connue. À notre connaissance, seuls deux articles considèrent le problème LSD sous incertitude de rendement. Ces deux articles sont limités à des nomenclatures à deux niveaux, multi-produit et un horizon de planification mono-période. Inderfurth et Langella (2006) donne une heuristique permettant de réduire les coûts de rejet des composants, d'achat des produits et d'opération de désassemblage, tandis que Inderfurth *et al.* (2015) propose un modèle mathématique pour le cas spécifique avec trois composants finaux et deux sous-ensembles. Les auteurs étudient l'impact d'une mauvaise spécification du processus sur la solution du problème. Enfin, une analyse statistique montre que le rendement incertain a un impact important dans la refabrication des moteurs.

*Incertitude de la demande et du rendement* : à notre connaissance, Liu et Zhang (2018) est la seule publication qui étudie simultanément l'incertitude de la demande en composants et de rendement du désassemblage. Dans cette étude, l'incertitude du rendement est modélisée avec une distribution uniforme, alors que la demande suit une distribution normale. Les auteurs fournissent une programmation mixte non linéaire en nombres entiers pour le système de désassemblage à plusieurs périodes et à deux niveaux. Un algorithme basé sur l'approximation est proposé par la suite pour résoudre le problème.

Comme les incertitudes liées au rendement et à la demande ont un impact sur la planification des opérations de désassemblage, les installations de reconditionnement sont également confrontées à un délai incertain. En effet, après chaque opération de désassemblage, les pièces doivent être remises à neuf avant d'être expédiées aux clients. Le délai de désassemblage qui englobe le temps de remise à neuf, de réparation et de nettoyage des composants dépend de la qualité des pièces démontées et ne peut être estimé avec précision. Guide Jr (2000) indique que cette incertitude peut interrompre la livraison des pièces démontées, et peut augmenter les coûts de stockage et de rupture de stock. Néanmoins, à notre connaissance, l'incertitude des délais de désassemblage n'a pas encore été étudiée dans le problème LSD.

Pour combler cette lacune dans la littérature, le présent document examine le problème LSD sous incertitude du délai de désassemblage pour un seul type de PFV, un environnement à plusieurs périodes et une nomenclature de désassemblage à deux-niveau. L'objectif est d'optimiser le processus de désassemblage dans un environnement incertain.

### 3. Description et formulation du problème

#### 3.1. Description du problème

Un problème de planification des opérations de désassemblage sur plusieurs périodes est étudié. Un seul type de PFV avec une nomenclature à deux-niveau est considéré. Nous supposons que la demande et le nombre de composants obtenus par l'opération de désassemblage sont connus. Le PFV (premier niveau) est démonté en  $N$  composants  $i = 1, \dots, N$  qui forment le deuxième niveau. La capacité d'ap-

provisionnement en PFV est illimitée. Par contre, la capacité du temps de désassemblage par période est limitée. Si cette capacité n'est pas suffisante pour satisfaire les besoins en composants, une capacité supplémentaire peut être ajoutée engendrant un coût additionnel  $u_t$ .

Dans un environnement certain, le désassemblage des produits en fin de vie se fait dans un délai réel égal au délai planifié (voir Figure 1). Cependant, en raison de la complexité des opérations de désassemblage qui sont souvent effectuées manuellement, le délai réel de désassemblage est incertain à chaque période  $t$ . Ce délai réel ( $L_t$ ) peut être inférieur, égal ou supérieur au délai planifié. Une fois que le processus de désassemblage est lancé à la période  $t$ , tous les composants désassemblés sont reçus à la période  $t' = t + L_t$  (voir Figure 1). Dans la vie réelle, les délais de désassemblage sont souvent incertains pour différentes raisons telles que les retards de transport, les pannes de machines et les problèmes de qualité.

### Environnement certain

Délais planifiés ( $LT_t$ )  
=  
Délais réels ( $L_t$ )

| Période              | 0 | 1 | 2  | 3   | 4 | 5 | 6  | 7  | 8   |
|----------------------|---|---|----|-----|---|---|----|----|-----|
| Quantité de produits |   |   |    |     |   |   | 80 | 50 | 100 |
| Réception planifiée  |   |   |    |     |   |   | 80 |    |     |
| Lancement            |   |   | 80 | 100 |   |   | 50 |    |     |

$$LT_6 = 3, LT_7 = 1, LT_8 = 4,$$

$$L_6 = 3$$

### Environnement incertain

Délais planifiés ( $LT_t$ )  
≠  
Délais réels ( $L_t$ )

| Période              | 0 | 1 | 2  | 3 | 4 | 5 | 6  | 7  | 8   |
|----------------------|---|---|----|---|---|---|----|----|-----|
| Quantité de produits |   |   |    |   |   |   | 80 | 50 | 100 |
| Réception planifiée  |   |   |    |   |   |   |    |    |     |
| Lancement            |   |   | 80 |   |   |   |    |    |     |

$$L_6 \in \{L_6^-, \dots, L_6^+\}$$

**Figure 1.** Processus de désassemblage sous incertitude de délais.

L'incertitude des délais peut causer des coûts de rupture ou de stockage. Lorsque jusqu'à une période donnée, la somme des quantités reçues est supérieure à la somme des demandes à satisfaire. Dans ce cas, un coût de stockage est supporté. Dans le sens inverse, c'est à dire quand la somme des demandes jusqu'à une période donnée est supérieure à la somme des quantités reçues, alors un coût de rupture est engendré. Dans cette recherche, nous avons traité le cas de croisement entre les ordres de désassemblage. Dans cette situation, si un ordre  $a$  est placé avant l'ordre  $b$ , ce dernier peut arriver avant le premier. L'objectif ici est d'optimiser les quantités de PFV à désassembler afin de minimiser la somme des coûts de stockage et de rupture pour tous les composants désassemblés, ainsi que les coûts de reconfiguration et de dépassement de la capacité sur un horizon de planification limité.

### 3.2. Formulation mathématique

Il existe plusieurs méthodes pour gérer l'incertitude dans les problèmes de planification. La méthode statistique, qui exprime les délais stochastiques sous forme des variables aléatoires avec des fonctions de distribution connues, est répandue dans de nombreuses recherches. Cependant, obtenir la fonction de distribution exactement n'est pas facile généralement en raison du manque d'informations, et le problème stochastique qui en résulte est notoirement difficile à résoudre même si la distribution exacte est donnée Fang *et al.* (2017). Dans cette partie, nous utilisons une approche basée sur des scénarios pour modéliser les délais stochastiques en fonction des données historiques. En suivant cette approche, nous supposons qu'il existe des cas possibles de délais stochastiques. Ces cas sont bornés sur des intervalles fermés connus :  $[L^-, L^+]$  et chaque cas correspond à un scénario. La probabilité de sa réalisation est connue et décrite comme suit : Si un processus de désassemblage commence à la période  $t$ , tous les composants sont disponibles à la période  $t + L_t^\omega$  pour le scénario  $\omega$ , et la probabilité du scénario  $\omega$  est de  $P_\omega = \prod_{t \in \mathcal{T}} P(L_t^\omega)$  où  $\Omega$  est l'ensemble des scénarios possibles et  $\omega$  est l'indice de scénario de délai,  $\omega \in \Omega$ .

Pour sélectionner le plan de désassemblage approprié et résoudre de manière optimale le problème de planification étudié, un modèle de programme stochastique linéaire en nombres entiers à deux étapes est proposé. Les décisions de la première étape doivent être prises avant toutes réalisations de paramètre incertain telles que la quantité des produits à désassembler et les quantités des heures supplémentaires sur l'horizon de planification. Les décisions de la deuxième étape sont prises consécutivement tels que les niveaux de stock et de rupture en stock de chaque composant à chaque période et chaque scénario. La liste complète des notations utilisées dans le présent article est donnée dans Tableau 3.1 :

Tableau 3.1.: Notations.

#### Indices

|          |   |
|----------|---|
| $t$      | indice de la période $t, \forall t \in \mathcal{T}$ |
| $i$      | indice de composant $i, \forall i \in \mathcal{N}$  |
| $\omega$ | indice de scénario $\omega, \omega \in \Omega$      |

#### Paramètres

|               |  |
|---------------|--|
| $\mathcal{T}$ | ensemble des périodes de l'horizon de planification  |
| $\mathcal{N}$ | ensemble des composants ( $N = \#\mathcal{N}$ )  |
| $\Omega$      | ensemble des scénarios possibles   |
| $R_i$         | rendement du désassemblage de composant $i$  |
| $D_{i,t}$     | demande externe de composant $i$ à la période $t$  |
| $I_{i,0}$     | niveau de stock initial de composant $i$   |
| $L_t^\omega$  | délai aléatoire à la période $t$ pour le scénario $\omega$ . Il varie entre $L^-$ et $L^+$ |
| $h_i$         | coût unitaire de stockage de composant $i$   |
| $s_t$         | coût d'installation à la période $t$   |
| $b_i$         | coût unitaire de rupture de stock de composant $i$   |
| $u_t$         | pénalité de dépassement d'une heure supplémentaire à la période $t$                        |
| $G$           | temps de désassemblage   |
| $M$           | Un grand nombre  |

#### Fonctions

|                     |                        |
|---------------------|------------------------|
| $\mathbb{E}(\cdot)$ | Espérance mathématique |
| $P(\cdot)$          | Valeur de probabilité  |



### Variables de décision de la première étape

|       |  |
|-------|--|
| $Z_t$ | quantité des produits lancée au désassemblage à la période $t$ |
| $Y_t$ | indicateur binaire de désassemblage à la période $t$           |
| $O_t$ | heures supplémentaires de désassemblage à la période $t$       |

### Variables de décision de la deuxième étape

|                  |  |
|------------------|--|
| $H_{i,t}^\omega$ | niveau de stock de composant $i$ à la période $t$ pour le scénario $\omega$                  |
| $B_{i,t}^\omega$ | niveau de rupture de composant $i$ à la période $t$ pour le scénario $\omega$                |
| $I_{i,t}^\omega$ | niveau de stock à la fin de la période $t$ . Il est égal à $H_{i,t}^\omega - B_{i,t}^\omega$ |

#### 3.2.1. Programmation linéaire en nombres entiers

Le problème LSD multi-période avec des délais de désassemblage stochastiques peut être formulé en tant que modèle de programmation linéaire en nombres entiers stochastique basé sur les scénarios. L'espérance mathématique du coût total notée  $\mathbb{E}(CT)$  peut être calculée en considérant toutes les combinaisons possibles de  $L_t^\omega$ . Notons que le nombre total de scénarios possibles est  $|\Omega| = \prod_{t \in \mathcal{T}} (L^+ - L^- + 1)$ .

La fonction objectif qui cherche à minimiser l'espérance mathématique de coût total peut être exprimé dans l'équation (1) :

$$\mathbb{E}(CT) = \min \sum_{t \in \mathcal{T}} \left( \sum_{\forall i \in \mathcal{N}} \sum_{\omega \in |\Omega|} P_\omega (h_i \cdot H_{i,t}^\omega + b_i \cdot B_{i,t}^\omega) + s_t \cdot Y_t + u_t \cdot O_t \right) \quad [1]$$

Les contraintes (2) définissent le niveau de stock de chaque composant  $i$  à la fin de chaque période  $t$  pour le scénario  $\omega$  :

$$H_{i,t}^\omega - B_{i,t}^\omega = I_{i,0} + \sum_{\tau \in \mathcal{T} \mid \tau + L_i^\omega \leq t} R_i Z_\tau - \sum_{\tau=1}^t D_{i,\tau} \quad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \Omega \quad [2]$$

Les contraintes (3) garantissent qu'un coût de désassemblage est généré dans une période  $t$  si une opération de désassemblage doit être effectuée à cette période, où le grand  $M$  est égal à la demande cumulative maximale de tous les articles sur l'horizon de planification (voir [8]) :

$$Z_t \leq Y_t \cdot M \quad \forall t \in \mathcal{T} \quad [3]$$

Les équations dans (4) représentent la quantité de désassemblage disponible à chaque période  $t$  :

$$G \cdot Z_t \leq C_t + O_t \quad \forall t \in \mathcal{T} \quad [4]$$

Les équations (5-7) fournissent les contraintes de non-négativité des variables de décision :

$$Z_t, O_t \geq 0 \quad \forall t \in \mathcal{T} \quad [5]$$

$$H_{i,t}^\omega, B_{i,t}^\omega \geq 0 \quad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \Omega \quad [6]$$

$$Y_t \in \{0, 1\} \quad \forall t \in \mathcal{T} \quad [7]$$

$$M = \max_{i \in \mathcal{N}} \left( \frac{\sum_{\forall i \in \mathcal{N}} \sum_{\forall t \in \mathcal{T}} D_{i,t}}{R_i} \right) \quad [8]$$

### 3.2.2. Simulation Monte Carlo et Programmation linéaire en nombres entiers

La complexité du problème peut augmenter de manière exponentielle si on considère un grand nombre des scénarios  $|\Omega|$  qui représente toutes les combinaisons possibles des délais stochastiques. Par la suite, la résolution du modèle (1)-(7) devient impossible.

L'espérance de coût total peut être estimée en utilisant la simulation *Monte Carlo (MC)*. Le principe de cette approche consiste à utiliser un échantillon discret obtenu par simulation pour approximer une distribution de probabilité que l'on ne sait pas calculer analytiquement (Decker, 1991; Fishman, 1996; Hammersley et Handscomb, 1964). Typiquement, cet échantillon pourra être utilisé pour estimer des espérances mathématiques par des moyennes empiriques.

Comme nous l'avons mentionné, le délai de désassemblage à chaque période ( $L_t$ ) est modélisé par une variable discrète aléatoire caractérisée par une distribution de probabilité connue qui varie entre  $L^-$  et  $L^+$ . Une estimation du coût total est obtenue par la génération d'un ensemble d'échantillons aléatoires indépendants de  $L_t^1, \dots, L_t^{|\vartheta|}$  de taille  $\vartheta$ , de sorte que  $\vartheta \subset \Omega, \omega \in \vartheta$ . Pour un scénario donné, chaque délai de démontage est généré en utilisant la distribution de probabilité  $P$ .

En utilisant les scénarios aléatoires générés de taille  $\vartheta$ , l'espérance mathématique définie dans (1) peut être estimée par un coût total moyen (CTM). Par la suite, le modèle PLNE peut être converti en un modèle MC-PLNE suivant :

$$CTM = \min \sum_{\forall t \in \mathcal{T}} \left( \sum_{\forall i \in \mathcal{N}} \sum_{\omega \in \vartheta} \frac{1}{|\vartheta|} \left( h_i \cdot H_{i,t}^\omega + b_i \cdot B_{i,t}^\omega \right) + s_t \cdot Y_t + u_t \cdot O_t \right) \quad [9]$$

$$H_{i,t}^\omega - B_{i,t}^\omega = I_{i,0} + \sum_{\tau \in \mathcal{T} \mid |\tau + L_i^\omega \leq t} R_i Z_\tau - \sum_{\tau=1}^t D_{i,\tau} \quad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \vartheta \quad [10]$$

$$H_{i,t}^\omega, B_{i,t}^\omega \geq 0 \quad \forall i \in \mathcal{N}, \forall t \in \mathcal{T}, \forall \omega \in \vartheta \quad [11]$$

sous contraintes (3-5) et (7).

L'équation (9) cherche à minimiser le coût total moyen (CTM). Ce CTM à minimiser est maintenant une estimation du critère exact. Les contraintes (10) et (11) fournissent une estimation de niveau de stock à la fin de chaque période  $t$ , en se basant sur les échantillons aléatoires générés.

*En se basant sur la loi des grands nombres, nous avons :*

$$\lim_{\vartheta \rightarrow +\infty} P(|CTM - \mathbb{E}(CT)| \geq \epsilon) = 1$$

Cette loi stipule que pour un grand nombre d'échantillon ( $\vartheta$ ) et pour tout petit nombre positif  $\epsilon$ , la probabilité que  $\epsilon$  converge presque sûrement vers l'espérance du coût total est égale à 1. Autrement dit, en utilisant un nombre de scénarios élevé, on peut obtenir une estimation assez précise du critère.

Pour trouver le nombre de scénarios nécessaires pour obtenir une bonne estimation du coût total exact avec une précision donnée (taux de confiance, petite déviation), nous nous appuyons sur la théorie des probabilités (Rubinstein, 2006). Selon Israel (1992), pour trouver un compromis entre le nombre des scénarios et la précision des résultats, la taille de l'échantillon ( $\vartheta$ ) peut être calculée à l'aide de l'équation



(12) :

$$\vartheta = \frac{F^2 \cdot p \cdot q}{e^2} \quad [12]$$

Où “ $F^2$  est l’abscisse de la courbe normale qui coupe une zone  $\alpha$  aux queues ( $1 - \alpha$ ) qui est égal au niveau de confiance souhaité, (par exemple, 95%),  $e$  est le niveau de précision souhaité,  $p$  est la proportion estimée d’un attribut présent dans la population, et  $q$  est égal à  $1 - p$ . La valeur de  $F$  est obtenue dans les tableaux statistiques sous la courbe normale”(Israel, 1992). Nous supposons que  $p = .5$  (variabilité maximale) et que nous désirions un niveau de confiance de 95% avec une précision de  $\pm 3\%$ , le résultat nécessite une simulation de 1000 échantillons aléatoires indépendants à chaque période.

## 4. Algorithme génétique

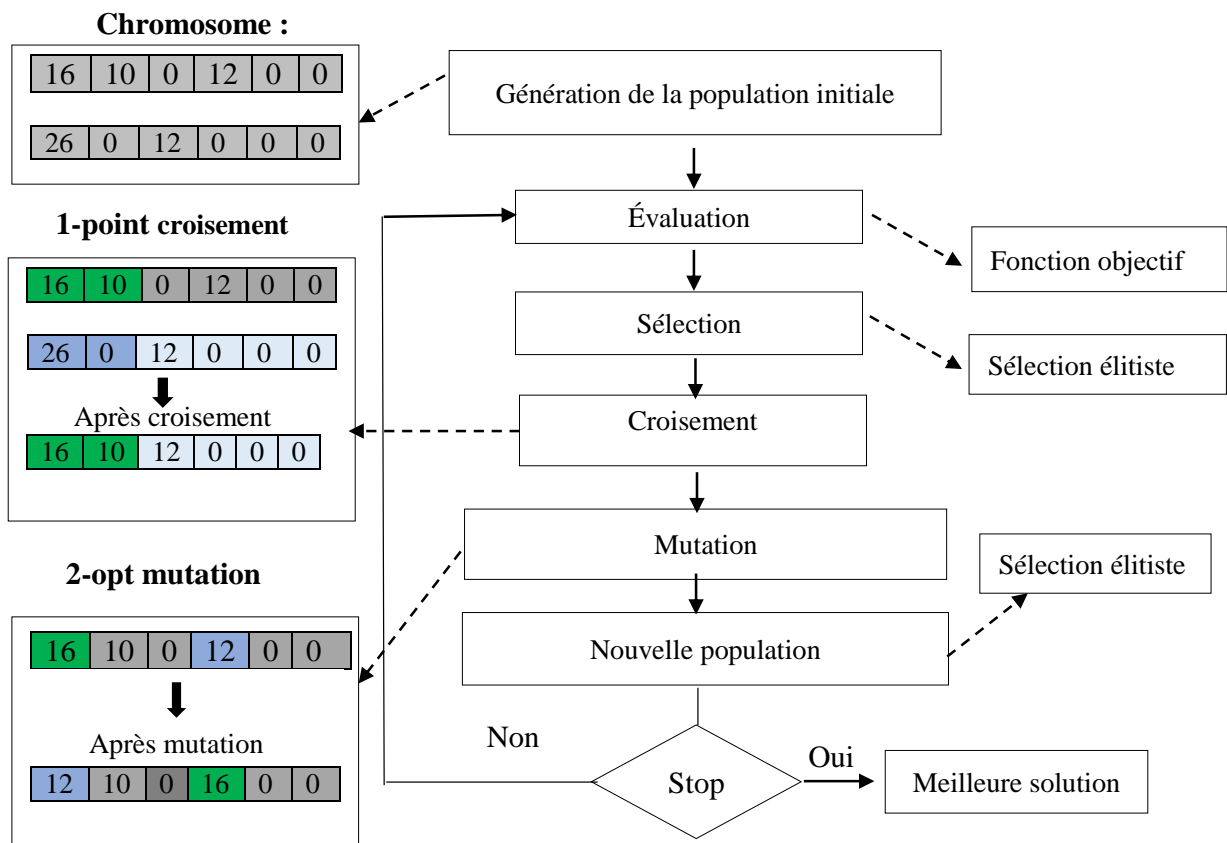
Bien que l’approche basée sur la simulation MC soit intéressante, elle peut souffrir de l’impossibilité d’obtenir des solutions optimales à mesure que la complexité du problème augmente. Une méthode de résolution approchée est nécessaire pour résoudre les problèmes de grandes tailles.

Diverses méta-heuristiques sont proposées dans la littérature afin de résoudre des problèmes d’optimisation. Les algorithmes génétiques semblent être les plus adaptés pour traiter des problèmes de planification des réapprovisionnement en produits finis ou en composants (voir, par exemple Ben-Ammar *et al.*, 2020, 2019; Ben-Ammar et Dolgui, 2018). En effet, la représentation des solutions et les opérateurs de reproduction (croisement et mutation) peuvent facilement être mis en place pour le problème considéré. En outre, dans la mesure où il n’existe pas de contraintes particulières, une méta-heuristique basée sur une recherche locale, telles que la recherche Tabou ou le recuit simulé, demanderait l’exploration d’un grand nombre de voisins et donc un temps de calcul significatif (Ben Ammar, 2014).

Dans cette recherche, un algorithme génétique (AG) couplé avec le modèle MC-PLNE est proposé pour calculer la quantité de produits à lancer à chaque période, afin de minimiser le coût total moyen exprimé en (9). Cette approche est modélisée et inspirée par le processus d’évolution naturelle qui est basée sur une population d’individus générée aléatoirement (Goren *et al.*, 2010). Figure 2 présente un organigramme de l’AG proposé et les différentes étapes de ce dernier seront présentées, en détail, dans les sous sections suivantes.

### 4.1. Génération de la population initiale

Dans la mise en œuvre d’un algorithme génétique, la première étape consiste à créer la population initiale d’individus de taille  $N_{pop}$ . Chaque individu est codé par un chromosome. Les variables de décisions dans le modèle MC-PLNE sont  $H_{i,t}^\omega, B_{i,t}^\omega, O_t, Z_t > 0$  si  $Y_t = 1$  et  $Z_t = 0$  si  $Y_t = 0$ . Dans ce papier, nous ne codons que la variable de décision  $Z_t$  qui représente la quantité de produits en fin de vie lancée à chaque période  $t$ , en tant que chromosomes. Les autres variables seront considérées comme des variables qui dépendent de  $Z_t$ . Notons que chaque chromosome est composé par des gènes. Dans notre cas, chaque gène d’un chromosome représente une quantité de produits à désassembler à chaque période  $t$ . La longueur du chromosome est égale au nombre de périodes  $T$ . Une heuristique constructive est proposée pour construire chaque individu. Cette heuristique est implémentée en plusieurs étapes. Figure 3 présente



**Figure 2.** Organigramme d'un algorithme génétique.

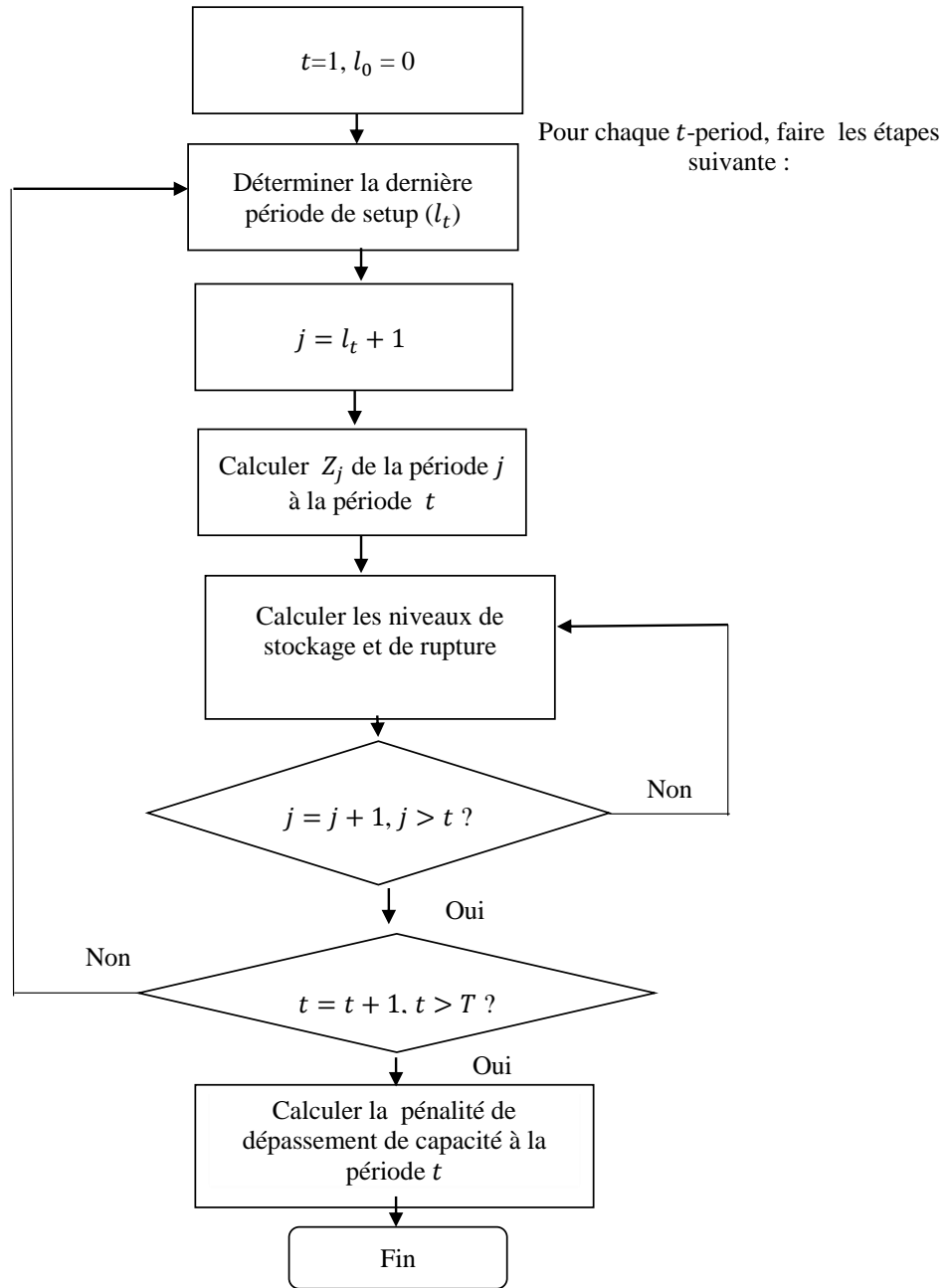
un schéma général de fonctionnement de cette heuristique. Les différentes étapes seront expliquées, en détail, dans ce qui suit.

**étape 1 :** La première étape consiste à générer aléatoirement les périodes de désassemblage ( $Z_t > 0$  et  $Y_t = 1$ ). Soit  $j < t$  la période telle que  $Z_j > 0$  et  $Z_k = 0$ , pour  $k = j + 1, \dots, t$ . Cela implique que  $Z_j$  est la quantité de produit nécessaire pour satisfaire toutes les demandes de la période  $j$  jusqu'à la période  $t$ , avec  $j \leq k \leq t$ . Par conséquent, le problème peut être décomposé en sous problèmes de  $t$ -période (un des périodes de  $j$  à  $t$ , et les autres des périodes de  $t + 1$  à  $T$ ). Par la suite,  $t$  est la dernière période de désassemblage de chaque sous-problème de la période  $t$ .

**étape 2 :** La deuxième étape permet de calculer pour chaque  $t$ -période, la quantité de désassemblage ( $Z_j$ ) à la période  $j$ . La valeur exacte de  $Z_j$ , est déterminée afin de satisfaire la demande maximale parmi tous les composants de la période  $j$  à  $t$  et tous les scénarios  $\vartheta$ . On peut déterminer  $Z_j$  en utilisant l'équation (13) sans tenir compte de la capacité de désassemblage et prendre ce plan comme un plan final.

$$Z_j = \max_{[\forall i, \forall \omega]} \left\lceil \frac{\sum_{i=1}^N \sum_{k=j}^t D_{i,k} - H_{i,j-1}^\omega + B_{i,j-1}^\omega}{R_i} \right\rceil \quad [13]$$

avec  $\lceil \cdot \rceil$  est le plus petit nombre entier supérieur ou égal à  $\cdot$ .



**Figure 3.** Procédure de la génération de chaque individu dans la population initiale.

**étape 3 :** Pour chaque  $t$ –période, nous pouvons calculer les niveaux de stock et de rupture en stock pour chaque composant  $i$  pour le scénario  $\omega$  à la fin de la période  $k$  ( $j \leq k \leq t$ ) en utilisant respectivement les équations (14) et (15).

$$H_{i,k}^{\omega} = (H_{i,k-1}^{\omega} + R_i \cdot Z_{t'|t+L_t^{\omega}=t'} - D_{i,k} + B_{i,k-1}^{\omega})^+ \quad \text{For } k = j, \dots, t, \forall \omega \in \vartheta, \forall i \in \mathcal{N} \quad [14]$$

$$B_{i,k}^{\omega} = (H_{i,k-1}^{\omega} + R_i \cdot Z_{t'|t+L_t^{\omega}=t'} - D_{i,k} + B_{i,k-1}^{\omega})^- \quad \text{For } k = j, \dots, t, \forall \omega \in \vartheta, \forall i \in \mathcal{N} \quad [15]$$

où  $(X)^+ = \max\{0, X\}$  et  $(X)^- = \max\{0, -X\}$

**étape 4 :** Après avoir déterminé toutes les quantités de produits à lancer tout au long de l'horizon de planification, calculer pour chaque période  $t$  la capacité supplémentaire en utilisant l'équation (16) et en pénalisant sa violation par la suite.

$$O_t = \left( G \times Z_t - C_t \right)^+ \quad \forall t \in \mathcal{T} \quad [16]$$

## 4.2. Évaluation

Nous utilisons l'équation (9) pour évaluer le coût total moyen de chaque individu de la population.

## 4.3. Sélection

La sélection de certains individus au sein de la population peut générer une nouvelle population. Il existe de nombreuses méthodes de sélection dans la littérature telle que la sélection par rang, la sélection élitiste, la sélection Boltzmann, la sélection à la roulette, etc. (Belkhamza *et al.*, 2018). Dans nos expériences, nous optons pour la méthode de sélection élitiste pour sélectionner les individus pour la reproduction d'une nouvelle population. Dans cette méthode chaque individu est sélectionné en fonction de ses performances. En effet,  $\frac{N_{pop}}{2}$  individus les plus performants sont sélectionnés pour subir les opérations de reproduction (croisement et mutation). Il est nécessaire de noter qu'à chaque génération, les doublons sont supprimés et remplacés par des individus moins performant afin de garder une diversification suffisante de la population. Ce choix évitera de converger rapidement vers un optimum local.

## 4.4. Croisement

L'opérateur de croisement permet de produire un nouvel individu à partir de parents sélectionnés pour obtenir une "meilleure" solution (Belkhamza *et al.*, 2018) (dans notre travail, une meilleure quantité des produits à désassembler à chaque période  $t$ ). Un opérateur de croisement peut former un nouvel individu (fils) à partir de deux individus parents. L'idée principale est d'échanger les gènes des parents afin de donner des enfants qui portent des propriétés combinées. Nous choisissons un opérateur de croisement à un seul point. Dans ce cas, le premier segment du premier parent est suivi par le deuxième segment du deuxième parent (voir 1-point croisement dans Figure 2). La création d'un enfant à partir d'un couple de parents est effectuée avec une probabilité de croisement  $P_c$  et le point de croisement est généré de manière aléatoire.

## 4.5. Mutation

L'opérateur de mutation permet de conserver la diversification et d'échapper d'un optimum local éventuel. Il s'agit de modifier au hasard certains individus de notre population en modifiant un gène par un autre (voir mutation 2-opt dans la Figure 2). L'opérateur de mutation est réalisé selon une probabilité  $P_m$ .

## 4.6. Nouvelle population

Une fois que nous avons créé de nouveaux individus par sélection, croisement et mutation, nous devons sélectionner ceux qui continueront à participer à l'amélioration de notre population. Nous avons utilisé ici la sélection élitiste pour sélectionner une nouvelle population de taille  $N_{pop}$ . Les individus de la nouvelle population sont sélectionnés à partir d'une population totale composée de la population initiale et des fils obtenus par croisement et mutation.

## 4.7. Critère d'arrêt

Le processus de l'AG est répété jusqu'à ce que le critère d'arrêt soit rempli. Le temps de calcul est un facteur indispensable dans la prise de décision. Pour cela, comme dans Liu et Zhang (2018), nous limitons le temps de calcul à 600 secondes pour l'exécution de l'algorithme proposé.

Les approches proposées ont été évaluées dans la section suivante. Plusieurs tests ont été envisagés pour examiner la robustesse de ces approches.

## 5. Résultats numériques

Afin de montrer le comportement des différentes approches, plusieurs expériences numériques ont été réalisées. Trois méthodes de résolution sont utilisées dans ce papier pour résoudre le problème étudié. La première approche basée sur le modèle PLNE, donne des solutions exactes en utilisant toutes les combinaisons possibles de délais aléatoires. La deuxième approche basée sur le modèle MC-PLNE, donne des solutions estimées en utilisant des échantillons générés aléatoirement. Toutes les formulations de ces deux approches sont implémentées en C en utilisant le solveur CPLEX 12.6 dans le délai fixe de 3600 secondes. Finalement, l'AG, est implémenté en java sur un PC avec processeur Intel (R) Core™ i7-5500 CPU @ 2.4 GHz et 8 Go de RAM sous Windows 10 Professional.

### 5.1. Génération d'instances

Afin de comparer et d'évaluer les méthodes PLNE, MC-PLNE et AG, les expériences de calcul ont impliqué différentes tailles et niveaux de complexité du problème. Les exemples numériques sont divisés en deux ensembles des données, où chaque ensemble se compose de plusieurs problèmes générés de manière aléatoire.

Le premier ensemble comprend un système de désassemblage de 15 composants et comprend 3 différents problèmes respectivement de 10, 20 et 30 périodes de temps. Les problèmes de cet ensemble sont de petites et moyennes tailles et ils ne peuvent pas tous être résolus par le modèle PLNE. Le second ensemble se compose de 5 problèmes de différentes tailles comme présenté dans Tableau 5.2. Ces problèmes sont assez importants et ils sont résolus que par l'algorithme génétique dans une limite d'une heure de calcul. Pour chaque problème, le délai de désassemblage a été pris comme une variable aléatoire indépendante. Ce dernier est borné sur un intervalle connu dont les limites inférieures et supérieures sont générées de façon aléatoires comme présentées dans Tableau 5.2.

**Tableau 5.2.** Caractéristiques de chaque problème dans le second ensemble.

| Ensemble | $L_t$             | $ \mathcal{N} $ | $ \mathcal{T} $ |
|----------|-------------------|-----------------|-----------------|
| 1        | $D \sim U(1, 20)$ | 20              | 30              |
| 2        | $D \sim U(1, 15)$ | 30              | 20              |
| 3        | $D \sim U(1, 20)$ | 30              | 30              |
| 4        | $D \sim U(1, 15)$ | 40              | 20              |
| 5        | $D \sim U(1, 20)$ | 40              | 30              |

Pour chaque problème (ensemble 1 et 2), la nomenclature de produit est générée de manière aléatoire où le rendement de désassemblage ( $R_i$ ) est généré en suivant la  $D \sim U(1, 5)$ . Sans perte de généralité, le niveau de stock initial de tous les composants a été fixé à 0. Tous les autres paramètres sont générés comme dans Tableau 5.3. Nous précisons que,  $D \sim U(a, b)$  signifie que le paramètre suit la distribution uniforme discrète caractérisée par l'intervalle  $[a, b]$ .

**Tableau 5.3.** Caractéristiques des données.

| Paramètres      | Valeur               |
|-----------------|----------------------|
| $D_{i,t}$       | $D \sim U(10, 100)$  |
| $h_i$           | $D \sim U(12, 20)$   |
| $s_t$           | $D \sim U(0, 1000)$  |
| $b_i$           | $2 \times h_i$       |
| $u_t$           | $D \sim U(20, 25)$   |
| $C_t$           | $D \sim U(280, 480)$ |
| $G$             | $D \sim U(5, 15)$    |
| $ \mathcal{V} $ | 1000                 |

Le bon choix des valeurs des paramètres de l'algorithme génétique fait une différence dans la convergence de la meilleure solution obtenue. Après des tests préliminaires, ces paramètres comprennent le taux de croisement ( $P_c = 0,8$ ), le taux de mutation ( $P_m = 0,1$ ) et la taille de la population ( $N_{pop} = 200$  chromosomes). Nous avons arrêté l'algorithme après 500 générations car nous avons constaté que l'espace de recherche est suffisamment exploré et que la population évolue rapidement.

## 5.2. Performance de l'algorithme génétique

Pour valider les performances de l'algorithme génétique, les résultats des tests sont mentionnés dans cette section. Ici, nous avons limité le temps d'exécution de CPLEX à 3600 secondes, afin d'obtenir la solution optimale exacte ( $\mathbb{E}(CT)$ ) et la solution optimale estimée (CTM). Les mesures de performance utilisées dans ce papier sont :

1. le temps de calcul en secondes (T(s)), nécessaires pour obtenir  $\mathbb{E}(CT)$ , CTM et la meilleure solution trouvée par l'AG après  $n$  générations ( $bestSol_n$ );



2. l'écart en pourcentage par rapport à la limite inférieure ou à la valeur de la solution optimale ( $G^*$ ). Ce dernier peut être obtenue directement à partir de CPLEX ;
3. l'écart en pourcentage ( $G^{++}$ ) entre CTM et  $\mathbb{E}(CT)$  comme indiqué dans (17).
4. l'écart en pourcentage ( $G^+$ ) entre  $bestSol_n$  trouvée au bout de 500 générations et le CTM comme indiqué dans (18).

$$G^{++} = \frac{CTM - \mathbb{E}(CT)}{\mathbb{E}(CT)} \times 100 \quad [17]$$

$$G^+ = \frac{bestSol_n - CTM}{CTM} \times 100 \quad [18]$$

Tableau 5.4 présente respectivement, les résultats d'optimisation obtenus avec les modèles PLNE, MC-PLNE et AG pour les problèmes du premier ensemble. Plus précisément, ce tableau donne l'espérance mathématique du coût total, le coût total moyen et la meilleure solution obtenue en résolvant l'algorithme génétique. Le temps de calcul et les écarts sont également présentés dans ce tableau. Dans certains cas, les résultats ne peuvent pas être générés par le modèle PLNE, ce qui est indiqué par un "-". Par conséquent, aucune comparaison sera faite, ce qui est indiqué par un "\*".

D'une manière générale, nous constatons que le modèle PLNE ne peut résoudre que des problèmes avec 10 périodes. Cela est due au nombre exponentiel de scénarios possibles qui dépend fortement du nombre de périodes. De la même manière, dans une limite de 3600 secondes, l'approche qui combine la simulation MC et le modèle PLNE se limite à un système de désassemblage de 15 composants et 30 périodes sous un nombre d'échantillons égal à 1000. Nous remarquons aussi que l'écart d'optimalité ( $G^{++}$ ) par rapport à la solution exacte obtenue en résolvant le premier problème ne dépasse pas 0,9%. Cela prouve la convergence de CTM vers la valeur exacte de l'espérance mathématique. En d'autres termes, un grand nombre d'échantillons est nécessaire pour trouver la bonne approximation de la solution exacte.

Compte tenu d'un faible écart moyen de  $G^+$  (moins de 0,5% pour tous les problèmes) et de  $G^{++}$  moins de 1,10% pour le premier problème, l'AG joue un rôle important pour trouver une solution très proche de l'optimum dans un délai très raisonnable.

Pour montrer l'efficacité de l'AG proposé pour les tests de grandes tailles, nous traitons les problèmes de deuxième ensemble dans ce que suit. Comme mentionné précédemment, compte tenu de la nature aléatoire de l'AG, nous avons réalisé 100 exécutions indépendantes du même ensemble de données de chaque problème. Pour chaque problème, la meilleure et la pire solutions connues, respectivement (MSC) et (PSC) sont sélectionnées. Par la suite, l'écart en moyenne entre ces deux solutions est obtenu comme présenté dans Tableau 5.5. Ce tableau montre clairement que les résultats fondés par l'AG sont stables. En effet, l'écart total à partir de la MSC ne dépasse pas 0,6 % pour tous les problèmes de l'ensemble 2.

Le temps de calcul nécessaire pour obtenir les meilleures solutions est limité à 600(s). Tableau 5.5, montre bien que l'approche proposée peut résoudre des systèmes de désassemblage allant jusqu'à 40 composants et 30 périodes. Cependant, le temps de résolution croît de manière considérable avec l'accroissement des nombre de périodes.

**Tableau 5.4.** Performances de l’algorithme génétique pour les problèmes de premier ensemble.

| <i>(a) Solutions PLNE</i> |                 |                 |            |        |        |           |  |
|---------------------------|-----------------|-----------------|------------|--------|--------|-----------|--|
| Problème                  | $ \mathcal{N} $ | $ \mathcal{T} $ | $ \Omega $ | $L_t$  | T(s)   | $G^*$ (%) |  |
| 1                         | 15              | 10              | $2^{10}$   | [4,5]  | 675,22 | 0.00      |  |
| 2                         | 15              | 20              | $15^{20}$  | [1,15] | -      | -         |  |
| 3                         | 15              | 30              | $20^{30}$  | [1,20] | -      | -         |  |

| <i>(b) Solutions MC-PLNE</i> |                 |                 |               |        |        |           |              |
|------------------------------|-----------------|-----------------|---------------|--------|--------|-----------|--------------|
| Problème                     | $ \mathcal{N} $ | $ \mathcal{T} $ | $ \vartheta $ | $L_t$  | T(s)   | $G^*$ (%) | $G^{++}$ (%) |
| 1                            | 15              | 10              | 1000          | [4,5]  | 557,18 | 0,00      | 0,9          |
| 2                            | 15              | 20              | 1000          | [1,15] | 2120   | 0,00      | *            |
| 3                            | 15              | 30              | 1000          | [1,20] | 3578   | 0,00      | *            |

| <i>(c) Solutions AG</i> |                 |                 |               |        |       |           |              |
|-------------------------|-----------------|-----------------|---------------|--------|-------|-----------|--------------|
| Problème                | $ \mathcal{N} $ | $ \mathcal{T} $ | $ \vartheta $ | $L_t$  | T(s)  | $G^+$ (%) | $G^{++}$ (%) |
| 1                       | 15              | 10              | 1000          | [4,5]  | 13,91 | 0,11      | 1.07         |
| 2                       | 15              | 20              | 1000          | [1,15] | 21,25 | 0,23      | *            |
| 3                       | 15              | 30              | 1000          | [1,20] | 31,50 | 0,10      | *            |

**Tableau 5.5.** Écart et Temps de calcul en moyenne pour chaque problème dans le deuxième ensemble.

| Problème | Écart (%) | T(s)   |
|----------|-----------|--------|
| 1        | 0.47      | 120.78 |
| 2        | 0.57      | 363.23 |
| 3        | 0.55      | 150.36 |
| 4        | 0.60      | 423.36 |
| 5        | 0.47      | 600    |

## 6. Conclusion

Dans ce papier, un problème de planification de désassemblage stochastique à plusieurs périodes a été traité. Les délais d’obtentions de tous les composants ont été pris comme des variables aléatoires avec des distributions de probabilité connue. Les cas d’une nomenclature à deux niveaux et un seul type de produit ont été considérés. En adoptant une approche basée sur des scénarios pour exprimer les délais de désassemblage stochastiques, le problème est formulé comme un modèle PLNE à deux étapes. Ce modèle fournit une évaluation exacte du coût total tout en considérant tous les scénarios possibles de délai de désassemblage. L’objectif est d’identifier la quantité des produits à désassembler à chaque période afin de minimiser l’espérance mathématique du coût total. Les résultats des tests montrent que la résolution du problème devient impossible en utilisant un système de désassemblage de 20 périodes. Pour surmonter cette difficulté, la combinaison de la programmation linéaire en nombres entiers et la simulation Monte Carlo (MC-PLNE) a été développée. Ce modèle est basé sur des échantillons générés aléatoirement. L’efficacité de cette approche a été testée et la convergence vers un coût exact est prouvée avec succès. Cependant, le modèle MC-PLNE ne peut pas résoudre le problème étudié lorsqu’il devient

de grandes tailles (15 composants et 30 périodes). Cela est due au fait que l'augmentation du nombre de périodes) et des composants, augmentent la taille du problème. Enfin, un algorithme génétique classique est construit pour résoudre ce type de problème et trouver des solutions très proche de l'optimale. La performance de cette approche a été testée en résolvant des problèmes générés aléatoirement.

Ce travail peut être étendu de plusieurs façons. Tout d'abord, la plupart des travaux qui traitent le problème LSD sous incertitude se sont focalisés sur les systèmes à deux niveaux et un seul type de produit en fin de vie. De point de vue pratique, les entreprises traitent généralement les systèmes de désassemblage portant sur des nomenclatures à plusieurs niveaux et plusieurs types de produits. Un domaine de recherche prometteur pour l'avenir peut être la résolution du problème du lot-sizing à plusieurs niveaux et plusieurs types des produits. En suite, l'hybridation de l'algorithme génétique avec d'autres méta-heuristiques ou heuristiques pourrait être un axe pour des futures recherches.

# Bibliographie

- BARBA-GUTIÉRREZ, Y. et ADENSO-DÍAZ, B. (2009). Reverse MRP under uncertain and imprecise demand. *The International Journal of Advanced Manufacturing Technology*, 40(3-4):413–424.
- BELKHAMSA, M., JARBOUI, B. et MASMOUDI, M. (2018). Two metaheuristics for solving no-wait operating room surgery scheduling problem under various resource constraints. *Computers & Industrial Engineering*, 126:494–506.
- BEN AMMAR, O. (2014). *Planification des réapprovisionnements sous incertitudes pour les systèmes d'assemblage à plusieurs niveaux*. Thèse de doctorat, Saint-Etienne, EMSE.
- BEN-AMMAR, O., BETTAYEB, B. et DOLGUI, A. (2019). Optimization of multi-period supply planning under stochastic lead times and a dynamic demand. *International Journal of Production Economics*, 218:106–117.
- BEN-AMMAR, O., CASTAGLIOLA, P., DOLGUI, A. et HNAIEN, F. (2020). A hybrid genetic algorithm for a multilevel assembly replenishment planning problem with stochastic lead times. *Computers & Industrial Engineering*, 149:106794.
- BEN-AMMAR, O. et 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.
- BRACH, F. et BRUSSET, X. (2014). La reverse logistique en tant qu'avantage compétitif. *Revue Française de Gestion Industrielle*, 33:33–53.
- DECKER, K. M. (1991). The monte carlo method in science and engineering : Theory and application. *Computer methods in applied mechanics and engineering*, 89(1-3):463–483.
- FANG, C., LIU, X., PARDALOS, P. M., LONG, J., PEI, J. et 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. *Science Business Media*.
- GODICHAUD, M. et AMODEO, L. (2018). Economic order quantity for multistage disassembly systems. *International Journal of Production Economics*, 199:16–25.
- GODICHAUD, M. et AMODEO, L. (2019). Eoq inventory models for disassembly systems with disposal and lost sales. *International Journal of Production Research*, 57(18):5685–5704.
- GOREN, H. G., TUNALI, S. et JANS, R. (2010). A review of applications of genetic algorithms in lot sizing. *Journal of Intelligent Manufacturing*, 21(4):575–590.
- GUIDE JR, V. D. R. (2000). Production planning and control for remanufacturing : industry practice and research needs. *Journal of operations Management*, 18(4):467–483.

- GUIDE JR, V. D. R. et SRIVASTAVA, R. (1997). An evaluation of order release strategies in a remanufacturing environment. *Computers & operations research*, 24(1):37–47.
- GUPTA, S. et TALEB, K. (1994). Scheduling disassembly. *The International Journal of Production Research*, 32(8):1857–1866.
- HAMMERSLEY, J. et HANDSCOMB, D. (1964). Monte carlo methods, methuen & co. *Ltd., London*, 40.
- ILGIN, M. A. et GUPTA, S. M. (2010). Environmentally conscious manufacturing and product recovery (ecmpro) : A review of the state of the art. *Journal of environmental management*, 91(3):563–591.
- INDERFURTH, K. et LANGELLA, I. M. (2006). Heuristics for solving disassemble-to-order problems with stochastic yields. *OR Spectrum*, 28(1):73–99.
- INDERFURTH, K., VOGELGESANG, S. et LANGELLA, I. M. (2015). How yield process misspecification affects the solution of disassemble-to-order problems. *International Journal of Production Economics*, 169:56–67.
- ISRAEL, G. D. (1992). Determining sample size. *EDIS*.
- JI, X., ZHANG, Z., HUANG, S. et 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. et 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. et LEE, D.-H. (2018). Multi-period disassembly levelling and lot-sizing for multiple product types with parts commonality. *Proceedings of the Institution of Mechanical Engineers, Part B : Journal of Engineering Manufacture*, 232(5):867–878.
- KIM, H.-J., LEE, D.-H. et 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. et 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. et 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. et ZÜST, R. (2003). Disassembly scheduling with multiple product types. *CIRP Annals*, 52(1):403–406.
- KIM, H.-J. et XIROUCHAKIS, P. (2010). Capacitated disassembly scheduling with random demand. *International Journal of Production Research*, 48(23):7177–7194.
- LANGELLA, I. M. (2007). Heuristics for demand-driven disassembly planning. *Computers & Operations Research*, 34(2):552–577.
- LEE, D.-H., KIM, H., CHOI, G. et XIROUCHAKIS, P. (2004). Disassembly scheduling : integer programming models. *Proceedings of the Institution of Mechanical Engineers, Part B : Journal of Engineering Manufacture*, 218(10):1357–1372.
- LEE, D.-H., XIROUCHAKIS, P. et ZUST, R. (2002). Disassembly scheduling with capacity constraints. *CIRP Annals*, 51(1):387–390.

- LIU, K. et 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. et XIROUCHAKIS, P. (2001). Disassembly scheduling with parts commonality using petri nets with timestamps. *Fundamenta Informaticae*, 47(3-4):295–306.
- PRAKASH, P., CEGLAREK, D. et TIWARI, M. (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. et VU, D. Q. (2020). A multi-stage stochastic integer programming approach for a multi-echelon lot-sizing problem with returns and lost sales. *Computers & Operations Research*, 116:104865.
- RUBINSTEIN, R. Y. (2006). Simulation and the monte carlo method, 1981. *New York, Wiley*.
- SLAMA, I., BEN-AMMAR, O., DOLGUI, A. et MASMOUDI, F. (2020). New mixed integer approach to solve a multi-level capacitated disassembly lot-sizing problem with defective items and backlogging. *Journal of Manufacturing Systems*, 56:50–57.
- SLAMA, I., BEN-AMMAR, O., MASMOUDI, F. et DOLGUI, A. (2019). Disassembly scheduling problem : literature review and future research directions. *IFAC-PapersOnLine*, 52(13):601–606.
- SUZANNE, E., ABSI, N. et BORODIN, V. (2020). Towards circular economy in production planning : Challenges and opportunities. *European Journal of Operational Research*.
- TALEB, K. N., GUPTA, S. M. et BRENNAN, L. (1997). Disassembly of complex product structures with parts and materials commonality. *Production Planning & Control*, 8(3):255–269.
- ULLERICH, C. et BUSCHER, U. (2013). Flexible disassembly planning considering product conditions. *International Journal of Production Research*, 51(20):6209–6228.
- WANG, H.-F. et 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.