Sample Average Approximation for Multi-Vehicle Collection-Disassembly Problem under Uncertainty

Muhammad Khoirul Khakim Habibi, Olga Battaïa, Van-Dat Cung, Alexandre Dolgui, Manoj Kumar Tiwari

To appear in the *International Journal of Production Research*
Vol. 00, No. 00, 00 Month 20XX, 1–20

Sample Average Approximation for Multi-Vehicle Collection-Disassembly Problem under Uncertainty

Muhammad Khoirul Khakim Habibi[a*], Olga Battaïa[b], Van-Dat Cung[c,d], Alexandre Dolgui[e] and Manoj Kumar Tiwari[f]

[a]*École Nationale Supérieure des Mines de Saint–Étienne, Lab. LIMOS, CNRS UMR6158, 42023 Saint–Étienne Cedex 2, France;*
[b]*ISAE-SUPAERO, Université de Toulouse, 10 Avenue Edouard Belin, 31400 Toulouse, France;*
[c]*Univ. Grenoble Alpes, G-SCOP, F-38000 Grenoble, France;*
[d]*CNRS, G-SCOP, F-38000 Grenoble, France;*
[e]*IMT Atlantique, LS2N, CNRS, La Chantrerie, 4, rue Alfred Kastler - B.P. 20722, F-44307 NANTES Cedex 3, France;*
[f]*Department of Industrial and Systems Engineering, Indian Institute of Technology Kharagpur, Kharagpur 721 302, West Bengal, India*

(*v5.0 released May 2017*)

The implementation of the circular economy is increasingly supported by many governments. It is performed by integrating the activities of reverse supply chain into those of forward supply chain. However, many companies that traditionally focus on the activities of forward supply chain have decided to collaborate with third-party reverse logistics providers to manage the reverse supply chain. This collaboration motivates the work presented in this paper to propose better decisions for decision makers in the providers under the fact that integrating decisions of the collection of End-of-Life products and their disassembly process proposes a reverse supply chain with better performance. In this paper, an integrated problem concerning those decisions is presented and formalised. It also deals with the uncertainty of the quality and the quantity of products as well as the demands of the associated components. Two approximate methods are developed to provide the solutions.

Keywords: Collection; Disassembly; Reverse Supply Chain; Stochastic Programming; Two-Phase Iterative Heuristic; Sample Average Approximation

1. Introduction

Nowadays, the evolution of the economic framework employed by the companies from the linear economy towards the circular economy is increasingly supported by many governments. The linear economy is characterised by "take-make-dispose" pattern where the raw materials are transformed into the final products in order to fulfil the demands of the clients and are disposed once they reach the end of life cycle. Based on the report of World Economic Forum (Forum 2014), the linear economy is arriving its limits in due to (i) the growth of resource prices and supply disruptions, (ii) the price volatility of metals, foods and non-agriculture outputs, (iii) the difficulty of creating sufficient competitive advantage or differentiation, (iv) the unpredicted consequences of the improvement of energy and resource efficiency, (v) the deceleration of agriculture productivity followed, (vi) the increasing risk of global supply chain's supply security and safety and (vii) the difficulties of getting virgin resources (water, land and atmosphere). Therefore, the implementation of the circular

*Corresponding author. Email: muhammad.habibi@emse.fr

economy is encouraged due to its advantages in terms of financial, social and environmental factors. This implementation is carried out by incorporating the activities of reverse supply chain (RSC) dealing with End-of-Life (EOL) products i.e. reusing, repairing, refurbishing and recycling, to those of forward supply chain in order to form a closed-loop (circular) supply chain. Xerox, Hewlett-Packard and Phillips are among success story regarding such implementation (Kumar and Putnam 2008; Pishvaee, Farahani, and Dullaert 2010; Alumur et al. 2012; Forum 2014).

However, many companies that traditionally focus on the activities of forward supply chain have decided to collaborate with other specialised companies to manage RSC such as La Fédération ENVIE (France) since it requires new facilities and activities that are typically costly and manual labour intensive. This fact motivates this work to propose better decisions for decision makers in such third-party reverse logistics providers (3PRLPs). In detail, this work has been put in place in the particular interest of the management of EOL products from the points of collection until the point of re-manufacturers and/or recyclers.

Habibi et al. (2017a,b) proved that integrating the decisions of the collection and disassembly processes of RSC leads to optimise and enhance the performance of the RSC in terms of total cost and the demand satisfaction. However, this work focuses on the deterministic case in which that the quality and the quantity of the EOL products as well as the demands of their components are well defined and only single vehicle is considered.

Table 1.: Comparison of Habibi et al. (2017a,b) and this paper

CHARACTERISTIC	**Habibi et al. (2017a,b)**	**This Paper**
Uncertainty	No	Yes
Parameters affected by uncertainty	No	Quantity of products returned Quantity of components Demand of component
No. of Vehicle	Single	Multiple

Reverse Supply Chain (RSC) needs to consider the uncertainty notably when it deals with End-of-Life (EOL) products. Based on McGovern and Gupta (2011), such products are often returned with imperfect or modified condition such as missing parts, components are replaced with higher quality ones etc. The quantity returned and the demand of component to sell are also highly uncertain.

Employing stochastic programming is an effective way to deal with such uncertainties. Managers of an RSC dealing with EOL products may refer to our work. They are able to optimize the performance of an RSC by minimizing the total cost containing the expected cost emerged from the uncertainties.

In this paper, we extend the works of Habibi et al. (2017a,b) in order to approach the field reality by taking into account the uncertainty of the quality and the quality of EOL products as well as the demands of their components. The quality of product is assumed equivalent to the quantity of its components after it is returned to collection centres. We also take into account the case of multi-vehicle since 3PRLPs often posses more than one vehicle. Figure 1 depicts the contribution of this paper compared to the previous work.

2. Literature Review

The problem in Habibi et al. (2017a,b), called Collection-Disassembly Problem (CDP), is a version of Production-Distribution Problem (PDP) or Production-Routing Problem (PRP) in RSC. It integrates decisions of two well-known and hard combinatorial problems i.e. vehicle routing and lot-sizing, to deal with the collection of EOL products and their disassembly process, respectively.

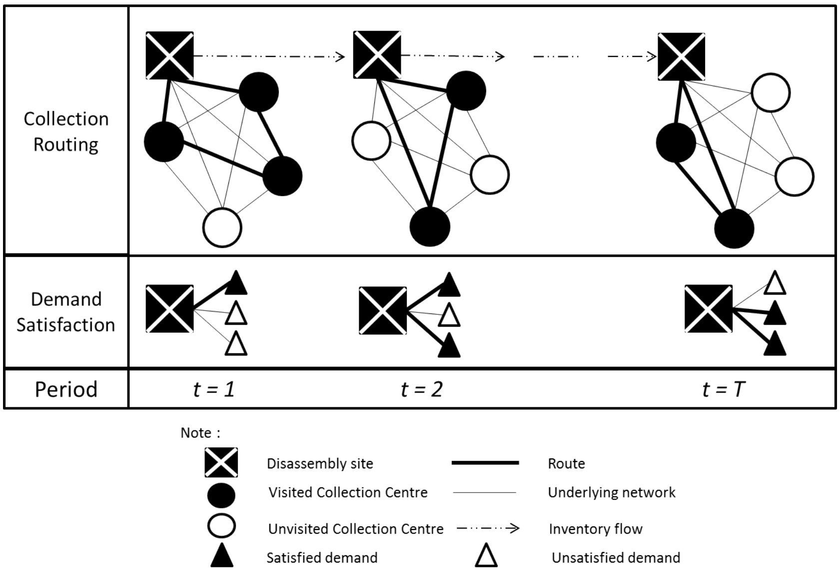

Figure 1.: Network Representations of Collection-Disassembly Problem Habibi et al. (2017a)

PDP differs from Newsvendor Problem. In their basic formulations, PDP concerns with multiple-period plan whilst Newsvendor Problem considers single period. However, we are aware that Newsvendor Problem is advanced in multi-period. Newsvendor Problem considers no vehicle routing whilst it is an important part of PDP. Compared to routing problems, PDP takes into account decisions on production process. As an adaptation of PDP into RSC, the routing part of CDP concerns about the collection of End-of-Life products and the lot-sizing part lies in their disassembly process. In this work, we assume that some parameters are under uncertainty and more than one vehicle are available to use.

To well position our work in the existing literature, relevant papers in PDP are reviewed. The papers propose models and approximate methods particularly in dealing with large size instances when the commercial solver is not able to provide optimal solutions in acceptable CPU times.

The majority of PDP works focuses on both production and distribution aspects by incorporating the decisions of production and routing aspects in tactical level decision. Based on the existing literature of PDP that mostly deals with continuous products, the objective function minimises the total cost of production, inventory and routing, simultaneously, by respecting the demands of retailers, their inventory limits, the production facility's capacity and its inventory limit. As CDP, PDP is also a combination of two well-known and hard combinatorial problems i.e. lot-sizing problem and VRP, to deal with forward supply chain.

Based on Boudia, Louly, and Prins (2007); Bard and Nananukul (2009, 2010); Armentano, a.L. Shiguemoto, and Lø kketangen (2011), the decisions of PDP throughout the planning horizon consist of:

(1) when and how much products to produce
(2) when to visit and how much to deliver to retailers as well as the routing
(3) inventory level for each retailer and the depot

There are four existing models as the main references of PDP with multi-vehicle under Maximum Level policy. First, the formulation of Boudia, Louly, and Prins (2007) which is completed with vehicle index. Second, the formulation proposed in Bard and Nananukul (2009, 2010) has no index regarding the vehicle. Third, the formulation of Armentano, a.L. Shiguemoto, and Lø kketangen (2011) deals with the case of multi-products where the available vehicles are indexed. Fourth, the formulation of Adulyasak, Cordeau, and Jans (2015b) deals with PDP under the uncertainty of demands of retailers. Adulyasak, Cordeau, and Jans (2015a) and Díaz-Madroñero, Peidro, and Mula (2015) provide extensive reviews on PDP. Table 2 contains works on PDP highlighting on its variants as well as solving methods proposed.

Table 2.: Variants and Solving Methods of Production-Distribution Problem

Problem	Authors	Solving Methods
PDP	Chandra and Fisher (1994)	Decomposition
	Fumero and Vercellis (1999)	Lagrangian Relaxation
	Buer, Woodruff, and Olson (1999)	Genetic Algorithm
	Bertazzi, Paletta, and Speranza (2005)	Decomposition
	Boudia, Louly, and Prins (2007)	GRASP
	Chen, Hsueh, and Chang (2009)	Decomposition
	Çetinkaya et al. (2009)	Decomposition
	Boudia and Prins (2009)	Memetic Algorithm
	Bard and Nananukul (2009, 2010)	Brand & Price
	Solyalı and Süral (2009)	Relaxation based Heuristic
	Shiguemoto and Armentano (2010)	Tabu Search
	Archetti et al. (2011)	Branch-and-Cut
		Mathematical Programming based Heuristic
	Armentano, a.L. Shiguemoto, and Lø kketangen (2011)	Tabu Search
	Calvete, Galé, and Oliveros (2011)	Ant Colony
	Adulyasak, Cordeau, and Jans (2012)	ALNS
	Absi et al. (2014)	Two-Phase Iterative Heuristics
	Adulyasak, Cordeau, and Jans (2014)	Branch-and-Cut
	Russell (2017)	Mathematical Programming Heuristics
	Solyalı and Süral (2017)	Multi-Phase Heuristic
	Qiu et al. (2018c)	Variable Neighborhood Search
PDP with Perishable Product	Amorim et al. (2013)	Branch-and-Bound
PDP under Demand Uncertainty	Adulyasak, Cordeau, and Jans (2015a)	Benders Decomposition
PDP with Backordering	Brahimi and Aouam (2016)	Relax-and-Fix Heuristic and Local Search
PDP with Carbon Cap-and-Trade	Qiu, Qiao, and Pardalos (2017)	Branch-and-Price Heuristic
PDP in Close Loop-Supply Chain	Qiu et al. (2018a)	Branch-and-Cut guided Search
PDP with Pollution Consideration	Kumar et al. (2015)	Self-Learning PSO
PDP with Startup Cost	Qiu et al. (2018b)	Branch-and-Cut
Rich PDP	Miranda et al. (2018)	Decomposition Heuristic
PDP in Reverse Supply Chain	Habibi et al. (2017a)	Branch-and-Bound
	Habibi et al. (2017b)	Two-Phase Iterative Heuristics
PDP in Reverse Supply Chain under Uncertainty	This Paper	SAA and Two-Phase Iterative Heuristics

GRASP stands for Greedy Randomized Adaptive Search Procedure
ALNS stands for Adaptive Large Neighborhood Search
PSO stands for Particle Swarm Optimization
SAA stands for Sampling Average Approximation

To the best of our knowledge, there is no work attempting to formalise the integration of decisions regarding the collection of EOL products and their disassembly process by taking into account the uncertainty of their quality and the quantity as well as the demands of their components. Also, there is no work proposing the case of multi-vehicle in such a problem. Therefore, a formulation filling this research gap is presented. Two methods are developed to provide the solutions to this problem.

3. Problem Formulation

This problem considers that a single site performs a disassembly process for treating a single type of EOL products available at dispersed collection centres. Some homogeneous vehicles with fixed capacity are available for collecting the EOL products.

The products' nomenclature is known and identical. Each product has several components where each component has uncertain quantity. The collected products are disassembled in the site in order to release the components requested. The site has a fixed capacity corresponding to its cycle time. A penalty cost is occured once the component demand is unmet. The demand of component is uncertain and assumed following some known distribution. The problem contains multi-period due to the presence of an inventory to store the collected EOL products. There is no salvage value or disposal cost for any leftover components. Following its characteristics, the problem is called as *Stochastic Multi-Vehicle Collection-Disassembly Problem* and abbreviated as *SMCDP*. In this work, the quality of product is assumed equivalent to the quantity of its components after it is returned to collection centers. We denote the quantity of each component a of the product at period

t under scenario ω as n^{ω}_{at}.

Parameters:

$\mathcal{A}$	set of component: $a = \{1, 2, \cdots, \lvert\mathcal{A}\rvert\}$
$\mathcal{N}$	set of nodes: $i, j = \{1, 2, \cdots, \lvert\mathcal{N}\rvert\}$ where 1 is the depot
$\mathcal{N}_c$	set of collection centres: $i, j = \{2, \cdots, \lvert\mathcal{N}\rvert\}$
Ω	set of finite scenario: $\omega = \{1, 2, \cdots, \lvert\Omega\rvert\}$
$\mathcal{T}$	planning horizon: $t = \{1, 2, \cdots, \lvert\mathcal{T}\rvert\}$
$\mathcal{K}$	set of vehicles: $k = 1, 2, \cdot \lvert\mathcal{K}\rvert$
ρ^{ω}	probability of scenario ω
n^{ω}_{at}	quantity of component a in the product at period t under scenario ω
S^{ω}_{it}	quantity of products available at collection centre i at period t under scenario ω
q^{ω}_{at}	demand of component a at period t under scenarion ω
Q	vehicle capacity
$InvCap$	inventory capacity
$DisCap$	disassembly line capacity imposed from its cycle time
CF	fixed vehicle dispatch cost
c_{ij}	mileage cost from node i to j
CD	unit disassembly cost
CH	unit holding cost
CP_a	unit penalty cost of component a.

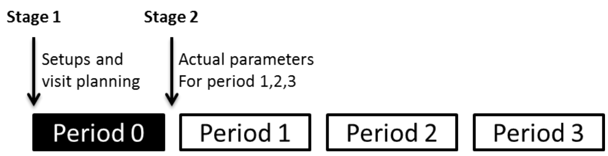

Figure 2.: Two-Stage Stochastic Problem

The problem is formalised as a two-stage stochastic programming. In this problem, a scenario ω in period t is a realisation of S^{ω}_{it}, n^{ω}_{at} and q^{ω}_{at} into a fixed value coming after the planning stage as depicted in Figure 2.

The first-stage decisions correspond to the planning of the routing of vehicle $k, \in K$ for each period as follows:

z_t	number of vehicles dispatched at period t
x^{k}_{ijt}	1 if node j visited immediately after i by vehicle k at period t. 0 otherwise.

Consequently, the second-stage decisions correspond to the load of vehicles and the disassembly decisions. These decisions will be taken after the realisation of the parameter under uncertainty. The decisions are:

$y^{k\omega}_{it}$	load of vehicle k after visiting node i at period t in scenario ω
I^{ω}_{t}	inventory level of EOL products at period t in scenario ω
P^{ω}_{t}	quantity of EOL products disassembled at period t in scenario ω
SO^{ω}_{at}	unmet demands of component a at period t in scenario ω

Stochastic Integer Linear Programming of SMCDP:

$$Min \sum_{t\in\mathcal{T}} (CF \cdot z_t + \sum_{k\in\mathcal{K}} \sum_{i,j\in\mathcal{N}} c_{ij} \cdot x^{k}_{ijt} + \sum_{\omega\in\Omega} \rho^{\omega} (CH \cdot I^{\omega}_{t} + CD \cdot P^{\omega}_{t} + \sum_{a\in\mathcal{A}} CP_a \cdot SO^{\omega}_{at})) \quad (1)$$

subject to:

$$\sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{N},i\neq j} x_{ijt}^{k} \leq 1 \qquad \forall i\in\mathcal{N}_c, \forall t\in\mathcal{T} \tag{2}$$

$$\sum_{i\in\mathcal{N}_c} x_{1it}^{k} \leq 1 \qquad \forall k\in\mathcal{K}, \forall t\in\mathcal{T} \tag{3}$$

$$\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{N}_c} x_{1it}^{k} \leq z_t \qquad \forall t\in\mathcal{T} \tag{4}$$

$$\sum_{i\in\mathcal{N},i\neq v} x_{ivt}^{k} = \sum_{j\in\mathcal{N},j\neq v} x_{vjt}^{k} \qquad \forall v\in\mathcal{N}, \forall k\in\mathcal{K}, \forall t\in\mathcal{T} \tag{5}$$

$$y_{it}^{k\omega} + (Q - S_{it}^{\omega})\cdot x_{1it}^{k} \leq Q \qquad \forall i\in\mathcal{N}_c, \forall k\in\mathcal{K}, \forall t\in\mathcal{T}, \forall\omega\in\Omega \tag{6}$$

$$y_{it}^{k\omega} - y_{jt}^{k\omega} + Q\cdot x_{ijt}^{k} + (Q - S_{jt}^{\omega} - S_{it}^{\omega})\cdot x_{jit}^{k} \leq Q - S_{jt}^{\omega} \quad i\neq j, \forall i,j\in\mathcal{N}_c, \forall k\in\mathcal{K}, \forall t\in\mathcal{T}, \forall\omega\in\Omega \tag{7}$$

$$I_t^{\omega} = I_{t-1}^{\omega} + \sum_{k\in K}\sum_{i,j\in\mathcal{N},i\neq j} S_{it}^{\omega}\cdot x_{ijt}^{k} - P_t^{\omega} \qquad \forall t\in\mathcal{T}, \forall\omega\in\Omega \tag{8}$$

$$n_{at}^{\omega}\cdot P_t^{\omega} + SO_{at}^{\omega} \geq q_{at}^{\omega} \qquad \forall a\in\mathcal{A}, \forall t\in\mathcal{T}, \forall\omega\in\Omega \tag{9}$$

$$\sum_{j\in\mathcal{N},i\neq j} S_{it}^{\omega}\cdot x_{ijt}^{k} \leq y_{it}^{k\omega} \leq \sum_{j\in\mathcal{N},i\neq j} Q\cdot x_{ijt}^{k} \qquad \forall i\in\mathcal{N}, \forall k\in\mathcal{K}, \forall t\in\mathcal{T}, \forall\omega\in\Omega \tag{10}$$

$$I_t^{\omega} \leq InvCap \qquad \forall t\in\mathcal{T}, \forall\omega\in\Omega \tag{11}$$

$$P_t^{\omega} \leq DisCap \qquad \forall t\in\mathcal{T}, \forall\omega\in\Omega \tag{12}$$

$$z_t \leq |\mathcal{K}| \qquad \forall t\in\mathcal{T} \tag{13}$$

$$x_{ijt}^{k} \in \{0,1\} \qquad \forall i,j\in\mathcal{N}, \forall k\in\mathcal{K}, \forall t\in\mathcal{T} \tag{14}$$

$$z_t, y_{it}^{k\omega}, SO_{at}^{\omega}, I_t^{\omega}, P_t^{\omega} \in \mathbb{Z}^{+} \qquad \forall a\in\mathcal{A}, \forall i\in\mathcal{N}, \forall k\in\mathcal{K}, \forall t\in\mathcal{T}, \forall\omega\in\Omega. \tag{15}$$

The objective function (1) minimises the total cost of the first-stage decision and the expected cost corresponding to the second-stage decisions. The first and second terms correspond to the dispatch and mileage vehicle costs. The latter terms consist of the expected costs of inventory, disassembly and penalty.

Constraints (2) state that a collection centre is visited at most once by any vehicle at period t. Constraints (3) ensure that vehicle k is dispatched at most once at period t. Constraints (4) determine number of vehicles dispatched at period t. The flows of visiting and leaving a node are conserved by constraints (5). The load of vehicle k after visiting a collection centre i in which it is the first node visited is ensured by constraints (6). Constraint (7) eliminate any subtour. Constraints (8) are the inventory balance of disassembly site. Constraints (9) impose the demand fulfilment. Constraints (10), (11), (12) and (13) are the limitation of load of vehicle, inventory level, disassembly and maximum number of vehicles, respectively. Constraints (14) and (15) define the decision variables.

4. Solution Methods

As far as our knowledge, Two-Phase Iterative Heuristics (Absi et al. 2014) provides high quality solutions in PDP. We also experience that this method with the type of Iterative Method with

Multi-Travelling Salesman Problem (IM-MultiTSP) performs very good in CDP and is also flexible enough to be adapted. Therefore, it is implemented in SMCDP in the hope that it also proposes good performance. Apart from that, we also propose an enhanced version of this method since this enhancements lead to better performance than the original one.

An algorithmic framework is implemented combined with both methods to provide statistical lower and upper bounds since the proposed formulation is stochastic discrete optimisation problem with finite number of scenario.

4.1 *Two-Phase Iterative Heuristic*

In this part, the implementation of this method is demonstrated into PDP and SMCDP, consecutively.

4.1.1 In PDP

This method is originally proposed in Absi et al. (2014) for dealing with PDP with multi-vehicle and single type of product. It decomposes the problem into two subproblems and solved them iteratively. The two subproblems are the lot-sizing subproblem with approximate visiting costs and the routing subproblem.

The lot-sizing subproblem with approximate visiting costs, also called as the first phase, deals with the decisions of when and how much products to produce, when to visit retailers and how much products to retailers. Consecutively, this phase provides the set of retailers served in each period. Also, the vehicle capacity is already taken into account in this subproblem. Accordingly, the second phase (the routing subproblem) aims to construct the route of vehicle dispatched for each period corresponding to the set of retailers served.

The approximate visiting costs of a retailer are initialised by multiplying the go-return running costs and the distance between the retailer and the production facility. For those who are served in a particular period, a particular procedure is used to update their corresponding approximate visiting costs (see next part).

A diversification mechanism of the approximate visiting costs are required to permit the method exploring the unvisited solution space. It is simply done by multiplying the current value of the costs by the number of retailers visited throughout the planning horizon plus one. One is to avoid zero multiplication when no retailer is visited. The method is provided in Algorithm 1.

Algorithm 1: Two-Phase Iterative Heuristic for PDP

```
solution ← ∅
Initialise the approximate visiting costs for all retailers and vehicles
while a stopping criteria is not met do
    while a stopping criteria is not met do
        Solve the lot-sizing subproblem with approximate visiting costs
        Get the set of retailers served
        Construct the routes to visit the served retailers in the routing subproblem
        Update solution (if necessary) and approximate visiting costs
    end
    Diversify the approximate visiting costs
end
```

4.1.2 In SMCDP

This part consist of the adaptation of Two-Phase Iterative Heuristic for solving SMCDP. Some modifications are required due to the uncertainty of several parameters.

The problem is decomposed into two subproblems: (i) *Stochastic Multi-Vehicle Reverse Lot-Sizing Problem with Approximate Visiting Costs* (SMRLP-AVC) and (ii) *Routing Problem.*

4.1.2.1 SMRLP-AVC. This problem is a simplification of SMCDP by replacing the travelling cost with so-called approximated visiting cost. It determines how many vehicles to use z_t, which collection centre to visit, how many EOL products to store into the inventory I_t^ω, how many EOL products to disassemble P_t^ω and how many penalty occurred SO_{at}^ω. Instead of using c_{ij}, it uses the approximate visiting costs denoted as SC_{it}^k. These costs are initialised using $c_{0i} + c_{i0}$ and updated throughout the method.

The decision variables of SMRLP-AVC are described as follows:

γ_{it}^k $\begin{cases} 1 & \text{if node } i \text{ is visited by vehicle } k \text{ at period } t \\ 0 & \text{otherwise} \end{cases}$

β_t^k $\begin{cases} 1 & \text{if vehicle } k \text{ visits any collection centre at period } t \\ 0 & \text{otherwise} \end{cases}$

$r_{it}^{k\omega}$ quantity of EOL products collected from node i by vehicle k at period t under scenario ω

4.1.2.2 Formulation of SMRLP-AVC.

$$\textbf{Min} \sum_{t\in\mathcal{T}} \left(CF \cdot z_t + \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{N}_c} SC_{it}^k \cdot \gamma_{it}^k + \sum_{\omega\in\Omega} \rho^\omega \left(CH \cdot I_t^\omega + CD \cdot P_t^\omega + \sum_{a\in\mathcal{A}} CP_a \cdot SO_{at}^\omega \right) \right) \tag{16}$$

Subject to:

$$(9), (11), (12)$$

$$I_t^\omega = I_{t-1}^\omega + \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{N}_c} r_{it}^{k\omega} - P_t^\omega \qquad \forall t \in \mathcal{T}, \forall \omega \in \Omega \tag{17}$$

$$r_{it}^{k\omega} = S_{it}^\omega \cdot \gamma_{it}^k \qquad \forall i \in \mathcal{N}_c, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}, \forall \omega \in \Omega \tag{18}$$

$$\sum_{i\in\mathcal{N}_c} r_{it}^{k\omega} \leq min\left\{ Q; max_a\{\frac{\sum_{t'=t}^{T} q_{at'}^\omega}{n_a}\} \right\} \qquad \forall k \in \mathcal{K}, \forall t \in \mathcal{T}, \forall \omega \in \Omega \tag{19}$$

$$\sum_{k\in\mathcal{K}} \gamma_{it}^k \leq 1 \qquad \forall i \in \mathcal{N}_c, \forall t \in \mathcal{T} \tag{20}$$

$$\sum_{i\in\mathcal{N}_c} \gamma_{it}^k \leq |\mathcal{N}_c| \cdot \beta_t^k \qquad \forall k \in \mathcal{K}, \forall t \in \mathcal{T} \tag{21}$$

$$\beta_t^k \leq \sum_{i\in\mathcal{N}_c} \gamma_{it}^k \qquad \forall k \in \mathcal{K}, \forall t \in \mathcal{T} \tag{22}$$

$$\sum_{k\in\mathcal{K}} \beta_t^k = z_t \qquad \forall t \in \mathcal{T} \tag{23}$$

$$z_t \leq |\mathcal{K}| \qquad \forall t \in \mathcal{T} \tag{24}$$

$$\gamma_{it}^k, \beta_t^k \in \{0, 1\} \qquad \forall i \in \mathcal{N}_c, \forall k \in \mathcal{K}, \forall t \in \mathcal{T} \tag{25}$$

$$SO_{at}^\omega, I_t^\omega, P_t^\omega, r_{it}^{k\omega}, z_t \in \mathbb{Z}^+ \qquad \forall a \in \mathcal{A}, \forall i \in \mathcal{N}_c, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}, \forall \omega \in \Omega. \tag{26}$$

The objective function (16) is to minimise the total cost consisting of total fixed cost of vehicles deployed, total approximate cost and the expected cost of second-stage decisions of inventory, quantity of EOL products disassembled and unmet demand.

Constraints (17) impose the inventory balance of EOL products. Constraints (18) state all EOL products belong to collection centre i have to be picked up once it is visited by any vehicle. Constraints (19) is the maximum limit of $r_{it}^{k\omega}$. Constraints (20) guarantee that a collection centre is visited at most once by any vehicle for each period. Constraints (21) state that β_t^k is equal to 1 if vehicle k visited at least one collection centre in period t. Otherwise, β_t^k is equal to 0 as imposed by constraints (22). Constraints (23) and (24) state that number of vehicles used in each period is limited to the number of available vehicles. The nature of the decision variables on both stages are imposed in constraints (25) and (26).

Based on the values of γ_{it}^k obtained by solving SMRLP-AVC, the route of each vehicle is constructed. If necessary, the decision values and SC_{it}^k are updated based on the objective value. Apart from its initial values, the diversification mechanism of SC_{it}^k is employed in order to move to the other solution space. The algorithm 2 provides the implementation of the method to SMCDP.

Algorithm 2: Two-Phase Iterative Heuristic for SMCDP

```
solution ← ∅ ;
Initialise SC_it^k, ∀i ∈ N_c, k ∈ K, t ∈ T
while a stopping criteria is not met do
    while a stopping criteria is not met do
        while a stopping criteria is not met do
            Solve SMRLP-AVC and get γ_it^k, ∀i ∈ N_c, ∀k ∈ K, ∀t ∈ T
            Solve Routing Problem
            Update solution (if necessary) and SC_it^k
        end
        Diversify SC_it^k
    end
    Multi-start procedure: SC_it^k = ρ_it · (c_0i + c_i0), ∀i ∈ N_c, k ∈ K, t ∈ T
end
```

Algorithm 3: Update of approximate visiting costs SC_{it}^k

```
forall t ∈ T do
    forall k ∈ K do
        forall i ∈ N_c do
            if i ∈ route_t^k then
                SC_it^k ← c_{i^- i} + c_{i i^+} − c_{i^- i^+}
            else
                SC_it^k ← Δ_it
            end
        end
    end
end
```

4.1.2.3 Routing Problem. The first phase provides the set of nodes served in each period. Thus, the routing problem becomes multi-TSP. To construct the route, we use the Lin-Kernighan Heuristic Lin and Kernighan (1973) as the the state-of-the-art heuristic of solving TSP.

4.1.3 *Adaptive Two-Phase Iterative Heuristic*

In this part, the enhancement of Two-Phase Iterative Heuristic for SMCDP is described. It is proposed since it leads to faster computational time with good solutions for CDP.

In Algorithm 2, one notes that the problem is decomposed into SMRLP-AVC and routing problem. The solutions and SC_{it}^{k} are updated if the corresponding fitness value is better than previous one. In this enhancement, this step is denoted as the first step. The enhancement expands the method by putting the second and third additional steps in order to propose a better solution.

The second step introduces SMRLP-AVC II in order to provide the solution of SMRLP-AVC serving less periods by introducing the parameter Z indicating number of periods served in SMRLP-AVC. The variable α_t is equal to 1 if period t is served. Otherwise, it is 0. In this step, the approximate visiting costs SC_{it}^{k} of the second step are identical to the first step. SMRLP-AVC II is formalised as follows:

4.1.3.1 *SMRLP-AVC II.*

$$\textbf{Min } 16$$

Subject to:

$$(9), (11), (12), (17) - (26)$$

$$z_t \leq |\mathcal{K}| \cdot \alpha_t \qquad \forall t \in \mathcal{T} \tag{27}$$

$$\sum_{t \in \mathcal{T}} \alpha_t \leq Z - 1 \tag{28}$$

$$\alpha_t \in \mathbb{Z}^{+} \qquad \forall t \in \mathcal{T}. \tag{29}$$

Based on our experiences in the deterministic CDP, the second step of enhancements indeed provides better optimality gaps but longer CPU times than Two-Phase Iterative Heuristic. Therefore, an adaptive procedure is required to deal with this issue as follows.

This procedure is carried out by introducing the probability of using the second step denoted as *prob*. These value is halved once the step has no contribution to the solution by comparing it with a random values *rand*. This method is depicted in Algorithm 4.

The Algorithm 4 provides this enhancement as well as the adaptive procedure. The Algorithm 3 is also used in this enhanced method to update SC_{it}^{k}.

4.2 *Sample Average Approximation*

Since SMCDP is stochastic discrete optimization problem, we adapt the Sample Average Approximation (SAA). This Monte Carlo-based sampling method is to tackle a problem having very large number of scenario denoted as Ω', which is intractable, by solving the problem with a set of smaller and tractable scenario Ω where $|\Omega| \ll |\Omega'|$ (Adulyasak, Cordeau, and Jans 2015b; Kleywegt, Shapiro, and Homem-de Mello 2002; Ghilas, Demir, and Woensel 2016).

The following is the procedure of SAA applied to our SMCDP:

1. Set replication $\mathcal{M}$ and generate scenario Ω as well as very large scenario Ω' independently. The probability of each scenario ω associated with $|\Omega|$ is $\rho^{\omega} = \dfrac{1}{\Omega}$.
2. For $s = 1 \rightarrow \mathcal{M}$, do :
   2.1. Solve SMCDP. Store the objective value Z_{Ω}^{s}, the vectors of the first stage solutions $(\mathbf{z}_{\Omega}^{s}, \mathbf{x}_{\Omega}^{s})$ and the vectors of the second stage solutions $(\mathbf{I}_{\Omega}^{s}, \mathbf{P}_{\Omega}^{s}, \mathbf{SO}_{\Omega}^{s})$. The average and

Algorithm 4: Adaptive Two-Phase Iterative Heuristic for SMCDP

```
solution ← ∅ ;
Initialise SC^k_it, ∀i ∈ N_c, k ∈ K, t ∈ T
while a stopping criterion is not met do
    prob = 1
    while a stopping criterion is not met do
        while a stopping criterion is not met do
            FIRST STEP ;
              • Solve SMRLP-AVC and get γ^k_it, ∀i ∈ N_c, ∀k ∈ K, ∀t ∈ T
              • Solve Routing Problem
              • Update solution (if necessary) and SC^k_it

            SECOND STEP
            Generate rand ;
            if rand ≤ prob then

                  • Solve SMRLP-AVC II and get γ^k_it, ∀i ∈ N_c, ∀k ∈ K, ∀t ∈ T
                  • Solve Routing Problem
                  • Update solution (if necessary) and SC^k_it

                if solution is not updated then
                    prob = prob/2
                end
            end
        end
        Diversify SC^k_it
    end
    Multi-start procedure: SC^k_it = ρ_it · (c_0i + c_i0), ∀i ∈ N_c, k ∈ K, t ∈ T
end
```

the variance of the objective value after s replication denoted as v^s_Ω and ${\sigma^s_\Omega}^2$ are obtained as follows:

$$v^s_\Omega = \frac{1}{s}\sum_{i=1}^{s} Z^i_\Omega$$

$$ {\sigma^s_\Omega}^2 = \frac{1}{s\cdot(s-1)}\sum_{i=1}^{s}(Z^i_\Omega - v^s_\Omega)^2$$

2.2. To obtain the second stage solutions $\mathbf{I}^s_{\Omega'}$, $\mathbf{P}^s_{\Omega'}$ and $\mathbf{SO}^s_{\Omega'}$ of a very large scenario Ω', use the best first stage solution after replication s denoted as $\hat{\mathbf{z}}^s_\Omega$ and $\hat{\mathbf{x}}^s_\Omega$. The corresponding objective value (upper bound) is denoted as $v_{\Omega'}(\hat{Z}^s)$ and its variance is obtained as follows:

$$\sigma^2_{\Omega'} = \frac{1}{\Omega'\cdot(\Omega'-1)}\sum_{\omega=1}^{\Omega'}(G_\omega - v_{\Omega'}(\hat{Z}^s))^2$$

where,

$$G_\omega = \sum_{t \in T} \Big\{ CF \cdot \hat{z}_t + \sum_{k \in K} \sum_{i,j \in N} c_{ij} \cdot \hat{x}^k_{ijt} + CH \cdot \tilde{I}^\omega_t + CD \cdot \tilde{P}^\omega_t + \sum_{a \in A} CP_a \cdot \widetilde{SO}^\omega_{at} \Big\}$$

Note that $\tilde{I}^\omega_t$, $\tilde{P}^\omega_t$ and $\widetilde{SO}^\omega_{at}$ correspond to the second stage solution for scenario $\omega \in \Omega'$.

2.3. Calculate the SAA gap ε and its variance σ^2_ε as follows:

$$\varepsilon = v_{\Omega'}(\hat{Z}^s) - v^s_\Omega$$
$$\sigma^2_\varepsilon = \sigma^2_{\Omega'} + {\sigma^s_\Omega}^2$$

3. Return $\hat{\mathbf{z}}^s_\Omega$ and $\hat{\mathbf{x}}^s_\Omega$ as the best solution.

5. Numerical Experiments

5.1 *Experimental Setup*

All formulations and algorithms were implemented in Java using Concert Technology and were solved by IBM CPLEX 12.6 on a PC with processor Intel®Core™i7 CPU 2.9 GHz and 4 Gb RAM under Windows 7 Professional.

The Monte Carlo simulation was used for scenario generation of the parameters associated with uncertainty (S^ω_{it}, n^ω_{at} and q^ω_{at}). They were generated independently by multiplying the corresponding values of deterministic CDP with random value following uniform distribution from 0 to 1.5. The number of vehicles were set to 1, 3 and 5 while the large scenario Ω' were set to 1000. To avoid memory issues, the maximum number of branch nodes of CPLEX for both two SMRLP-AVCs and large scenario problem of SAA is limited to 75000. The methods were tested using instances 49, 61, 73, 85 and 97 of Data Sets of Random 1, Random 2, Cluster 1 and Cluster 2 presented in Habibi et al. (2017b). Their characteristic is provided in Table 3.

Table 3.: The Characteristic of Instances in All Data Sets

Instance	Characteristic				
	$\lvert\mathcal{N}\rvert$	$\lvert\mathcal{T}\rvert$	$\lvert\mathcal{A}\rvert$	Demand	$DisCap$
49	10	10	10	$U(40\% : 60\%) \cdot \boldsymbol{S}$	∞
61	10	5	10	$U(40\% : 60\%) \cdot \boldsymbol{S}$	∞
73	5	25	10	$U(40\% : 60\%) \cdot \boldsymbol{S}$	∞
85	5	10	10	$U(40\% : 60\%) \cdot \boldsymbol{S}$	∞
97	5	10	5	$U(40\% : 60\%) \cdot \boldsymbol{S}$	∞

$U(a : b)$ indicates that the corresponding parameter was generated with uniform distribution with parameter a and b

$\boldsymbol{S}$ is the average of supply of EOL products for all collection centres and all periods.

The three stopping criteria in Algorithms 2 and 4 are as follows:

(1) standard deviation of last ten fitness values, maximum iteration and CPU time are less than 5 %, 100 and 7200 seconds, respectively
(2) maximum number of diversification mechanism is 5
(3) maximum number of multi-start procedure is 5.

5.2 *Results*

For the sake of simplicity, $\mathcal{H}$ and $\mathcal{H}^*$ refer to Two-Phase Iterative Heuristic and Adaptive Two-Phase Iterative Heuristic, respectively. The summary of all solutions obtained is provided in Table 4. The details of all solutions are provided in Tables 7 - 9.

Table 4.: Results of All Data Sets

Data Sets	$\mathcal{H}$			$\mathcal{H}^*$		
	μ	σ	CPU Time	μ	σ	CPU Time
Random 1	539.53	26.90	1782.00	541.04	27.81	2030.58
Random 2	539.71	27.54	2162.05	540.76	27.87	2369.33
Cluster 1	539.27	27.61	1485.28	539.70	26.95	1423.26
Cluster 2	539.16	27.68	1438.43	540.13	26.73	1465.40

μ is average
σ is standard deviation
CPU time is in seconds

According to Table 4, both methods provide solutions with no significant difference in terms of average and standard deviation. In terms of CPU times, $\mathcal{H}$ requires longer time in solving the data set of Random 1 rather than $\mathcal{H}^*$ as shown by Figures 3 and 4.

Based on Figure 3, one notes that both methods are stable to solve the instances although there is a variation of scenario $\mathcal{M}$. However, the increase of the number of available vehicles $\mathcal{K}$ causes longer CPU times for both methods.

In order to further elaborate the results obtained, we conducted a sensitivity analysis on instance 61 of Random 2 data set. The analysis was conducted by varying the distribution of parameters under uncertainty and the multiplier value of increasing the penalty cost CP_a. We tested three types of distribution (Normal, Poisson and Uniform) and three values of penalty multiplication (3, 5 and 10). For normal distribution, the mean and standard deviation of each parameter under uncertainty were set to the corresponding deterministic value in CDP and 50 % of that value, respectively. For Poisson distribution, the only parameter was set to the corresponding deterministic value in CDP. For uniform distribution, it was set equally as mentioned in the previous section. We also define the small and large scenarios of SAA, Ω and Ω', as 100 and 1100, respectively.

In this regard, we add information regarding the two means for evaluating the stochastic solutions proposed by both solving methods called the Expected Value of Perfect Information (EVPI) and the Value of Stochastic Solutions (VSS). According to Birge and Louveaux (2011), EVPI and VSS measure the amount paid by a decision maker in return for a perfect information and is the cost incurred for ignoring the uncertainty, respectively. In practice, EVPI is the difference between the average objective value of solutions obtained for each scenario solved independently and the objective value of stochastic solution. Meanwhile, VSS is the difference between the objective value of stochastic solution and the objective value obtained using the average values of parameters under uncertainty. In our work, we obtain the two values based on SAA's very large set of scenarios Ω'.

Tables 5 and 6 depict the results obtained. EVPI and VSS are provided in % against their corresponding objective function for SAA's very large set of scenario. One can see that the increase of penalty multiplier follows to the increase of SAA's average and standard deviation. It is also noticed that the value of EVPI (in %) is following the increase of the penalty multiplier. Concerning the variation on the distribution type, the Poisson distribution proposes the lowest SAA's average and the Uniform distribution has the lowest EVPI (in %).

Table 5.: Average Results on Penalty Multiplier Changes

Penalty Multiplier	$\mathcal{H}$					$\mathcal{H}^*$				
	μ	σ	CPU Time	EVPI %	VSS %	μ	σ	CPU Time	EVPI %	VSS %
3	1405.2	55.1	1315.9	54.1	3.1	1438.2	54.7	1921.3	53.7	4.1
5	2779	63.7	1332.1	46.8	2.8	1376.6	72.9	1662.4	51.8	9
10	6118	169	1257.9	40.5	1.4	6287.2	137	1828.6	40.8	1.4

Table 6.: Average Results of Distribution Type Change

Distribution	$\mathcal{H}$					$\mathcal{H}^*$				
	μ	σ	CPU Time	EVPI %	VSS %	μ	σ	CPU Time	EVPI %	VSS %
Normal	3715.1	250.3	1038.3	53.3	0	3820	221.3	1421.1	53.2	0
Poisson	3014.4	17.8	1515.1	49	3.7	1697.1	24.4	1975.1	54.2	10.6
Uniform	3572.7	19.7	1352.6	39.1	3.7	3584.8	18.9	2016.1	38.9	3.9

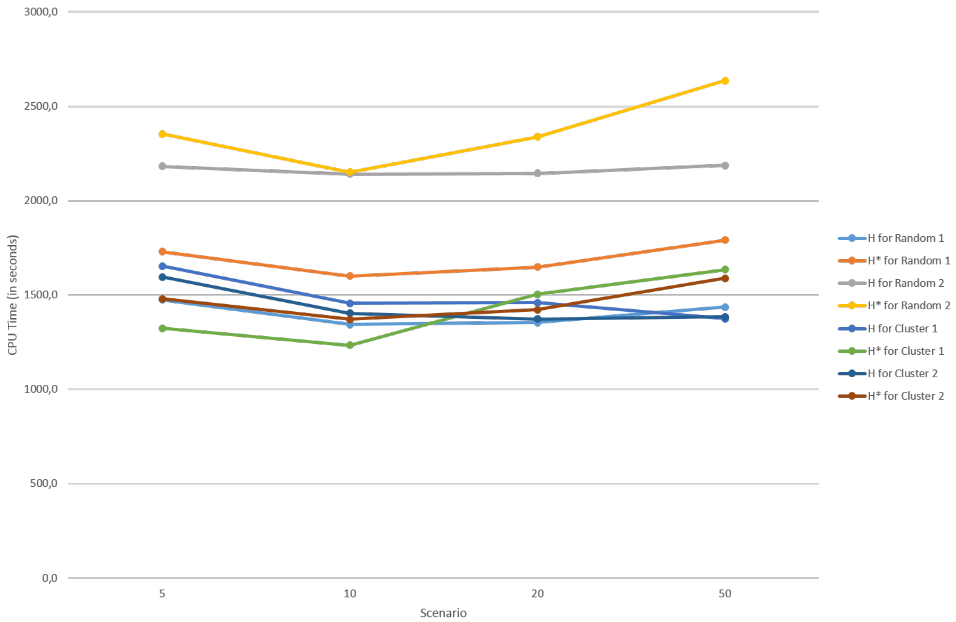

Figure 3.: Average CPU Times (in seconds) for All $\mathcal{M}$

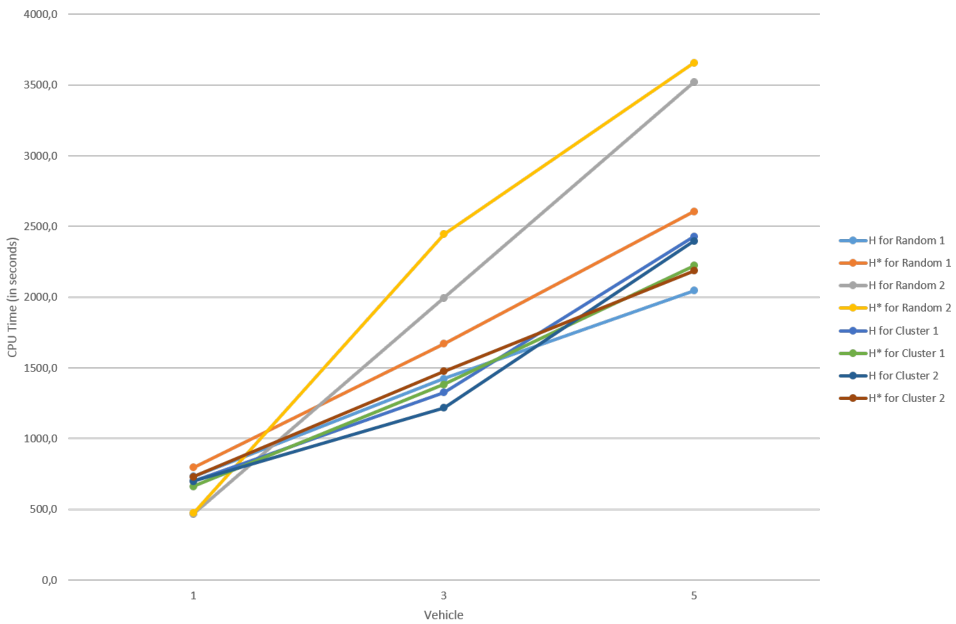

Figure 4.: Average CPU Times (in seconds) for All $\mathcal{K}$

6. Conclusions and Future Works

In this paper, the Stochastic Multi-Vehicle Collection-Disassembly Problem is formalised. It deals with the uncertainty of the quality and the quantity of EOL product as well as the demands of the associated components. The uncertain parameters are the number of components in each EOL product collected, the availability of EOL products at collection centres and their demands.

The problem is formalised as two-stage stochastic programming model in which the first-stage decisions have to be taken during the planning stage before any realisation of the uncertain parameters. The second-stage decisions are taken consecutively.

The first-stage decisions correspond to the number of vehicles dispatched and their routing decisions. Whilst, the second-stage decisions correspond to the inventory level, the number of disassembled and the unmet demands.

Accordingly, two approximate methods are developed to deal with i.e. Two-Phase Iterative Heuristic ($\mathcal{H}$) and Adaptive Two-Phase Iterative Heuristic ($\mathcal{H}^*$). $\mathcal{H}$ decomposes the problem into two subproblems: the stochastic reverse lot-sizing problem with approximate visiting costs and the routing problem. Then, the two subproblems are solved iteratively. $\mathcal{H}^*$ is an enhanced version of $\mathcal{H}$ through additional steps and an adaptive procedure to improve the solution provided by both subproblems. In deterministic problem, $\mathcal{H}^*$ outperforms $\mathcal{H}$.

Both methods are combined with the algorithmic framework of Sample Average Approximation (SAA). This framework allow to solve a problem having very large scenario, which is intractable, by solving the problem with smaller and tractable scenario. Also, the scenario generation of each uncertain parameter is carried out using a Monte Carlo-based simulation.

Four data sets were tested in order to demonstrate the applicability of the methods to provide feasible solutions. A sensitivity analysis was conducted to evaluate the performance of both methods for different penalty costs and for various distribution types of parameters under uncertainty.

The realisation of the parameters under uncertainty in a such problem comes right after the planning stage. It is also possible that such realisation related to the second-stage decision variables might also occurs in each period following the multi-stage stochastic programming. To formalise the problem, the formulation of SMCDP can be extended with additional constraints related to the second-stage decisions e.g. the inventory level, products disassembled and unmet demands. These constraints ensure the consistency of the decisions between scenarios.

Furthermore, companies commonly deals with more than one type of products. It indicates that extending SMCDP by dealing with multi-products is highly possible. Consequently, additional index related to EOL products needs to be incorporated in term of formulation. Due to its flexibility and performance, the Two-Phase Iterative Heuristic of (Absi et al. 2014) combined with the rollout algorithm of Bertsekas, Tsitsiklis, and Wu (1997) may also be implemented to tackle such problem.

7. Acknowledgement

The government of Auvergne-Rhône-Alpes region of France is gratefully acknowledged for funding this work.

References

Absi, N, C Archetti, S Dauzère-Pérès, and D Feillet. 2014. "A two-phase iterative heuristic approach for the production routing problem." *Transportation Science* 49 (4): 784–795.

Adulyasak, Yossiri, Jean-François Cordeau, and Raf Jans. 2014. "Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems." *INFORMS Journal on Computing* 26 (1): 103–120.

Table 7.: Results of Random 1

Instance	$\lvert\mathcal{K}\rvert$	$\lvert\Omega\rvert$	$\lvert\mathcal{M}\rvert$	$\mathcal{H}$			$\mathcal{H}^*$		
				μ	σ	CPU Time	μ	σ	CPU Time
49	1	5	200	497.6	20.8	1094.5	493.5	20.3	1150.9
		10	100	487.4	20.2	847.7	500.8	19.6	1011
		20	50	480.1	20.3	804.8	493.4	21.9	1087.9
		50	20	486.3	24.2	753.3	500.3	20.6	818.4
	3	5	200	494.9	21.3	2266.1	495	22	2189.7
		10	100	494.4	20.7	2044.9	501	21.1	2033.4
		20	50	496.8	20.8	1846.7	491.1	19.3	1863.5
		50	20	496.2	28.2	1631.6	499.2	21.6	1915.6
	5	5	200	496.1	21.2	3165.3	501.6	22.3	3020.1
		10	100	491.4	21	2620.7	492.4	21.9	2740.7
		20	50	495	21.8	2787.3	500.3	23.3	2762.9
		50	20	501	19.9	3898.2	506.1	21.3	2741.1
61	1	5	200	239.4	10.3	505.1	246.1	10	435.2
		10	100	239.5	10.3	380.7	247.3	9	378.8
		20	50	247.2	9.6	362.5	239.7	9.2	351.3
		50	20	241	11.5	333.8	242.1	8.9	283.8
	3	5	200	242.1	9.8	722.7	245.3	9.6	681.3
		10	100	240.1	9.3	686.1	241.9	10.2	589
		20	50	245.2	9.8	403.9	235.8	9.8	568
		50	20	236.5	9.1	500.7	245.5	13	485.2
	5	5	200	241.8	10.5	869.3	244.7	9.5	779.5
		10	100	238.1	10.3	733.1	242	11	651.4
		20	50	242	9.6	636.4	241.3	9.3	596.2
		50	20	242	10.9	751.4	239.4	9.6	565.1
73	1	5	200	1223.9	50.6	1882.6	1222.8	51.1	1767.2
		10	100	1233.4	48.1	1509.5	1210.2	55	1530.6
		20	50	1220.6	47.2	1568.4	1232	48	1503.8
		50	20	1217.4	49.1	1493.9	1234.2	40.7	1550.9
	3	5	200	1227	51.4	3304.9	1225.2	49.3	3690.6
		10	100	1224.4	50.5	3470.4	1216.3	55	3962.8
		20	50	1219.4	42.3	3150.9	1218.8	57.5	4160.3
		50	20	1216.9	54.2	3578.6	1236.1	47	4497.5
	5	5	200	1225.7	47.2	4285	1243.6	54.4	6365.8
		10	100	1225.3	50.2	4272	1221.9	48.9	6288.2
		20	50	1221.5	46.2	5372.5	1230.5	48.4	7675.5
		50	20	1226.3	41.6	5667.6	1224.5	67.5	9996.3
85	1	5	200	498.8	20.3	566.2	501.4	21.7	784.4
		10	100	504.9	22.6	501.3	491.7	20.2	643.7
		20	50	499.4	20.1	489.5	488.8	20.9	519.2
		50	20	494.2	19.5	394.4	491.8	18.9	516.4
	3	5	200	495.2	21.1	919.5	501.7	22.3	1468.9
		10	100	498.8	20.7	898.2	497	23.5	1237.7
		20	50	496.7	21.2	770.6	494.7	20.9	1038
		50	20	494.2	18.3	728.1	500.1	27.9	931.1
	5	5	200	490.6	22.9	1184.6	498.6	20.8	1678.3
		10	100	492.9	21	1056.6	494	18	1413.2
		20	50	499.2	19.9	1027.3	495.5	21.2	1284.2
		50	20	497	21.3	886.6	495.2	20.1	1195.1
97	1	5	200	243.7	9.5	340.3	242.5	10.1	432.4
		10	100	244.5	9.3	329.8	243.1	10.2	421
		20	50	239.3	11.9	248.2	241.1	10.6	338.7
		50	20	244.6	8.5	260.3	245.7	9.8	387.9
	3	5	200	242.2	10	417.7	240.9	10.2	728.2
		10	100	242.1	10	391.4	245.1	10.9	536.9
		20	50	236.8	9.8	377.5	247.8	11	423.9
		50	20	240.8	10.7	370.2	243.4	10.4	440.4
	5	5	200	242.2	9.6	598.1	241.6	9.6	763.5
		10	100	238	10.3	409.2	245.4	9.8	571.9
		20	50	241.9	9.2	455.5	246.4	9.4	542.2
		50	20	248.2	9.8	292.5	243.9	9	524.1

μ is average
σ is standard deviation
CPU time is in seconds

Adulyasak, Yossiri, Jean-François Cordeau, and Raf Jans. 2015a. "The production routing problem: A review of formulations and solution algorithms." *Computers & Operations Research* 55: 141–152.

Adulyasak, Yossiri, Jean-François Cordeau, and Raf Jans. 2012. "Optimization-based adaptive large neighborhood search for the production routing problem." *Transportation Science* 48 (1): 20–45.

Adulyasak, Yossiri, JF Cordeau, and Raf Jans. 2015b. "Benders decomposition for production routing under demand uncertainty." *Operations Research* 63 (4): 851–867.

Alumur, Sibel a., Stefan Nickel, Francisco Saldanha-da Gama, and Vedat Verter. 2012. "Multi-period reverse logistics network design." *European Journal of Operational Research* 220 (1): 67–78.

Amorim, P., M.a.F. Belo-Filho, F.M.B. Toledo, C. Almeder, and B. Almada-Lobo. 2013. "Lot sizing versus batching in the production and distribution planning of perishable goods." *International Journal of Production Economics* 146 (1): 208–218.

Archetti, Claudia, Luca Bertazzi, Giuseppe Paletta, and M. Grazia Speranza. 2011. "Analysis of the maximum level policy in a production-distribution system." *Computers & Operations Research* 38 (12): 1731–

Table 8.: Results of Random 2

Instance	$\lvert\mathcal{K}\rvert$	$\lvert\Omega\rvert$	$\lvert\mathcal{M}\rvert$	$\mathcal{H}$			$\mathcal{H}^*$		
				μ	σ	CPU Time	μ	σ	CPU Time
49	1	5	200	493.5	21	1083.2	495.2	21	986.7
		20	50	491.2	23.8	1045.2	496.9	21.6	872.6
		5	200	493.5	21	1083.2	495.2	21	986.7
		50	20	495.4	20.4	835.8	498	22	781.6
	3	10	100	498.2	21.3	1806.3	492.7	19.5	1678.9
		20	50	491.1	22.3	1647.2	505.4	19.9	1478.9
		5	200	490.6	20.8	2187.7	498.9	21.2	2161.6
		50	20	492	28.9	1929.7	496.3	23.4	1637.6
	5	10	100	493.2	22.5	5976.1	502.4	21.5	2500
		20	50	492	24.8	6545.2	505.4	20	2482.6
		5	200	493.8	21.9	6001	493.1	21.2	2983.3
		50	20	497.6	21.2	4535.5	499.2	19.5	2642.6
61	1	10	100	243	9.1	278.5	242.9	10.4	307.2
		20	50	244.1	9.3	243.6	243.4	10.6	288.8
		5	200	247.3	9.5	341.1	241	10.1	309.5
		50	20	240.2	8.9	245.6	246.6	9	246.6
	3	10	100	243.6	10.7	474.8	239.5	10.7	591.8
		20	50	237.8	10.8	484.2	241.8	9.6	532.2
		5	200	243.7	10.5	535.5	246.1	9.6	525.8
		50	20	237.8	9.5	376.4	242.4	10.8	398.9
	5	10	100	242.9	10.3	644	237.5	9.3	675.3
		20	50	244.9	9.2	502.1	240	10	631.3
		5	200	249.3	9.7	708.7	244.2	9.2	779.5
		50	20	240.2	9.8	561.4	237.6	8.4	595.1
73	1	10	100	1225.5	47.7	2844.3	1223.4	50.2	2742.8
		20	50	1231	58.2	2422	1237	52.3	2533.2
		5	200	1209.4	53.1	2806	1231.8	54.8	3201.2
		50	20	1218.7	48.6	2403.6	1221.4	55.2	3031
	3	10	100	1232.8	51	5773.8	1233.6	47	7329.7
		20	50	1220.2	47.7	6601.9	1228.6	57.1	8286.1
		5	200	1224.3	47.1	5757	1214	50.6	7632.9
		50	20	1218.7	46	6751.6	1221.4	54.5	10630.4
	5	10	100	1227.7	49.9	8989	1238.6	50.2	11097.4
		20	50	1223.1	57.4	8883.3	1217.1	53.7	14068.7
		5	200	1226.1	50.3	8530	1221.7	53.2	11650.8
		50	20	1219.4	56.8	11522.3	1217.1	53.8	15947.8
85	1	10	100	497.7	23.6	619.7	499.3	21.1	658
		20	50	501.6	17.6	610.7	495.1	21.6	555.6
		5	200	501.9	19.7	770.4	492.6	19.3	819.7
		50	20	499.3	24.9	528.7	501.1	27.6	492.7
	3	10	100	497.5	19.3	985.1	495.8	22.2	1271.1
		20	50	497.5	21.5	1021.5	494.1	20.3	943.7
		5	200	496.7	22.2	1199.6	497.7	20.9	1227
		50	20	491.7	24.3	846.6	492.7	24.1	827.5
	5	10	100	490.4	21.5	1313	501.4	20.2	1296.2
		20	50	500	22.3	1173.1	496.9	18.2	1235.6
		5	200	502.1	20.3	1378	495	20.2	1708.2
		50	20	500.2	16.8	1159.8	489.5	20.4	1318.6
97	1	10	100	236.6	10.4	292.9	236.6	9.6	349.6
		20	50	240.5	9.1	272.8	241.5	9.3	297.6
		5	200	239.2	10	363.9	243.4	10	343.2
		50	20	242.2	8.7	276.6	243.6	10.1	276.6
	3	10	100	248	9.9	399.3	239.8	10.2	448.1
		20	50	240.9	9.5	270.8	242.5	9.9	438.3
		5	200	236.3	9.3	444.5	242.5	10	510.7
		50	20	237.9	9.6	418	243.6	9.3	364.8
	5	10	100	241.6	9.8	561.9	237.9	9.4	335.4
		20	50	246.2	11.3	424.4	239.1	10.2	425.3
		5	200	242.4	10.6	618.8	253.5	9.5	452.4
		50	20	240.9	11	416.3	242.8	10.3	336.4

μ is average
σ is standard deviation
CPU time is in seconds

1746.

Armentano, V.a., a.L. Shiguemoto, and a. Lø kketangen. 2011. “Tabu search with path relinking for an integrated productiondistribution problem.” *Computers & Operations Research* 38 (8): 1199–1209.

Bard, Jonathan F., and Narameth Nananukul. 2009. “Heuristics for a multiperiod inventory routing problem with production decisions.” *Computers & Industrial Engineering* 57 (3): 713–723.

Bard, Jonathan F, and Narameth Nananukul. 2010. “Computers & Operations Research A branch-and-price algorithm for an integrated production and inventory routing problem.” *Computers and Operation Research* 37 (12): 2202–2217.

Bertazzi, Luca, G Paletta, and MG Speranza. 2005. “Minimizing the total cost in an integrated vendormanaged inventory system.” *Journal of heuristics* 11 (5): 393–419.

Bertsekas, DP, JN Tsitsiklis, and Cynara Wu. 1997. “Rollout algorithms for combinatorial optimization.” *Journal of Heuristics* 3 (3): 245–262.

Birge, John R, and Francois Louveaux. 2011. *Introduction to stochastic programming.* Springer Science &

Table 9.: Results of Cluster 1

Instance	$\|\mathcal{K}\|$	$\|\Omega\|$	$\|\mathcal{M}\|$	$\mathcal{H}$			$\mathcal{H}^*$		
				μ	σ	CPU Time	μ	σ	CPU Time
49	1	10	100	493.2	22.1	790.5	496.1	19	614.3
		20	50	489.9	22.3	782.6	500	20.4	692
		5	200	489.7	19.8	1008.6	497.4	20.6	668.3
		50	20	503.9	17.7	725.3	497.2	20.6	706.1
	3	10	100	493.9	21.1	1288.5	487.7	20.4	1227
		20	50	500.7	18.4	1314.7	492.9	23.3	1207.5
		5	200	497.7	22	1528.8	499.5	20.2	1450.4
		50	20	489.1	16.9	1383	505.3	25.9	1214.8
	5	10	100	498	21	5431.5	496.7	19.7	1888.1
		20	50	501.4	19.5	4459.2	480.5	20.7	2099.5
		5	200	480.7	21.4	5917	484.7	21.3	2155.5
		50	20	489.9	22	2969.9	495.5	20.1	2245.9
61	1	10	100	239.1	10.4	315.2	243	9.9	308.7
		20	50	240.5	9.5	329.9	239.4	10.7	256.3
		5	200	243.4	9.7	267.5	246.6	10.5	333.6
		50	20	244.3	10	400.2	247	10.2	309.3
	3	10	100	239.7	11.5	494.1	234.1	9.6	439.9
		20	50	241.2	11	463.4	243.6	9.2	410.7
		5	200	242.8	10.4	495.4	241.4	9.3	566.7
		50	20	237.6	9.6	364.2	241.3	11.5	440.7
	5	10	100	241	10	803.8	236.4	10.7	701.1
		20	50	237.5	9	701.2	243	9	566.6
		5	200	241.7	10.1	895.1	241.7	9.9	680.5
		50	20	246.6	10.7	660.8	236.6	10.5	507.1
73	1	10	100	1214.1	57.6	1681.3	1224.6	55.9	1432.6
		20	50	1207.3	47.7	1344.7	1229.2	50.4	1501.9
		5	200	1220.1	51.6	1740.5	1230.2	52.3	1602.8
		50	20	1236.6	61	1326	1220	43.8	1515.1
	3	10	100	1229.2	55.3	3065.3	1224.2	47.7	3009.2
		20	50	1224.8	48.6	3516.6	1217.7	45	4726.1
		5	200	1223.1	50.8	3612.4	1222	49	3155.6
		50	20	1225.6	55.3	3870.4	1211.9	51.8	4366.4
	5	10	100	1225	49.1	4431.5	1228.4	48.9	5279.9
		20	50	1224.1	53	5325.7	1222.2	51.5	7561
		5	200	1217.3	51.3	5139.9	1239.1	53	5021.3
		50	20	1231.1	49.5	5421.2	1217.4	52.4	9495.7
85	1	10	100	494.6	20.2	513.2	496.4	22.5	538.3
		20	50	497.8	22.9	478.4	498	20.5	478.3
		5	200	493.9	23.2	614.6	495.8	20.6	547.2
		50	20	499.4	19.7	428.1	492.4	18.7	488.2
	3	10	100	499.2	18.2	868.1	496.1	20.2	956.3
		20	50	493	18.3	891.9	500.2	23.3	987.1
		5	200	496.5	19.5	941.6	500	20.6	952.1
		50	20	497.3	19	809.7	488.1	22.3	949.2
	5	10	100	498.7	19.3	992	496.9	21.3	984.5
		20	50	502.5	21.8	1086.4	491.9	18.9	1047.5
		5	200	492.4	20.4	1305.7	497.9	22.8	1288.5
		50	20	488	20.5	1074.4	500.6	23.1	1132.1
97	1	10	100	237.7	10.5	310.3	242.4	10.4	334
		20	50	243.3	11.5	295.6	244.1	10.8	272.3
		5	200	241.5	10.2	350.2	243.6	9.2	355.2
		50	20	239.8	8.6	312.3	242.8	11.4	296.4
	3	10	100	242.2	9.4	379.7	240.8	10.2	370
		20	50	245.2	9.2	385	243.1	10.7	413.7
		5	200	240.1	10.5	441.3	240.5	10.1	458.6
		50	20	243.6	8.9	399.2	246	10.2	357.7
	5	10	100	239.3	9.2	479.1	241.5	10.1	414.1
		20	50	243.6	9.4	499.8	242	10.6	326.3
		5	200	244.9	10.1	525.7	243.9	9.6	610.9
		50	20	239.8	12.1	468.1	242	8.5	479.3

μ is average
σ is standard deviation
CPU time is in seconds

Business Media.

Boudia, M., M.a.O. Louly, and C. Prins. 2007. "A reactive GRASP and path relinking for a combined productiondistribution problem." *Computers & Operations Research* 34 (11): 3402–3419.

Boudia, M., and C. Prins. 2009. "A memetic algorithm with dynamic population management for an integrated productiondistribution problem." *European Journal of Operational Research* 195 (3): 703–715.

Brahimi, Nadjib, and Tarik Aouam. 2016. "Multi-item production routing problem with backordering: a MILP approach." *International Journal of Production Research* 54 (4): 1076–1093.

Buer, MG Van, DL Woodruff, and RT Olson. 1999. "Solving the medium newspaper production/distribution problem." *European Journal of Operational Research* 115 (2): 237–253.

Calvete, Herminia I., Carmen Galé, and María-José Oliveros. 2011. "Bilevel model for production-distribution planning solved by using ant colony optimization." *Computers & Operations Research* 38 (1): 320–327.

Çetinkaya, Sla, Halit Üster, Gopalakrishnan Easwaran, and Burcu Baris Keskin. 2009. "An Integrated

Table 10.: Results of Cluster 2

Instance	$\|\mathcal{K}\|$	$\|\Omega\|$	$\|\mathcal{M}\|$	$\mathcal{H}$			$\mathcal{H}^*$		
				μ	σ	CPU Time	μ	σ	CPU Time
49	1	10	100	486.2	19.9	704.4	503.6	19.8	595
		20	50	497.8	20.3	747.3	490.5	22.6	613.4
		5	200	503.2	21	903.6	497.2	21.2	723.8
		50	20	502	23.5	705	492.4	20.1	637.3
	3	10	100	500.2	20.4	1532.2	498.1	21.5	1512.9
		20	50	491.4	22.9	1273.8	497.6	18.9	1428.7
		5	200	488.3	20.8	1730.4	496.2	21	1829.5
		50	20	490.4	21	1360.1	491.7	20.3	1615
	5	10	100	499	19.5	6153.6	497.9	23.1	2115.3
		20	50	497.8	22.1	4860	496.5	18.7	2017.9
		5	200	501	22.8	6611.2	497.5	21.4	2387.4
		50	20	502	26.9	3893.3	496.4	17.3	2377.4
61	1	10	100	239.8	10.1	362.1	247.2	9.5	375.1
		20	50	240.5	9.4	365.2	239.6	12.1	381.7
		5	200	241.8	10.8	485.6	238.2	10.1	414.6
		50	20	243.5	8.8	338.1	242.3	9.4	336.6
	3	10	100	239.8	9.9	593.7	242.7	9.2	645.1
		20	50	233.7	9.5	576.5	241.9	9.3	482.1
		5	200	239.2	10.5	619.9	243	9.4	771.4
		50	20	240.4	8.5	531.9	245.8	10.2	621.5
	5	10	100	244.5	10	741.8	242.6	10.3	738.5
		20	50	232.3	10.3	717.4	239.6	8.7	691.3
		5	200	241.1	11	915.1	242.5	9.9	924.2
		50	20	239.1	9.5	598.3	239.2	8.5	762.1
73	1	10	100	1215.6	53.2	1675.4	1223.9	53.2	1807.7
		20	50	1217.4	47.6	1422.6	1229.1	48	1590.1
		5	200	1218.5	50.7	1916.9	1222.5	50.1	2118.9
		50	20	1216.8	48.5	1665.1	1225.1	51.4	1652.7
	3	10	100	1217.9	54	2626	1232.7	47.7	3798.4
		20	50	1223.7	52.2	3017.7	1224.2	51.8	3873.3
		5	200	1224.7	50.3	3494.5	1222.2	51.1	3673.4
		50	20	1230.7	57.8	3009.5	1226.1	41.6	4485.9
	5	10	100	1230.6	48	3550	1214.7	50.9	5099.8
		20	50	1230	54	4700.4	1225.2	54.4	7049.2
		5	200	1227.7	53.1	3927.8	1219.2	50.1	5221.7
		50	20	1217.3	52.1	5893.3	1213.6	56.9	8144.8
85	1	10	100	490	23	531.8	494.7	21.8	550.9
		20	50	490	21.4	493.4	500.6	18.2	478.8
		5	200	495.4	21.3	494.8	494	19.8	628.3
		50	20	494.6	20.4	398.1	495.5	21.2	468.9
	3	10	100	501.4	22.2	760.4	494.1	20	996.7
		20	50	493.1	22.8	665.2	496.1	17.6	782.9
		5	200	498.4	19.5	766.4	501.6	19.7	1035.1
		50	20	492.1	20.3	679.9	498.3	19	751.7
	5	10	100	493.6	21.7	1047.8	492.4	18.8	1291.7
		20	50	491.1	20.8	948.3	493.9	20.9	1023.7
		5	200	489.9	21.3	1186.9	497.1	20.3	1337.8
		50	20	495.1	21.8	965.2	493.3	18.8	1044.3
97	1	10	100	246.4	9.9	177.5	239	10.7	319.7
		20	50	243.1	11.7	206.8	247.2	9.5	291.9
		5	200	245.8	11.7	234.8	241.4	9.7	360.1
		50	20	240	9	164.4	241	9.5	289.4
	3	10	100	242.5	10.6	268.2	244.8	9.7	325.3
		20	50	243.6	9	261.5	244	9.8	273.4
		5	200	244.4	9.1	311.5	241.7	9.3	347.6
		50	20	236	12.3	271.5	239.2	9.6	286.9
	5	10	100	244.5	9.6	322.8	242.2	9.5	392.5
		20	50	242.6	11.1	318.5	241.5	9.1	352.7
		5	200	242.8	10.5	324.1	246.7	10.3	418.5
		50	20	246.9	9.4	286.7	248.1	8.4	361.4

μ is average
σ is standard deviation
CPU time is in seconds

Outbound Logistics Model for Frito-Lay: Coordinating Aggregate-Level Production and Distribution Decisions." *Interfaces* 39 (5): 460–475.

Chandra, Pankaj, and ML Fisher. 1994. "Coordination of production and distribution planning." *European Journal of Operational Research* 72 (3): 503–517.

Chen, Huey-Kuo, Che-Fu Hsueh, and Mei-Shiang Chang. 2009. "Production scheduling and vehicle routing with time windows for perishable food products." *Computers & Operations Research* 36 (7): 2311–2319.

Díaz-Madroñero, Manuel, David Peidro, and Josefa Mula. 2015. "A review of tactical optimization models for integrated production and transport routing planning decisions." *Computers & Industrial Engineering* .

Forum, World Economic. 2014. *Towards the circular economy: Accelerating the scale-up across global supply chains.* Technical Report January. Geneva: World Economic Forum.

Fumero, F, and C Vercellis. 1999. "Synchronized development of production, inventory, and distribution schedules." *Transportation science* 33 (3): 330–340.

Ghilas, Veaceslav, Emrah Demir, and Tom Van Woensel. 2016. "A scenario-based planning for the pickup and delivery problem with time windows, scheduled lines and stochastic demands." *Transportation Research Part B: Methodological* 91: 34–51.

Habibi, M.K. Khakim, Olga Battaïa, Van-Dat Cung, and Alexandre Dolgui. 2017a. "Collection-disassembly problem in reverse supply chain." *International Journal of Production Economics* 183: 334–344.

Habibi, MK Khakim, Olga Battaïa, Van-Dat Cung, and Alexandre Dolgui. 2017b. "An efficient two-phase iterative heuristic for Collection-Disassembly problem." *Computers & Industrial Engineering* 110: 505–514.

Kleywegt, AJ, Alexander Shapiro, and T Homem-de Mello. 2002. "The sample average approximation method for stochastic discrete optimization." *SIAM Journal on Optimization* 12 (2): 479–502.

Kumar, Ravi Shankar, Karthik Kondapaneni, Vijaya Dixit, a. Goswami, L.S. Thakur, and M.K. Tiwari. 2015. "Multi-objective modeling of production and pollution routing problem with time window: A self-learning particle swarm optimization approach." *Computers & Industrial Engineering* .

Kumar, Sameer, and Valora Putnam. 2008. "Cradle to cradle: Reverse logistics strategies and opportunities across three industry sectors." *International Journal of Production Economics* 115 (2): 305–315.

Lin, S., and B. W. Kernighan. 1973. "An Effective Heuristic Algorithm for the Traveling-Salesman Problem." *Operations Research* 21 (2): 498–516.

McGovern, S, and SM Gupta. 2011. *The disassembly line: balancing and modeling*. 1st ed. New York: McGraw-Hill.

Miranda, Pedro L, Jean-François Cordeau, Deisemera Ferreira, Raf Jans, and Reinaldo Morabito. 2018. "A Decomposition Heuristic for a Rich Production Routing Problem." *Computers & Operations Research* .

Pishvaee, Mir Saman, Reza Zanjirani Farahani, and Wout Dullaert. 2010. "A memetic algorithm for bi-objective integrated forward/reverse logistics network design." *Computers & Operations Research* 37 (6): 1100–1112.

Qiu, Yuzhuo, Ming Ni, Liang Wang, Qinqin Li, Xuanjing Fang, and Panos M Pardalos. 2018a. "Production routing problems with reverse logistics and remanufacturing." *Transportation Research Part E: Logistics and Transportation Review* 111: 87–100.

Qiu, Yuzhuo, Jun Qiao, and Panos M Pardalos. 2017. "A branch-and-price algorithm for production routing problems with carbon cap-and-trade." *Omega* 68: 49–61.

Qiu, Yuzhuo, Liang Wang, Xiaoling Xu, Xuanjing Fang, and Panos M Pardalos. 2018b. "Formulations and branch-and-cut algorithms for multi-product multi-vehicle production routing problems with startup cost." *Expert Systems with Applications* .

Qiu, Yuzhuo, Liang Wang, Xiaoling Xu, Xuanjing Fang, and Panos M Pardalos. 2018c. "A variable neighborhood search heuristic algorithm for production routing problems." *Applied Soft Computing* 66: 311–318.

Russell, Robert A. 2017. "Mathematical programming heuristics for the production routing problem." *International Journal of Production Economics* 193: 40–49.

Shiguemoto, André Luís, and Vinícius Amaral Armentano. 2010. "A tabu search procedure for coordinating production, inventory and distribution routing problems." *International Transactions in Operational Research* 17 (2): 179–195.

Solyalı, Oğuz, and Haldun Süral. 2009. "A relaxation based solution approach for the inventory control and vehicle routing problem in vendor managed systems." In *Modeling, computation and Optimization*, 171–189. World Scientific.

Solyalı, Oğuz, and Haldun Süral. 2017. "A multi-phase heuristic for the production routing problem." *Computers & Operations Research* 87: 114–124.