CIRRELT

Centre interuniversitaire de recherche
sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre
on Enterprise Networks, Logistics and Transportation

A Decomposition Algorithm for Dynamic Reverse Supply Chain Network Design Under Uncertainty

Mohammad Jeihoonian
Masoumeh Kazemi Zanjani
Michel Gendreau

October 2016

CIRRELT-2016-55

Bureaux de Montréal :
Université de Montréal
Pavillon André-Aisenstadt
C.P. 6128, succursale Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
Pavillon Palasis-Prince
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

A Decomposition Algorithm for Dynamic Reverse Supply Chain Network Design Under Uncertainty

Mohammad Jeihoonian[1,2,*], Masoumed Kazemi Zanjani[1,2], Michel Gendreau[2,3]

[1] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT)

[2] Department of Mechanical and Industrial Engineering, Concordia University, 1455 De Maisonneuve Blvd. W., Montreal, Canada H3G 1M8

[3] Department of Mathematics and Industrial Engineering, Polytechnique Montréal, P.O. Box 6079, Station Centre-Ville, Montréal, Canada H3C 3A7

Abstract. Motivated by the recovery practices of modular-structured products, this study addresses designing a reverse supply chain network while incorporating uncertainty in quantity of the return stream over a planning horizon. The stochastic parameter is modeled as a scenario tree in which each stage of decision making corresponds to a unique time period. The concerned problem is formulated as a multi-stage mixed-integer stochastic programming model. Considering a scenario clustering decomposition scheme, the proposed model is decomposed into smaller scenario cluster sub-models such that the sub-models are associated with a number of sub-trees that share a certain number of predecessor nodes in the scenario tree. The sub-models are coordinated into an implementable solution via a Lagrangean progressive hedging-based method which employs a Benders decomposition-based algorithm as a viable solution approach for each scenario cluster sub-model. Based on a realistic scale case, computational results indicate a consistent performance efficiency of the proposed scenario clustering decomposition approach.

Keywords: Reverse supply chain, durable products, multi-stage stochastic.

Acknowledgements. The authors gratefully acknowledge the financial support from le Fonds de recherche Nature et technologies (FRQNT) and the Natural Sciences and Engineering Research Council of Canada (NSERC).

---

* Corresponding author: Mohammad.Jeihoonian@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2016

1. Introduction

Reverse supply chain (RSC) network design refers to the decisions in terms of locations and capacities of facilities associated with the collection and recovery of end-of-life products in addition to the allocation of physical flows among these facilities and secondary markets. Designing RSC networks for durable products (e.g., large household appliances) that are distinguished by their modular structure and their long life cycle is a complex problem. It is explained by the fact that such category of products can be dissembled into several components, namely modules and parts along with a bulk of damaged yet recyclable components referred to as residues. Depending on the category and quality status of each component in the reverse bill of materials (BOM), a particular recovery process would be desired to reclaim the economic value residing in a specific component. For example, remanufacturing is a typical option for a used module in a good condition. Nonetheless, a poor quality (damaged) module is considered as residues that can be recycled to separate the precious raw materials from mixed scrap. Observing the variation in market demands of brand-new durable products, a similar tendency can be expected in generating end-of-life durable goods. In this regard, a dynamic perspective should be considered to accommodate such fluctuations in the RSC planning over a planning horizon. To date, a few contributions have addressed this concern [1–3]. Considering an application-oriented approach, Salema et al. [2] proposed a graph-based scheme to design a dynamic recovery network to capture the fluctuations in the rate of returns in a deterministic setting. In another attempt, Alumur et al. [3] developed a mixed-integer programming (MIP) formulation to model a RSC network design in a multi-period setting while considering the reverse BOM. The proposed model was also analyzed for a real-life industrial case.

An inherent characteristic of the recovery systems is the uncertainty in quality/quantity of returns. Needless to say, a successful designing of RSC networks requires the inclusion of such critical factors into the decision making problem. Most studies in the literature have utilized two-stage stochastic programming approaches to explicitly deal with uncertainty in static (single-period) settings [4–7]. The overview of the current literature indicates that most of the previous research in the context of designing RSCs under uncertainty is limited to single-period settings. In such studies, the common sources of uncertainty entail quality/quantity of returns, demands, and economical parameters such as shipping costs. For instance, Fonseca et al. [8]

provided a two-stage stochastic programming model for designing a RSC network under uncertainty in transportation costs. The dynamic nature of the product returns combined with the uncertainty in the quantity of returns lead to a natural extension of the static RSC network design into a dynamic setting under uncertainty which consequently calls for multi-stage stochastic programming [9] as a suitable approach to be adopted. In a multi-stage stochastic program, modeling the uncertain parameter, e.g., quantity of returns, as a scenario tree allows the adjustment of the decisions while more information on the uncertain parameter is available to the decision maker. In this line of research, Cardoso et al. [10] proposed a multi-stage mixed-integer programming (MS-MIP) model to maximize the expected net present value of designing a close-loop supply chain network over a planning horizon under uncertainty in demand. In a similar vein, more recently, Zeballos et al. [11] used a scenario tree approach for discretization of stochastic demand and quantity of returns over the planning horizon. The resulting MS-MIP model was solved by a commercial software. As it can be observed, the number of studies in the context of designing a dynamic recovery network while accounting uncertainty is limited. To fill the existing void of research, on the modeling side, the first contribution of this study is to address the problem of designing a RSC in a multi-period setting considering the reverse BOM of durable products. In the underlying problem of interest, the quantity of returns is stochastic and non-stationary during the planning horizon. It is worth noting that a push market is assumed for the recovered modules, parts, and materials, which is a realistic assumption in many industries. Hence, the demand for the recovered items is considered as a deterministic yet dynamic parameter. Through modeling the uncertain factor as a scenario tree, the problem is modeled as a MS-MIP in which one seeks to maximize the expected profit. The non-homogeneity characteristic of the components of a durable good is also incorporated in the design decisions through defining a finite number of quality levels. To the best of our knowledge, none of the aforementioned studies in dynamic RSC planning have addressed the impact of the quality status of components on the choice of the recovery option while accounting uncertainty in the quantity of returns.

One complicating aspect of MS-MIP models is their computational intractability, particularly due to the exponential growth in the number of decision variables over the stages of the scenario tree of the stochastic parameter(s). Scenario clustering decomposition schemes have been shown to successfully solve large-scale multi-stage stochastic programming problems

[12–15]. The prime idea of scenario clustering decomposition is to divide the scenario tree into a set of scenario clusters such that they share some ancestor nodes. In most studies in the context of dynamic recovery network design, the size of test instances is quite small allowing the plain use of MIP solvers. Therefore, as the second contribution, on the methodological side, a heuristic inspired by a scenario clustering decomposition scheme [14, 15] is provided to solve the resulting large-scale MS-MIP problem. This algorithm revolves around decomposing the scenario tree into smaller sub-trees. The MS-MIP model would consequently be broken down into smaller sub-models corresponding to each sub-tree. Afterwards, the scenario cluster sub-models are coordinated by Lagrangean penalty terms in the objective function and a progressive hedging-based scheme [16] is applied for updating Lagrangean multipliers [17]. It is noteworthy to state that each scenario cluster sub-model per se is a hard to solve problem. Hence, as the third contribution, a Benders decomposition-based (BD) solution algorithm [18] is developed to tackle each scenario cluster sub-model, which is enhanced with a Pareto-optimal cuts selection strategy [19].

The remainder of this article is organized as follows. In the next section, the description of the problem investigated in this article is provided and its formulation is introduced. Section 3 elaborates the details of the solution methodology including the scenario clustering decomposition and Benders decomposition schemes. Computational experiments on a case of large household appliances, i.e., washing machines, is presented in Section 4. Finally, Section 5 concludes this paper.

2. Problem statement

2.1. Problem description

Considering a dynamic RSC network design context, as shown in Figure 1, in each period in the planning horizon, used products that are of non-homogeneous quality status are acquired in collection zones and then shipped to disassembly centers. The returns are then graded into multiple quality levels in disassembly centers. As noted in the preceding section, depending on the quality level of the component in the reverse BOM of a durable product, it can be sent to a particular facility for the recovery process. Hence, high quality modules are sent to remanufacturing centers and high quality parts are used for part harvesting to make them “like-new” components. These components are then offered at a lower price compared to the brand-new components at their corresponding marketplaces. For instance, in a washing machine, its motor and washing tube are categorized, respectively, as modules and parts. The high quality level motor is profitable for remanufacturing while a poor quality washing tube is sent to the bulk recycling center. Damaged components are also shipped to bulk recycling facilities to recycle precious raw materials. The unprocessed raw materials are then purchased by a third-party logistics provider. More precisely, it is assumed that there exists an infinite demand for recyclable raw materials in markets. It is also assumed that the waste of residues is safely disposed of in bulk recycling facilities at zero cost, as it is a sunk cost.

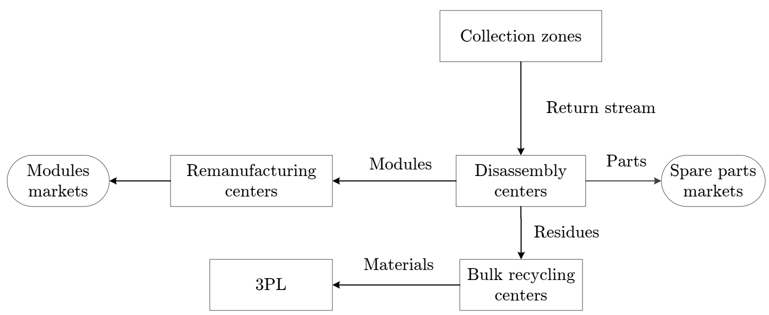

Figure 1: The reverse supply chain network

In a deterministic setting, the design decisions in each period revolve around the location of each facility including disassembly, remanufacturing, and bulk recycling centers to be installed in the RSC network. It should be noted that a dynamic RSC design provides the flexibility to adjust the number and the location of facilities as the quantity of returns evolves over time. More precisely, depending on the quantity of returns over the planning horizon, either some of the existing facilities are closed (in case of a reduced return stream) or new facilities are opened (in case of an increasing return stream). Furthermore, the planning decisions include the physical flows and inventory levels at each facility. The objective function is to maximize the profit over the planning horizon. The original equipment manufacturer gains revenues from remanufacturing when the remanufactured modules are sold in the secondary markets; from reusable parts when they are sold to spare parts markets; from bulk recycling when the unprocessed raw materials are purchased by the third-party logistics provider. The total cost comprises the fixed costs of the installation of facilities along with inventory holding, processing, and transportation costs in the RSC network. Furthermore, the following assumptions are made regarding the problem setting.

- Demands of remanufactured modules and reusable parts are deterministic yet dynamic over the planning horizon;
- The return stream is categorized with respect to a finite set of quality levels;
- The unit collection, disassembly, and remanufacturing costs are quality status-dependent;
- Capacities of facilities are not subject to change within time periods.

2.2. Modeling uncertain returns

In the problem of interest, it is assumed that the quantity of returns is uncertain and dynamic; hence, it evolves as a discrete time stochastic process over the planning horizon. As noted earlier, the dynamic and uncertain nature of returns quantities require the adjustment of the design decisions during the planning horizon. To this end, the planning horizon is discretized into a finite set of time periods such that the decisions are implemented at the end of each time period. Considering the multi-period setting together with uncertainty, the stochastic quantity of returns parameter can be interpreted as a scenario tree in which each stage indicates the realization of the uncertain parameter. It is assumed that each stage corresponds to a single time period. In a given stage, each node represents a distinguishable state of random return concerning the available information up to this stage. In the underlying problem, each node is directly connected to two other nodes while moving away from the root node. In other words, each node in the scenario tree has only one sibling except the root node. Besides, a return quantity scenario is defined as the full path from the root node, i.e., the current state of world, to a leaf node at the last stage of the scenario tree. Figure 2 illustrates a scenario tree with four stages.

2.3. Problem formulation

Given the RSC network design problem described in Section 2.1 and the scenario tree representing the uncertain quantity of returns, in this section, the latter problem is formulated as a MS-MIP model. In this model, the location of facilities depends on the quantity of returns, hence this decision

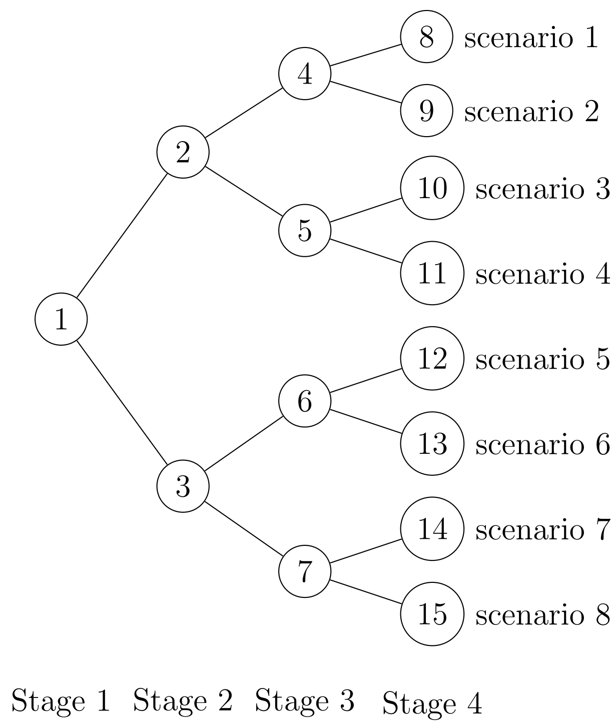

Figure 2: Scenario tree for the random quantity of returns

is defined for each possible realization of the stochastic parameter in each period, represented by a node in each stage of the scenario tree. In a similar fashion, the quantity of acquisition, disassembly quantity, the flow between the disassembly and recovery facilities, shipped quantities of recovered items, as well as the inventory levels of recovered items at different facilities also depend on the return quantity, thus are defined for each node. Furthermore, in any given period in the planning horizon (represented as a stage in the scenario tree), the decision maker cannot foresee future outcomes of the return quantity; therefore, location and flow decisions must satisfy the non-anticipativity condition (NAC). The latter indicates that such decisions in a given period (stage) are identical for scenarios with a common ancestor node in that period. For instance, in Figure 2, scenarios 1 and 2 share node 2 in stage 2; therefore, the location and flow decisions must be identical for both scenarios at this stage and consequently are defined exclusively for the ancestor node 2. On the other hand, since inventory decisions are state variables that depend on the main design/flow decisions, the NAC would automatically apply. The problem notation is provided in Appendix A. The compact formulation of the MS-MIP model corresponding to the dynamic RSC network design problem under investigation can be stated as follows. It should be noted that the NAC is implicitly taken into consideration while using a compact formulation of a multi-stage stochastic problem. On the contrary, the latter condition must be explicitly stated in a split-variable or clustered formulation.

Total revenue

$$\sum_{n \in Tree} pr(n) \Bigg\{ \sum_{t \in T} \sum_{a \in A} \sum_{o \in O} \sum_{p \in P} Ps_p QS_{aopt}(n) + \sum_{t \in T} \sum_{d \in D} \sum_{w \in W} \sum_{l \in L} Pw_l QW_{dwlt}(n) + \sum_{t \in T} \sum_{b \in B} \sum_{r \in R} Pe_r BR_{brt}(n) \Bigg\} \tag{1}$$

Total cost

Fixed cost

$$\sum_{n \in Tree} pr(n) \Bigg\{ \sum_{t \in T} \sum_{a \in A} fa_a YA_{at}(n) + \sum_{t \in T} \sum_{d \in D} fd_d YD_{dt}(n) + \sum_{t \in T} \sum_{b \in B} fb_b YB_{bt}(n) \Bigg\} \tag{2}$$

Processing cost

$$\sum_{n \in Tree} pr(n) \Bigg\{ \sum_{t \in T} \sum_{c \in C} \sum_{a \in A} \sum_{q \in Q} ca_{aq} QA_{caqt}(n) + \sum_{t \in T} \sum_{a \in A} \sum_{d \in D} \sum_{l \in L} cd_{dl} QD_{adlt}(n) \sum_{t \in T} \sum_{a \in A} \sum_{b \in B} cb_b QB_{abt}(n) \Bigg\} \tag{3}$$

Inventory holding cost

$$\sum_{n\in Tree} pr(n)\Bigg\{\sum_{t\in T}\sum_{a\in A}\sum_{p\in P} hp_p IP_{apt}(n) + \sum_{t\in T}\sum_{a\in A}\sum_{l\in L} hl_l IL_{alt}(n) + \sum_{t\in T}\sum_{a\in A} hbIB_{at}(n) + \sum_{t\in T}\sum_{d\in D}\sum_{l\in L} hl_l ID_{dlt}(n)\Bigg\} \tag{4}$$

Transportation cost

$$\sum_{n\in Tree} pr(n)\Bigg\{\sum_{t\in T}\sum_{c\in C}\sum_{a\in A}\sum_{q\in Q} ta_{ca}QA_{caqt}(n)\sum_{t\in T}\sum_{a\in A}\sum_{o\in O}\sum_{p\in P} ts_{aop}QS_{aopt}(n) + \sum_{t\in T}\sum_{a\in A}\sum_{d\in D}\sum_{l\in L} td_{adl}QD_{adlt}(n) + \sum_{t\in T}\sum_{a\in A}\sum_{b\in B} tbQB_{abt}(n) + \sum_{t\in T}\sum_{d\in D}\sum_{w\in W}\sum_{l\in L} tw_{dwl}QW_{dwlt}(n)\Bigg\} \tag{5}$$

Supply constraints

$$\sum_{a\in A} QA_{caqt}(n) = \psi_{cqt}(n) \quad c\in C, q\in Q, t\in T, n\in Tree \tag{6}$$

Flow balance constraints

$$IP_{apt}(n) = IP_{ap(t-1)}(m) + \sum_{c\in C}\sum_{q\in Q}\gamma_{pq}QA_{caqt}(n) - \sum_{o\in O} QS_{aopt}(n) \quad a\in A, p\in P, t\in T, n\in Tree, m=a(n) \tag{7}$$

$$IL_{alt}(n) = IL_{al(t-1)}(m) + \sum_{c\in C}\sum_{q\in Q}\delta_{lq}QA_{cat}(n) - \sum_{d\in D} QD_{adlt}(n) \quad a\in A, l\in L, t\in T, n\in Tree, m=a(n) \tag{8}$$

$$IB_{at}(n) = IB_{a(t-1)}(m) + \sum_{c\in C}\sum_{q\in Q}\beta_q QA_{caqt}(n) - \sum_{b\in B} QB_{abt}(n) \quad a\in A, t\in T, n\in Tree, m=a(n) \tag{9}$$

$$ID_{dlt}(n) = ID_{dl(t-1)}(m) + \sum_{a\in A} QD_{adlt}(n) - \sum_{w\in W} QW_{dwlt}(n) \quad d\in D, l\in L, t\in T, n\in Tree, m=a(n) \tag{10}$$

$$\sum_{a\in A}\eta_r QB_{abt}(n) = BR_{brt}(n) \quad b\in B, r\in R, t\in T, n\in Tree \tag{11}$$

Demand constraints

$$\sum_{a \in A} QS_{aopt}(n) = ds_{opt} \quad o \in O, p \in P, t \in T, n \in Tree \tag{12}$$

$$\sum_{d \in D} QW_{dwlt}(n) = dw_{wlt} \quad w \in W, l \in L, t \in T, n \in Tree \tag{13}$$

Capacity constraints of facilities

$$\sum_{c \in C} \sum_{q \in Q} QA_{caqt}(n) \leq caa_a YA_{at}(n) \quad a \in A, t \in T, n \in Tree \tag{14}$$

$$\sum_{a \in A} QD_{azlt}(n) \leq cad_{dl} YD_{dt}(n) \quad d \in D, l \in L, t \in T, n \in Tree \tag{15}$$

$$\sum_{a \in A} QB_{abt}(n) \leq cab_b YB_{bt}(n) \quad b \in B, t \in T, n \in Tree \tag{16}$$

In model (1)-(16), the objective function accounts for maximizing the expected profit. Constraint (6) ensures the acquisition of the return stream for each node and each time period. Constraints (7)-(10) are inventory balance restrictions, respectively, for parts, modules, and residues at disassembly centers in addition to remanufacturing facilities. Flow balance restriction in each bulk recycling center is imposed by Constraints (11). Constraints (12)-(13) ensure the demand satisfaction of parts and remanufactured modules at their corresponding marketplaces in each time period. Constraints (14)-(16) impose capacity restriction on disassembly, remanufacturing, and bulk recycling centers.

3. Solution methodology

Solving the MS-MIP model (1)-(16) by a commercial solver for real-size instances is a challenge. This is due to the existence of the three sets of binary variables that increase exponentially in number as the number of stages in the planning horizon is increased. As noted earlier, the computational complexity has motivated the authors to propose a heuristic scenario clustering decomposition (HSCD) algorithm. This algorithm comprises two major steps: (1) Scenario cluster decomposition (SCD) and (2) Scenario cluster coordination (SCC). In the SCD step, first, the scenario tree is partitioned into a set of scenario cluster sub-trees. Then, for each sub-tree, the corresponding MS-MIP model is represented in a compact formulation. Furthermore, the

NACs corresponding to common nodes in the original scenario tree are introduced in the objective function of each scenario cluster sub-model. Finally, in the SCC step, the aforementioned sub-models are coordinated into an implementable solution by means of a Lagrangian Progressive Hedging-based algorithm. The details of the aforementioned steps are provided as follows.

3.1. Step 1: SCD

3.1.1. Decomposing the scenario tree

Definition 1. Given that n^{δ} and S represent, respectively, the set of nodes that belong to stage δ and the set of scenario clusters, according to [14, 15], a break stage δ^* is defined as a stage in the scenario tree such that the following equation holds: $|S| = |n^{\delta^*+1}|$.

In Figure 3, if the second stage is chosen as the break stage, i.e., $\delta^* = 2$, four scenario clusters ($|S| = 4$) are obtained such that each shares node 1. Furthermore, the first and the second scenario cluster sub-trees share node 2 and the other two share node 3 in the original scenario tree (see Figure 3).

Let introduce N^s as the set of nodes that belong to scenario cluster s, $\Delta = \{1, 2, .., \delta^*\}$, N_1 as the set of nodes corresponding to the stages in Δ, $N_2 = N \backslash N_1$, $N_1^s = N_1 \cap N^s$, $N_2^s = N_2 \cap N^s$. Moreover, let ζ^{ω} be the likelihood of scenario ω, Ω_s be the set of scenarios in scenario cluster sub-tree s, and $\zeta^s(n) = \sum_{\omega \in \Omega_s} \zeta^{\omega}$.

3.1.2. Formulating scenario cluster sub-models

Following the scenario tree decomposition, the MS-MIP model is formulated for each sub-tree in a compact representation. The NACs for any node in N_2 are implicitly considered by formulating each scenario cluster sub-tree in a compact representation. However, these constraints are required to explicitly be taken into account for every node in N_1. The purpose of introducing the NACs is to coordinate and link $|S|$ scenario cluster sub-models into an implementable solution. Let $X_t^s(n)$ be the vector of flow and location variables in the MS-MIP (1)-(16). Let P be the vector of unit prices of selling brand-new and recovered components at the marketplaces. Let F and C be, respectively, the vector of fixed costs of opening facilities in the reverse network and the vector of procurement, processing, inventory carrying, and transportation costs. Moreover, let η^n be scenario cluster sub-trees that share node n, $\underline{s}_{\eta^n} = min\{s | \forall s \in \eta^n\}$, $\overline{s}_{\eta^n} = max\{s | \forall s \in \eta^n\}$. The NACs can therefore be stated as follows.

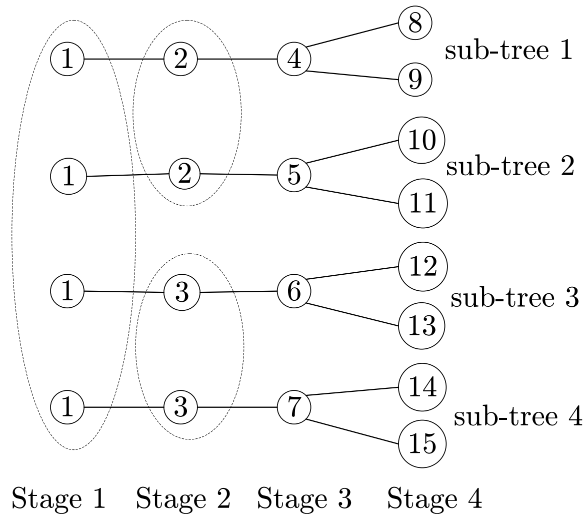

Figure 3: The scenario cluster sub-trees

$$X_t^s(n) - X_t^{s+1}(n) \leq 0 \quad \forall s = \underline{s}_{\eta^n}, ..., (\underline{s}_{\eta^n}) - 1, t \in \Delta, n \in N_1 \tag{17}$$

$$X_t^{\overline{s}_{\eta^n}}(n) - X_t^{\underline{s}_{\eta^n}}(n) \leq 0 \quad t \in \Delta, n \in N_1 \tag{18}$$

For instance, in the sub-trees shown in Figure 3, the NACs for the location of disassembly centers, i.e., $YA_{at}(n)$, is expressed as follows.

$$\begin{aligned}
YA^1_{a2}(2) - YA^2_{a2}(2) \leq 0 \quad \forall a \in A \\
YA^2_{a2}(2) - YA^1_{a2}(2) \leq 0 \quad \forall a \in A \\
YA^3_{a2}(3) - YA^4_{a2}(3) \leq 0 \quad \forall a \in A \\
YA^4_{a2}(3) - YA^3_{a2}(3) \leq 0 \quad \forall a \in A
\end{aligned}$$

By dualizing the NACs and using a Lagrangean multiplier vector, i.e., $\mu_t^s(n)$, the MS-MIP model (1)-(16) can be reformulated as the following multi-stage scenario cluster Lagrangean decomposition (MSCLD) problem. [14, 15].

$$
\begin{aligned}
Z_{MSCLD}(\mu,s) = max & \sum_{s=1}^{|S|}\sum_{n\in N_1^s}\sum_{t\in\Delta}\zeta^s(n)\{PX_t^s(n)-FX_t^s(n)-CX_t^s(n)\} \\
& + \sum_{s=1}^{|S|}\sum_{n\in N_2^s}\sum_{t\notin\Delta}pr(n)\{PX_t^s(n)-FX_t^s(n)-CX_t^s(n)\} \\
& \sum_{s=\underline{s}_{\eta n}}^{(\bar{s}_{\eta n})-1}\sum_{n\in N_1}\sum_{t\in\Delta}\mu_t^s(n)\{X_t^{s+1}(n)-X_t^s(n)\} \\
& + \sum_{n\in N_1}\sum_{t\in\Delta}\mu_t^{\bar{s}_{\eta n}}(n)\{X_t^{\underline{s}_{\eta n}}(n)-X_t^{\bar{s}_{\eta n}}(n)\} \\
& s.t. \quad (6)-(16) \quad \forall t\in\Delta, n\in\{N_1,N_2\}
\end{aligned} \tag{19}
$$

As it can be seen, (19) is a relaxation of MS-MIP (1)-(16) for all $\mu_t^s(n)\geq 0; \quad \forall s\in S, n\in N_1^s$, and $t\in\Delta$. Thus, the value of its objective function, $Z_{MSCLD}(\mu,s)$, is an upper bound on the optimal solution of the original MS-MIP model.

Definition 2. The dual problem (Lagrangean dual) of the original MS-MIP model with respect to NACs (17)-(18), for a given break stage δ^*, can be represented as

$$Z_{MSCLD} = min_{\mu\geq 0}Z_{MSCLD}(\mu,s) \tag{20}$$

The Lagrangean dual problem (20) is solved by an iterative sub-gradient-based scheme to identify an upper bound on the original MS-MIP model (1)-(16). It should be stated that model (19) can further be decomposed into $|S|$ subproblems in accordance with each scenario cluster sub-tree. Its objective function can also be attained through summing up each individual sub-model objective function as follows.

$$Z_{MSCLD}(\mu,s) = \sum_{s=1}^{|S|} Z_{MSCLD}^s(\mu) \tag{21}$$

3.2. Step 2: SCC

3.2.1. Lagrangean progressive hedging-based algorithm (LPHA)

In order to update Lagrangean multipliers, a Lagrangean progressive hedging-based scheme is considered as presented in Escudero et al. [14, 17].

The progressive hedging algorithm was firstly introduced in the seminal work of Rockafellar and Wets [16] for solving multi-stage stochastic linear programming models.

Definition 3. The classical sub-gradient vector $g_t(n)$ in which $n \in N_1^s, t \in \Delta$ can be defined as follows [14].

$$g_t(n) = \begin{pmatrix} X_t^{\underline{s}_{\eta^n}}(n) - X_t^{\bar{s}_{\eta^n}}(n) \\ X_t^{(\underline{s}_{\eta^n})+1}(n) - X_t^{\underline{s}_{\eta^n}}(n) \\ . \\ . \\ . \\ X_t^{\bar{s}_{\eta^n}}(n) - X_t^{((\bar{s}_{\eta^n})-1)}(n) \end{pmatrix}$$

In LPHA scheme, apart from the classical sub-gradient vector, a new modified vector is defined.

Definition 4. Denoted by $\overline{g}_t(n)$, a non-necessary feasible pseudo sub-gradient vector can be represented as follows.

$$\overline{g}_t(n) = \begin{pmatrix} \overline{X}_t(n) - X_t^{\bar{s}_{\eta^n}}(n) \\ \overline{X}_t(n) - X_t^{\underline{s}_{\eta^n}}(n) \\ . \\ . \\ . \\ \overline{X}_t(n) - X_t^{((\bar{s}_{\eta^n})-1)}(n) \end{pmatrix}$$

where $\overline{X}_t(n) = \sum_{s \in \eta^n} \zeta^s(n) X^s(n)$ such that, $n \in N_1^s$ and $t \in \Delta$. In fact, $\overline{X}_t(n)$ indicates an approximate expected value over the set of scenario cluster sub-trees that share node n. Denoting by $\underline{Z}_{MSCLD}$ a lower bound on (19), the details of LPHA method are summarized in Algorithm 1.

The termination conditions require that either the sub-gradient vector $g_t^k(n)$ is less than a threshold, i.e., 0.01, or the value of Z_{MSCLD} is not improved in a sequence of consecutive iterations, i.e., 5 iterations.

Each iteration of LPHA calls for the solutions of $|S|$ independent sub-models. In the context of the dynamic RSC network design investigated in this study, each sub-model is itself a large-scale optimization problem which cannot be solved by the plain use of MIP engines, e.g., CPLEX. Therefore, a Benders decomposition-based algorithm tailored to the particular structure of each scenario cluster sub-model is proposed in the next section. This algorithm is nested within the LPHA algorithm to solve the $|S|$ scenario cluster sub-models.

Algorithm 1 - LPHA

Step 0: **Initialization** $\mu_t^s(n) \leftarrow 0 \quad \forall s \in S, n \in N_1^s, t \in \Delta$, $\bar{\alpha} \leftarrow 1$, Iteration counter $k \leftarrow 1$

Solve $|S|$ scenario cluster sub-models (19) independently

Compute the initial value of $Z_{MSCLD}(\mu, s)$

while termination conditions are not satisfied **do**

Step 1: Calculate $g_t^k(n)$ and $\bar{g}_t^k(n) \quad \forall n \in N_1^s, t \in \Delta$

Step 2: Update Lagrangean multipliers

$\mu_t^{k+1}(n) \leftarrow max\{0, \mu_t^k(n) + \bar{\alpha}.\frac{(Z_{MSCLD}(\mu^k,s) - \underline{Z}_{MSCLD})}{\left\|\bar{g}_t^k(n)\right\|^2}.\bar{g}k_t(n)\}$

Step 3: Solve $|S|$ scenario cluster sub-models with μ^{k+1} and update $Z_{MSCLD}(\mu^{k+1}, s)$

Step 4: $k \leftarrow k+1$

end while

3.2.2. Benders decomposition-based algorithm

The hallmark of Benders decomposition is to exploit the decomposable structure present in the formulation of the MIP model. In other words, in a MIP model, integer/binary variables are seen as complicating variables such that when fixing them, the MIP model reduces to smaller subproblems (PSP), which can be solved individually to generate cutting planes for the master problem (MP). The PSP and MP are then solved sequentially and iteratively until a termination criterion is satisfied. As the classical Benders decomposition turns out to converge slowly in the underlying problem due to the degeneracy of PSP, a cut selection strategy based on the work of Papadakos [19] is proposed to expedite the convergence of the solution process.

3.2.2.1 Benders reformulation

Given a particular scenario cluster sub-tree $s = (\underline{s}_{\eta^n}) + 1, ..., \overline{s}_{\eta^n}$ in (21) and a vector of feasible location decisions, i.e., $\overline{\boldsymbol{Y^s}} = \{YA_{at}^s(n), YD_{dt}^s(n), YB_{bt}^s(n)\}$, the PSP can be formulated as follows.

$$
\begin{aligned}
Z^{s}_{MSCLD}(\mu) = max \sum_{n\in N_1^s}\sum_{t\in\Delta}\zeta^s(n)\Bigg\{ & \sum_{a\in A}\sum_{o\in O}\sum_{p\in P} Ps_p QS^s_{aopt}(n) + \sum_{d\in D}\sum_{w\in W}\sum_{l\in L} Pw_l QW^s_{dwlt}(n) \\
& + \sum_{b\in B}\sum_{r\in R} Pe_r BR^s_{brt}(n) - \sum_{c\in C}\sum_{a\in A}\sum_{q\in Q} ca_{aq} QA^s_{caqt}(n) + \sum_{a\in A}\sum_{d\in D}\sum_{l\in L} cd_{dl} QD^s_{adlt}(n) \\
& - \sum_{a\in A}\sum_{b\in B} cb_b QB^s_{abt}(n) - \sum_{a\in A}\sum_{p\in P} hp_p IP^s_{apt}(n) - \sum_{a\in A}\sum_{l\in L} hl_l IL^s_{alt}(n) - \sum_{a\in A} hb IB^s_{at}(n) \\
& - \sum_{d\in D}\sum_{l\in L} hl_l ID^s_{dlt}(n) - \sum_{c\in C}\sum_{a\in A}\sum_{q\in Q} ta_{ca} QA^s_{caqt}(n) - \sum_{a\in A}\sum_{o\in O}\sum_{p\in P} ts_{aop} QS^s_{aopt}(n) \\
& - \sum_{a\in A}\sum_{d\in D}\sum_{l\in L} td_{adl} QD^s_{adlt}(n) - \sum_{a\in A}\sum_{b\in B} tb QB^s_{abt}(n) - \sum_{d\in D}\sum_{w\in W}\sum_{l\in L} tw_{dwl} QW^s_{dwlt}(n) \\
& + \sum_{n\in N_2^s}\sum_{t\notin\Delta} pr(n)\Bigg\{ \sum_{a\in A}\sum_{o\in O}\sum_{p\in P} Ps_p QS^s_{aopt}(n) - \sum_{d\in D}\sum_{w\in W}\sum_{l\in L} Pw_l QW^s_{dwlt}(n) \\
& - \sum_{b\in B}\sum_{r\in R} Pe_r BR^s_{brt}(n) - \sum_{c\in C}\sum_{a\in A}\sum_{q\in Q} ca_{aq} QA^s_{caqt}(n) + \sum_{a\in A}\sum_{d\in D}\sum_{l\in L} cd_{dl} QD^s_{adlt}(n) \\
& - \sum_{a\in A}\sum_{b\in B} cb_b QB^s_{abt}(n) - \sum_{a\in A}\sum_{p\in P} hp_p IP^s_{apt}(n) - \sum_{a\in A}\sum_{l\in L} hl_l IL^s_{alt}(n) - \sum_{a\in A} hb IB^s_{at}(n) \\
& - \sum_{z\in Z}\sum_{l\in L} hl_l ID^s_{dlt}(n) - \sum_{c\in C}\sum_{a\in A}\sum_{q\in Q} ta_{ca} QA^s_{caqt}(n) - \sum_{a\in A}\sum_{o\in O}\sum_{p\in P} ts_{aop} QS^s_{aopt}(n) \\
& - \sum_{a\in A}\sum_{d\in D}\sum_{l\in L} td_{adl} QD^s_{adlt}(n) - \sum_{a\in A}\sum_{b\in B} tb QB^s_{abt}(n) - \sum_{d\in D}\sum_{w\in W}\sum_{l\in L} tw_{dwl} QW^s_{dwlt}(n) \Bigg\} \\
& + \sum_{n\in N_1}\sum_{t\in\Delta}\Bigg\{ \sum_{a\in A}\sum_{o\in O}\sum_{p\in P} (\mu^{1,s-1}_{aopt}(n) - \mu^{1,s}_{aopt}(n)) QS^s_{aopt}(n) \\
& + \sum_{d\in D}\sum_{w\in W}\sum_{l\in L} (\mu^{2,s-1}_{dwlt}(n) - \mu^{2,s}_{dwlt}(n)) QW^s_{dwlt}(n) + \sum_{b\in B}\sum_{r\in R} (\mu^{3,s-1}_{brt}(n) - \mu^{3,s}_{brt}(n)) BR^s_{brt}(n) \\
& + \sum_{c\in C}\sum_{a\in A}\sum_{q\in Q} (\mu^{4,s-1}_{caqt}(n) - \mu^{4,s}_{caqt}(n)) QA^s_{caqt}(n) + \sum_{a\in A}\sum_{d\in D}\sum_{l\in L} (\mu^{5,s-1}_{adlt}(n) - \mu^{5,s}_{adlt}(n)) QD^s_{adlt}(n) \\
& + \sum_{a\in A}\sum_{b\in B} (\mu^{6,s-1}_{abt}(n) - \mu^{6,s}_{abt}(n)) QB^s_{abt}(n) \Bigg\} \qquad (22)
\end{aligned}
$$

$$
s.t. \quad (6) - (13) \quad \forall t \in \Delta\backslash\{1\}, n \in N^s
$$

$$\sum_{c\in C}\sum_{q\in Q} QA^s_{caqt}(n) \leq caa_a \overline{YA}^s_{at}(n) \quad a\in A, t\in\Delta, n\in N^s \tag{23}$$

$$\sum_{a\in A} QD^s_{azlt}(n) \leq cad_{dl} \overline{YD}^s_{dt}(n) \quad d\in D, l\in L, t\in\Delta, n\in N^s \tag{24}$$

$$\sum_{a\in A} QB^s_{abt}(n) \leq cab_b \overline{YB}^s_{bt}(n) \quad b\in B, t\in\Delta, n\in N^s \tag{25}$$

where $\boldsymbol{\mu^{1,s}}$,..., $\boldsymbol{\mu^{6,s}}$ denote the set of Lagrangean multipliers for scenario cluster sub-tree s. Note that for sub-tree $s = \underline{s}_{\eta^n}$, the terms of the objective function (22) that correspond to Lagrangean multipliers are written as the following compact representation: $\sum_{n\in N_1}\sum_{t\in\Delta}\{\mu_t^{\overline{s}_{\eta^n}}(n) - \mu_t^{\underline{s}_{\eta^n}}(n)\} X_t^{\underline{s}_{\eta^n}}(n)$.

Let $\boldsymbol{\upsilon^{1,s}}$,..., $\boldsymbol{\upsilon^{11,s}}$ be the set of dual variables corresponding to constraints (6)-(13) and (23)-(25) in which $\boldsymbol{\upsilon^{9,s}}$, $\boldsymbol{\upsilon^{10,s}}$, and $\boldsymbol{\upsilon^{11,s}}$ are non-negative. The dual subproblem (DSP) can be formulated as follows .

$$\begin{aligned} Z^s_{\boldsymbol{\upsilon}}(\overline{\boldsymbol{Y^s}}) = \min \quad & \sum_{c\in C}\sum_{q\in Q}\sum_{t\in\Delta}\sum_{n\in N^s} \psi_{cqt}(n)\upsilon^{1,s}_{cqt}(n) + \sum_{o\in O}\sum_{p\in P}\sum_{t\in\Delta}\sum_{n\in N^s} ds_{opt}\upsilon^{7,s}_{opt}(n) \\ &+ \sum_{w\in W}\sum_{l\in L}\sum_{t\in\Delta}\sum_{n\in N^s} dw_{wlt}\upsilon^{8,s}_{wlt}(n) + \sum_{a\in A}\sum_{t\in\Delta}\sum_{n\in N^s} caa_a\overline{YA}^s_{at}(n)\upsilon^{9,s}_{at}(n) \\ &+ \sum_{d\in D}\sum_{l\in L}\sum_{t\in\Delta}\sum_{n\in N^s} cad_{dl}\overline{YD}^s_{dt}(n)\upsilon^{10,s}_{dl}(n) \\ &+ \sum_{b\in B}\sum_{t\in\Delta}\sum_{n\in N^s} cab_b\overline{YB}^s_{bt}(n)\upsilon^{11,s}_{bt}(n) \end{aligned} \tag{26}$$

$$s.t. \quad (\boldsymbol{\upsilon^{1,s}}, \boldsymbol{\upsilon^{2,s}}, ..., \boldsymbol{\upsilon^{11,s}}) \in \Lambda^s \tag{27}$$

where Λ^s denotes the polyhedron defined by the constraints of the DSP for a particular scenario cluster sub-tree s. Let θ^s be a surrogate variable that is an upper bound on (22). Furthermore, let $\rho(.)$ entails all terms in the objective function of DSP (26) independent of the location variables. The master problem (MP) can be stated as follows.

$$\max \quad \theta^s - \sum_{n\in N_1^s}\sum_{t\in\Delta}\zeta^s(n)\left\{\sum_{a\in A} fa_a YA^s_{at}(n) + \sum_{d\in D} fd_d YD^s_{dt}(n) + \sum_{b\in B} fb_b YB^s_{bt}(n)\right\}$$

$$
\begin{aligned}
&- \sum_{n \in N_2^s} \sum_{t \notin \Delta} pr(n) \Bigg\{ \sum_{a \in A} fa_a YA^s_{at}(n) + \sum_{d \in D} fd_d YD^s_{dt}(n) + \sum_{b \in B} fb_b YB^s_{bt}(n) \Bigg\} \\
&+ \sum_{n \in N_1} \sum_{t \in \Delta} \Bigg\{ \sum_{a \in A} (\mu^{7,s-1}_{at}(n) - \mu^{7,s}_{at}(n)) YA^s_{at}(n) + \sum_{d \in D} (\mu^{8,s-1}_{dt}(n) - \mu^{8,s}_{dt}(n)) YD^s_{dt}(n) \\
&+ \sum_{b \in B} (\mu^{9,s-1}_{bt}(n) - \mu^{9,s}_{bt}(n)) YB^s_{bt}(n) \Bigg\} \qquad (28)
\end{aligned}
$$

$$
\begin{aligned}
s.t. \quad & \theta^s \leq \rho(\hat{\boldsymbol{\upsilon}}^{(\boldsymbol{m},\boldsymbol{s})^T}) + \sum_{a \in A} \sum_{t \in \Delta} \sum_{n \in N^s} caa_a YA^s_{at}(n) \hat{\upsilon}^{9,s}_{at}(n) \\
& + \sum_{d \in D} \sum_{l \in L} \sum_{t \in \Delta} \sum_{n \in N^s} cad_{dl} YD^s_{dt}(n) \hat{\upsilon}^{10,s}_{dl}(n) + \sum_{b \in B} \sum_{t \in \Delta} \sum_{n \in N^s} cab_b YB^s_{bt}(n) \hat{\upsilon}^{11,s}_{bt}(n) \qquad (29) \\
& 0 \leq \rho(\hat{\boldsymbol{\kappa}}^{(\boldsymbol{m},\boldsymbol{s})^T}) + \sum_{a \in A} \sum_{t \in \Delta} \sum_{n \in N^s} caa_a YA^s_{at}(n) \hat{\kappa}^{9,s}_{at}(n) \\
& + \sum_{d \in D} \sum_{l \in L} \sum_{t \in \Delta} \sum_{n \in N^s} cad_{dl} YD^s_{dt}(n) \hat{\kappa}^{10,s}_{dl}(n) + \sum_{b \in B} \sum_{t \in \Delta} \sum_{n \in N^s} cab_b YB^s_{bt}(n) \hat{\kappa}^{11,s}_{bt}(n) \qquad (30) \\
& YA^s_{at}(n), YD^s_{dt}(n), YB^s_{bt}(n) \in \{0,1\} \quad \forall a \in A, d \in D, b \in B, t \in \Delta, n \in N^s \qquad (31)
\end{aligned}
$$

where $\boldsymbol{\kappa^s}$ denotes extreme rays of Λ^s when the DSP is unbounded for a given location solution and scenario cluster sub-tree s. As for sub-tree $s = \underline{s}_{\eta^n}$, the fourth term of (28) is written as follows.

$$
\begin{aligned}
&\sum_{n \in N_1} \sum_{t \in \Delta} \Bigg\{ \sum_{a \in A} (\mu^{7,\overline{s}_{\eta^n}}_{at}(n) - \mu^{7,\underline{s}_{\eta^n}}_{at}(n)) YA^{\underline{s}_{\eta^n}}_{at}(n) + \sum_{d \in D} (\mu^{8,\overline{s}_{\eta^n}}_{dt}(n) - \mu^{8,\underline{s}_{\eta^n}}_{dt}(n)) YD^{\underline{s}_{\eta^n}}_{dt}(n) \\
&+ \sum_{b \in B} (\mu^{9,\overline{s}_{\eta^n}}_{bt}(n) - \mu^{9,\underline{s}_{\eta^n}}_{bt}(n)) YB^{\underline{s}_{\eta^n}}_{bt}(n) \Bigg\}
\end{aligned}
$$

At each iteration of the Benders decomposition algorithm, if the DSP is bounded, an optimality cut (29) is generated given a vector of optimal dual solutions. Otherwise, a feasibility cut (30) is introduced to the MP to eliminate values of location decisions for which the PSP is infeasible.

3.2.2.2 Pareto-optimal cuts

The degeneracy of the PSP implies that there exist multiple optimal solutions for the DSP such that each of these leads to a distinct optimality cut. An efficient implementation of Benders decomposition algorithm requires a cut selection scheme to choose the deepest cut among the various optimality cuts which can be generated by arbitrarily taking optimal dual solutions. Papadakos [19] proposed a dual selection strategy to expedite the Benders algorithm. In the context of the underlying problem, let Γ indicates the polyhedron defined as $\Gamma = \{\boldsymbol{Y} : (31) \text{ holds}\}$.

Definition 5. A core point is defined as any point $\boldsymbol{Y^0}$ in the relative interior of the convex hull of Γ, i.e., $\boldsymbol{Y^0} \in ri(\Gamma^c)$. Γ^c and $ri(.)$ indicate the convex hull and the relative interior of Γ, respectively.

Definition 6. An optimality cut (29) associated with $(\boldsymbol{v_1^{1,s}}, \boldsymbol{v_1^{7,s}}, \boldsymbol{v_1^{8,s}}, \boldsymbol{v_1^{9,s}}, \boldsymbol{v_1^{10,s}}, \boldsymbol{v_1^{11,s}}) \in \Lambda^s$ dominates the one that corresponds to $(\boldsymbol{v_2^{1,s}}, \boldsymbol{v_2^{7,s}}, \boldsymbol{v_2^{8,s}}, \boldsymbol{v_2^{9,s}}, \boldsymbol{v_2^{10,s}}, \boldsymbol{v_2^{11,s}}) \in \Lambda^s$ if and only if

$$
\begin{aligned}
&\rho(\boldsymbol{\hat{v}_1^{(m,s)}}^T) + \sum_{a\in A}\sum_{t\in\Delta}\sum_{n\in N^s} caa_a YA^s_{at}(n)\hat{\upsilon}^{9,s}_{1at}(n) \\
&+ \sum_{d\in D}\sum_{l\in L}\sum_{t\in\Delta}\sum_{n\in N^s} cad_{dl} YD^s_{dt}(n)\hat{\upsilon}^{10,s}_{1dl}(n) + \sum_{b\in B}\sum_{t\in\Delta}\sum_{n\in N^s} cab_b YB^s_{bt}(n)\hat{\upsilon}^{11,s}_{1bt}(n) \\
&\le \rho(\boldsymbol{\hat{v}_2^{(m,s)}}^T) + \sum_{a\in A}\sum_{t\in\Delta}\sum_{n\in N^s} caa_a YA^s_{at}(n)\hat{\upsilon}^{9,s}_{2at}(n) \\
&+ \sum_{d\in D}\sum_{l\in L}\sum_{t\in\Delta}\sum_{n\in N^s} cad_{dl} YD^s_{dt}(n)\hat{\upsilon}^{10,s}_{2dl}(n) + \sum_{b\in B}\sum_{t\in\Delta}\sum_{n\in N^s} cab_b YB^s_{bt}(n)\hat{\upsilon}^{11,s}_{2bt}(n)
\end{aligned}
$$

for all $\boldsymbol{Y}$ with a strict inequality for at least one extreme point. A Pareto-optimal cut by definition is an optimality cut that is not dominated by any other cut. It can be obtained through using the optimal solution of the following auxiliary DSP.

$$
\begin{aligned}
Z^s_{\boldsymbol{v}}(\boldsymbol{Y}^0) = \min \quad & \sum_{c\in C}\sum_{q\in Q}\sum_{t\in\Delta}\sum_{n\in N^s} \psi_{cqt}(n)\upsilon^{1,s}_{cqt}(n) + \sum_{o\in O}\sum_{p\in P}\sum_{t\in\Delta}\sum_{n\in N^s} ds_{opt}\upsilon^{7,s}_{opt}(n) \\
+ \sum_{w\in W}\sum_{l\in L}\sum_{t\in\Delta}\sum_{n\in N^s} & dw_{wlt}\upsilon^{8,s}_{wlt}(n) + \sum_{a\in A}\sum_{t\in\Delta}\sum_{n\in N^s} ca_{at} YA^0_{at}(n)\upsilon^{9,s}_{at}(n) \\
+ \sum_{d\in D}\sum_{l\in L}\sum_{t\in\Delta}\sum_{n\in N^s} & cad_{dl} YD^0_{dt}(n)\upsilon^{10,s}_{dl}(n) + \sum_{b\in B}\sum_{t\in\Delta}\sum_{n\in N^s} cab_b YB^0_{bt}(n)\upsilon^{11,s}_{bt}(n) \\
s.t. \quad & (\boldsymbol{v^{1,s}}, \boldsymbol{v^{2,s}}, ..., \boldsymbol{v^{11,s}}) \in \Lambda^s \qquad (32)
\end{aligned}
$$

In this modified Benders decomposition algorithm, one starts with an initial core point, i.e., $\boldsymbol{Y}^0 = \{0.5\}$, to build a Part-optimal cut to be added to the MP. In the subsequent iterations, when the solution to the MP yields a feasible PSP, the auxiliary DSP (32) is solved, using a new core point that is the convex combination of the MP solution and the previous value of the core point, to generate a non-dominated cut. To this end, a non-negative parameter, i.e., λ, is considered as the weight of the core point $\boldsymbol{Y}^0$ in the convex combination to update the value of the core point throughout the solution process. The value of this parameter is assigned to be 0.5 [6, 20]. The description of the proposed Benders decomposition-based method is outlined in Algorithm 2.

Algorithm 2 - Benders decomposition-based algorithm

$UB \leftarrow \infty$, $LB \leftarrow -\infty$, $\boldsymbol{Y^0} = \{0.5\}$, $\lambda \leftarrow 0.5$
while $(UB - LB)/UB \leq \epsilon$ **do**
Solve auxiliary-DRSP (32)
Add Pareto-optimal cut (29) to the MP
Solve the MP
Update UB
Solve the DSP
if the DSP is unbounded **then**
    Add the feasibility cut (30) to the MP
    $\boldsymbol{Y^0} \leftarrow \lambda \boldsymbol{Y^0} + \xi$
else
    Add the optimality cut (29) to the MP
    Update LB, if necessary
    $\boldsymbol{Y^0} \leftarrow \lambda \boldsymbol{Y^0} + (1-\lambda)\overline{\boldsymbol{Y}}$
end if
end while
Solve the PSP

4. Numerical example

In this section, the performance of applying the solution scheme on the proposed model is investigated with respect to a set of test problems. To this end, first, the specific settings of the concerned case example is provided. The example is a typical large household appliance, i.e., a washing machine, that follows the settings of a case study presented in [21]. It should be noted that the parameter settings of the proposed MS-MIP model are carefully estimated vis-à-vis recent market data and current CLSC/RSC network design literature ([3, 4]). Then, it is followed by the computational results section. In this study, all algorithms are implemented in C++ programming language using Concert Technology with IBM-ILOG CPLEX 12.60 on an Intel Quad Core 3.40 GHz with 8 GB RAM. Moreover, the default settings of CPLEX are employed to solve the DSP and the MP in the Benders decomposition algorithm.

4.1. Experimental design

The BOM of the washing machines is described in Table 1. More specifically, each washing machine consists of ten parts (e.g., balance) and two modules (e.g., motor). All used machines acquired in collection points are of two quality levels, i.e., high and poor.

Table 1: Components of the case example

Description	Value
Parts	washing tube:1 (3.5 kg), cover:1 (2.5 kg), balance:1 (2.5 kg), frame:1 (11.5 kg), condenser:1 (0.5 kg), hose:1 (1 kg), small electric parts:1 (1 kg), electric wire:1 (1 kg), transformer:1 (1 kg), PCB:1 (0.5 kg)
Modules	motor:1 (5 kg), clutch:1 (4 kg)

The uncertain parameter, i.e., the quantity of returns, is normally distributed with a mean of 400 and a variance equal to 20% of the mean for high quality returns. As for the poor quality ones, the mean is considered to be 600 while the variance is equal to 20% of the mean. These normal distributions are then approximated by a 2-point discrete distribution (high and low ratio) through using the Gaussian quadrature method [22]. The time horizon is divided into three equal time periods such that each of them spans five years. Consequently, the time horizon is clustered into four stages (stage zero is the present time). Moreover, the scenario tree of the stochastic quantity of returns entails fifteen nodes and eight scenarios.

In Appendix B, as depicted in Tables B1 - B5, a summary of other parameters used in the case example is provided. Besides, shipping costs are selected from $Uniform(4,\ 7)$ for the used washing machines, $Uniform(1,\ 4)$ for each type of components, and $Uniform(0.1,\ 0.5)$ for bulk of residues. Capacities of facilities are randomly generated aligned with the stochastic quantity of returns and the BOM. For example, the capacity of disassembly centers are chosen between $Uniform(2 \times MeanCaa, 3 \times MeanCaa)$ where $MeanCaa = |C| \times (400 + 600)/|A|$. Moreover, the fixed cost of installing a facility is proportional to its capacity, so that a facility with high capacity level requires a greater investment.

In order to carry out the experiments, four main classes within each five test instances are considered as shown in Table 2. The detailed information on the size of the classes including the number of constraints, continuous variables, and binary variables are shown in Table 3. It is worth noting that the largest class of test instances (C4) reasonably reflects real-size RSCs in the context of durable products.

Table 2: Test problem classes

Class	C	A	D	B	O	W
1	40	5	5	2	20	20
2	50	10	10	5	25	25
3	60	10	10	5	30	30
4	70	15	15	7	35	35

Table 3: Size of test problems

Class	# Constraint	# Continuous variable	# Binary variable
1	5992	28014	168
2	8680	68390	350
3	9800	80290	350
4	12432	140434	518

4.2. Computational results

On each of the twenty test instances, the proposed decomposition scheme, i.e., HSCD, is applied to find an upper bound within the stopping criteria, i.e., either the sub-gradient vector is less than 1% or the current value of the upper bound is not improved in 5 iterations. The second stage is chosen as the break stage leading to four scenario cluster sub-models. As for the resolution of each scenario cluster sub-model, the Benders decomposition-based algorithm described in the preceding section is employed with the stopping criteria of either 1% optimality gap or 3600 seconds time limitation. Alternatively, for the sake of comparison, considering the first stage as the break stage, each test instance is also solved using the Benders decomposition-based algorithm where the termination condition is either a time limit of 24 hours or an optimality gap of 1% for each of the resulting two scenario clusters. It should be noted that by decomposing the scenario tree in the first stage, two sub-models are obtained such that each of them can be independently solved. The optimal solution of each sub-model individually yields a sub-optimal solution to the optimal solution of the MS-MIP model. Note that it is not required to impose any NAC in the first stage of the tree as it corresponds to time period zero where the initial inventory levels are zero.

Table 4 presents the results obtained by HSCD and Benders decomposition algorithm algorithms for all twenty test instances. For the former approach, columns "Time" and "#Iteration" indicate the total CPU time in seconds and the number of iterations, respectively. The fourth column shows the best upper bound on the MS-MIP model identified through applying HSCD. As for the latter approach, column "Time" indicates the amount of time required to solve the MS-MIP model (1)-(16) within 1% of optimality gap while column "Profit" gives the value of the objective function within the dedicated time limit and the optimality gap. Moreover, column "gp(%)" denotes the relative difference between upper and lower bounds reported by Benders decomposition algorithm within 24 hours running. It should be stated that the runtime of BD approach is considered as the maximum of the solution times of the two scenario cluster sub-models in each test instance. The last column, "Gap(%), expresses the relative difference in percentage between the solutions obtained by the two approaches.

The results show that the performance of the HSCD scheme is quite promising in the sense that it provides an upper bound on the objective function of the MS-MIP model in significantly less amount of time compared to the BD approach, i.e, 2 hours over all test instances. Given the low gap values in the last column (0.58% on average), even though the solutions are not necessarily feasible, they are quite close to those provided by the BD approach. More precisely, the average solution time of HSCD in solving the test instances of C1, C2, and C3 is, respectively, 4 minutes, 1.7 hours, and 1.9 hours. It increases to 4.5 hours for the last class of test problems. Furthermore, the infeasibility rate of the dualized NACs in the HSCD algorithm is on average 1% or less for each class of test problems.

On other hand, except the test instances of the first class, the second instance of C2, and the fourth instance of C3, the BD algorithm is unable

Table 4: Comparison of HSCD and BD algorithms

Class	HSCD			BD			Gap(%)
	Time (sec)	# Iteration	$Z_{MSCLD}(\mu, s)$	Time (sec)	gp(%)	Profit	
C1	132	10	7,662,040	110	≤ 1	7,650,870	0.14
	183	9	6,412,000	120	≤ 1	6,351,610	0.94
	319	10	7,600,490	1371	≤ 1	7,596,140	0.06
	160	8	7,564,320	182	≤ 1	7,443,070	1.60
	422	10	6,881,080	2019	≤ 1	6,864,140	0.25
C2	5695	12	7,201,340	$\geq 24hr$	1.15	7,106,850	1.31
	2231	10	6,622,640	54701	≤ 1	6,611,560	0.17
	8391	15	6,568,970	$\geq 24hr$	1.10	6,558,930	0.15
	7729	13	6,296,800	$\geq 24hr$	1.40	6,280,200	0.26
	7100	15	7,095,570	$\geq 24hr$	1.12	6,978,190	1.65
C3	4190	13	9,854,950	$\geq 24hr$	1.50	9,684,640	1.73
	9113	16	9,617,400	$\geq 24hr$	1.14	9,562470	0.57
	5445	13	9,577,230	$\geq 24hr$	1.10	9,568740	0.09
	7184	14	9,643,130	61971	≤ 1	9,626,300	0.17
	7081	14	9,481,810	$\geq 24hr$	1.20	9,378,090	1.10
C4	13593	19	8,676,980	$\geq 24hr$	1.98	8,631,030	0.53
	16050	21	9,434,940	$\geq 24hr$	1.63	9,431,570	0.04
	17171	23	8,612,290	$\geq 24hr$	1.20	8,611,640	0.01
	19265	25	9,874,380	$\geq 24hr$	1.53	9,839,310	0.35
	14622	19	8,617,780	$\geq 24hr$	1.11	8,578,770	0.46
Average	7304	15	-	-	-	-	0.58

to obtain the optimal solution of the concerned test instances within the dedicated time limit and the optimality gap. Particularly, once the results of C1 for which the optimal solutions are given by the decomposition method are concerned, HSCD provides the upper bounds close to the optimal solution, i.e., 0.58% on average. As for other instances, high quality feasible solutions are reported by BD after the 24 hours time limit. More specifically, the optimality gap of the algorithm reported is less on 1.8% on average for such instances. In the targeted instances, HSCD also provides an upper bound close to the feasible solution identified by Benders decomposition as shown in the last column of Table 4.

5. Concluding remarks

In this study, a reverse supply chain network problem in a multi-period setting was addressed for taking back and recovery of used products that are of heterogeneous quality states. The concerned problem arises in the context of durable products which typically are composed of many components. As the inherent uncertainty in quantity of returns is assumed to evolve as a discrete time stochastic process during the planning horizon, a scenario tree was generated to model the uncertain parameter. The resulting multi-stage decision making problem was modeled as a MS-MIP model to address the decisions on the location of facilities and the quantity of flows in the reverse supply chain network.

In order to solve the proposed model for realistic sizes, a heuristic scenario clustering decomposition was proposed which mainly decomposes the scenario tree into a set of cluster of scenarios. The scenario clusters were independently solved by the Benders decomposition-based algorithm and coordinated in an implementable solution thorough a Lagrangean Progressive hedging-based scheme. The proposed solution scheme provided good upper bounds on the objective function of the original stochastic model in a reasonable amount of running time. It can be noticed not only by the closeness of the upper bounds to the solutions reported by the Benders decomposition-based algorithm, but also by the fact that the infeasibility rate of the dualized NACs is small for each class of test problems.

Given the multi-period setting, the underlying problem can be extended through accounting uncertainty in quality status of the return stream and demands. Another promising venue of research is to address the willingness of durable goods holders to return their used units by means of financial incentives.

Acknowledgments

The authors gratefully acknowledge the financial support from Le Fonds de recherche du Québec-Nature et technologies (FRQNT) and the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] H. Min, H.-J. Ko, The dynamic design of a reverse logistics network from the perspective of third-party logistics service providers, International Journal of Production Economics 113 (1) (2008) 176–192.

[2] M. I. G. Salema, A. P. Barbosa-Povoa, A. Q. Novais, Simultaneous design and planning of supply chains with reverse flows: a generic modelling framework, European Journal of Operational Research 203 (2) (2010) 336–349.

[3] S. A. Alumur, S. Nickel, F. Saldanha-da Gama, V. Verter, Multi-period reverse logistics network design, European Journal of Operational Research 220 (1) (2012) 67–78.

[4] O. Listeş, A generic stochastic model for supply-and-return network design, Computers & Operations Research 34 (2) (2007) 417–442.

[5] M. I. G. Salema, A. P. Barbosa-Povoa, A. Q. Novais, An optimization model for the design of a capacitated multi-product reverse logistics network with uncertainty, European Journal of Operational Research 179 (3) (2007) 1063–1077.

[6] M. Jeihoonian, M. Kazemi Zanjani, M. Gendreau, Closed-Loop Supply Chain Network Design under Uncertain Quality Status: Case of Durable Products, International Journal of Production Economics (In press).

[7] W. Chen, B. Kucukyazici, V. Verter, M. J. Sáenz, Supply chain design for unlocking the value of remanufacturing under uncertainty, European Journal of Operational Research 247 (3) (2015) 804–819.

[8] M. C. Fonseca, Á. García-Sánchez, M. Ortega-Mier, F. Saldanha-da Gama, A stochastic bi-objective location model for strategic reverse logistics, Top 18 (1) (2010) 158–184.

[9] J. R. Birge, F. Louveaux, Introduction to stochastic programming, Springer Science & Business Media, 2011.

[10] S. R. Cardoso, A. P. F. Barbosa-Póvoa, S. Relvas, Design and planning of supply chains with integration of reverse logistics activities under demand uncertainty, European Journal of Operational Research 226 (3) (2013) 436–451.

[11] L. J. Zeballos, C. A. Méndez, A. P. Barbosa-Povoa, A. Q. Novais, Multi-period design and planning of closed-loop supply chains with uncertain supply and demand, Computers & Chemical Engineering 66 (2014) 151–164.

[12] L. F. Escudero, M. A. Garín, M. Merino, G. Pérez, On BFC-MSMIP strategies for scenario cluster partitioning, and twin node family branching selection and bounding for multistage stochastic mixed integer programming, Computers & Operations Research 37 (4) (2010) 738–753.

[13] P.-L. Carpentier, M. Gendreau, F. Bastin, Optimal scenario set partitioning for multistage stochastic programming with the progressive hedging algorithm, Technical report CIRRELT-2013-55, CIRRELT .

[14] L. F. Escudero, M. A. Garín, A. Unzueta, Cluster Lagrangean decomposition in multistage stochastic optimization, Computers & Operations Research 67 (2016) 48–62.

[15] M. Kazemi Zanjani, O. S. Bajgiran, M. Nourelfath, A hybrid scenario cluster decomposition algorithm for supply chain tactical planning under uncertainty, European

Journal of Operational Research 252 (2) (2016) 466–476.

[16] R. T. Rockafellar, R. J.-B. Wets, Scenarios and policy aggregation in optimization under uncertainty, Mathematics of operations research 16 (1) (1991) 119–147.

[17] L. F. Escudero, M. A. Garín, G. Pérez, A. Unzueta, Scenario Cluster Decomposition of the Lagrangian dual in two-stage stochastic mixed 0–1 optimization, Computers & Operations Research 40 (1) (2013) 362–377.

[18] J. F. Benders, Partitioning procedures for solving mixed-variables programming problems, Numerische Mathematik 4 (1) (1962) 238–252.

[19] N. Papadakos, Practical enhancements to the Magnanti–Wong method, Operations Research Letters 36 (4) (2008) 444–449.

[20] M. Jeihoonian, M. Kazemi Zanjani, M. Gendreau, Accelerating Benders Decomposition for Closed-Loop Supply Chain Network Design: Case of Used Durable Products with Different Quality Levels, European Journal of Operational Research 251 (3) (2016) 830–845.

[21] P.-J. Park, K. Tahara, I.-T. Jeong, K.-M. Lee, Comparison of four methods for integrating environmental and economic aspects in the end-of-life stage of a washing machine, Resources, conservation and recycling 48 (1) (2006) 71–85.

[22] A. C. Miller III, T. R. Rice, Discrete approximations of probability distributions, Management science 29 (3) (1983) 352–362.

Appendix A.

Nomenclature

Sets

A	Set of disassembly centers
$a(n)$	Immediate predecessor of node n in the scenario tree
B	Set of bulk recycling centers
C	Set of collection zones
D	Set of remanufacturing centers
L	Set of modules
n, m	Nodes of the scenario tree
O	Set of secondary markets for spare parts
P	Set of parts
R	Set of raw materials

T Set of time periods

$Tree$ Scenario tree

W Set of secondary markets for modules

Parameters

β_q The mass of residues in the returned product with quality level q shipped to bulk recycling centers from disassembly centers

δ_{lq} The number of remanufacturable module l in the returned product with quality level q shipped to remanufacturing centers from disassembly centers

η_r The ratio of recyclable material r

γ_{pq} The number of reusable part p in the returned product with quality level q shipped to secondary markets from disassembly centers

$\psi_{cqt}(n)$ Quantity of returns with quality level q in collection zone c in period t at node n of the scenario tree

ca_{aq} Processing cost per unit of the returned product with quality level q at disassembly center a

caa_a Capacity of disassembly center a

cad_{dl} Capacity of remanufacturing center d for module l

cd_{dl} Remanufacturing cost per unit of module l at remanufacturing center d

ds_{opt} Demand for part p at spare market o in period t

dw_{wlt} Demand for module l at secondary market w in period t

fa_a Fixed cost of opening disassembly center a

fb_b Fixed cost of opening bulk recycling center b

fd_d Fixed cost of opening remanufacturing center d

hb Unit holding cost of residues in disassembly centers

hl_l Unit holding cost of module l in disassembly centers/remanufacturing centers

hp_p Unit holding cost of part p in disassembly centers

Pe_r Unit price of selling recyclable raw materials to the third-party provider

$pr(n)$ Probability of node n of the scenario tree

Ps_p Unit price of part p at spare parts markets

Pw_l Unit price of module l at module markets

ta_{ca} Shipping cost per unit of the returned product from collection zone c to disassembly center a

tb_{ab} Shipping cost per kg of residues from disassembly center a to bulk recycling center b

td_{adl} Shipping cost per unit of module l from disassembly center a to remanufacturing center d

ts_{aop} Shipping cost per unit of part p from disassembly center a to spare market o

tw_{dwl} Shipping cost per unit of module l from remanufacturing center d to secondary market w

Binary decision variables

$YA_{at}(n)$ A binary variable which is equal to one if disassembly center a is opened in period t at node n of the scenario tree and zero otherwise

$YB_{bt}(n)$ A binary variable which is equal to one if bulk recycling center b is opened in period t at node n of the scenario tree and zero otherwise

$YD_{dht}(n)$ A binary variable which is equal to one if remanufacturing center z is opened in period t at node n of the scenario tree and zero otherwise

Non-negative decision variables

$BR_{brt}(n)$ The quantity of recyclable material r purchased by the third-party provider from bulk recycling center b in period t at node n of the scenario tree

$IB_{at}(n)$ Inventory level of residues in disassembly center a by the end of period t at node n of the scenario tree

$ID_{dlt}(n)$ Inventory level of module l in remanufacturing center d by the end of period t at node n of the scenario tree

$IL_{alt}(n)$ Inventory level of module l in disassembly center a by the end of period t at node n of the scenario tree

$IP_{apt}(n)$ Inventory level of part p in disassembly center a by the end of period t at node n of the scenario tree

$QA_{caqt}(n)$ The quantity of returns with quality level q shipped from collection zone c to disassembly center a in period t at node n of the scenario tree

$QB_{abt}(n)$ The quantity of residues shipped from disassembly center a to bulk recycling center b in period t at node n of the scenario tree

$QS_{aopt}(n)$ The number of part p shipped from disassembly center a to spare parts market o in period t at node n of the scenario tree

$QW_{dwlt}(n)$ The number of module l shipped from remanufacturing center d to secondary market w in period t at node n of the scenario tree

$QD_{adlt}(n)$ The number of module l shipped from disassembly center a to remanufacturing center d in period t at node n of the scenario tree

Appendix B.

Tables B1 to B5 present a summary of parameter settings of the proposed MS-MIP model

Table B1: Parameter settings for modules

Description	Value	
	Motor	Clutch
Pw_l	150	75
hl_l	3	3

Table B2: Parameter settings for raw materials

Description	Value		
	Plastic	Steel	Copper
pe_r	1.5	1	6
η_r	0.3	0.3	0.3

Table B3: Parameter settings for parts

Type of part	Value	
	Ps_p	hp_p
Washing tube	40	1.5
Cover	10	1.5
Balance	50	1.5
Frame	10	1.5
Condenser	30	1.5
Transformer	30	1.5
Small electric	10	1.5
Hose	40	1.5
Electric wire	40	1.5
PCB board	70	1.5

Table B4: Quality level-dependent parameter settings

Parameter	Quality levels	
	High	Poor
δ_{lq}	1, 1	0, 1
γ_{pq}	1, 1, 1, 1, 1, 1, 1, 1, 1, 1	0, 0, 1, 0, 0, 0, 0, 0, 0, 1
β_q	3	30
ca_{aq}	1	2

Table B5: Other parameter settings

Description	Value	Description	Value
cb_b	2	hb	1
ds_{opt}	$\{200, 201, ..., 400\}$	dw_{wlt}	$\{200, 201, ..., 400\}$
fa_a	$Uniform(400000,\ 600000)$	fd_d	$Uniform(700000,\ 900000)$
fb_b	$Uniform(200000,\ 400000)$	cd_{dl}	3