Article

The Integrated Production-Inventory-Routing Problem with Reverse Logistics and Remanufacturing: A Two-Phase Decomposition Heuristic

Zakaria Chekoubi [1,2,*], Wajdi Trabelsi [2,3], Nathalie Sauer [2,4] and Ilias Majdouline [1]

[1] Laboratoire d'Innovation Durable et de Recherche Appliquée, Université Internationale d'Agadir—Universiapolis, Agadir 80000, Morocco
[2] Laboratoire de Génie Informatique, de Production et de Maintenance (EA 3096), 57070 Metz, France
[3] ICN Business School, 54003 Nancy, France
[4] UFR MIM, Université de Lorraine, 54000 Nancy, France
[*] Correspondence: zakaria.chekoubi@e-polytechnique.ma

Abstract: Sustainable supply chains depend on three critical decisions: production, inventory management, and distribution with reverse flows. To achieve an effective level of operational performance, policymakers must consider all these decisions, especially in Closed-Loop Supply Chains (CLSCs) with remanufacturing option. In this research paper, we address the Integrated Production-Inventory-Routing Problem with Remanufacturing (IPIRP-R) of returned End-Of-Life (EOL) products. The aim behind solving this optimization problem is to minimize conjointly the total manufacturing, remanufacturing, setup, inventory, and routing costs over the planning horizon. A two-phase decomposition heuristic is developed to solve the model iteratively. Our study finds its originality in the fact of jointly optimizing the Capacitated Lot-Sizing Problem with Remanufacturing (CLSP-R) option and the Vehicle Routing Problem with Simultaneous Pick-up and Delivery (VRPSPD) in a single framework. Numerical results showed that our solution approach provides good solutions regarding small and medium-scale size instances under acceptable computational time, especially for problems occurring with significant manufacturing and remanufacturing costs under relatively low pickup requests.

Keywords: Integrated Production-Inventory-Routing Problem; lot-sizing problem with remanufacturing; Vehicle Routing Problem (VRP); simultaneous pick-up and delivery; reverse logistics; Mixed Integer Linear Programming (MILP); decomposition heuristic

1. Introduction

In a traditional supply chain, the complexity of joint planning of production, inventory, and distribution operations, as well as the lack of information shared between stakeholders have led to dealing with these operations in a separate manner. Nowadays, companies are realizing the importance of improving their supply chains which are becoming increasingly complex. In fact, several industries, such as beverage production and the pharmaceutical industry, exhibit fluctuations in the production rate between production periods. They also need to pick up EOL products for economic and regulatory reasons. As a result, industrial decision-makers perceive the coordination and integration of production, inventory management and distribution operations as an opportunity for a competitive advantage. This advantage can be increased and sustainable if this integration is considered in the context of CLSCs. Accordingly, several companies have recently made efforts to integrate the concept of reverse logistics within their decision-making systems for several purposes such as the growing concern about the industrial impact on the environment, the restrictions of regulations and laws, government regulations on recycled products and waste disposal,

and increased energy consumption, as well as strong competition between companies. Figure 1 depicts a typical industrial example of CLSC from the beverage industry.
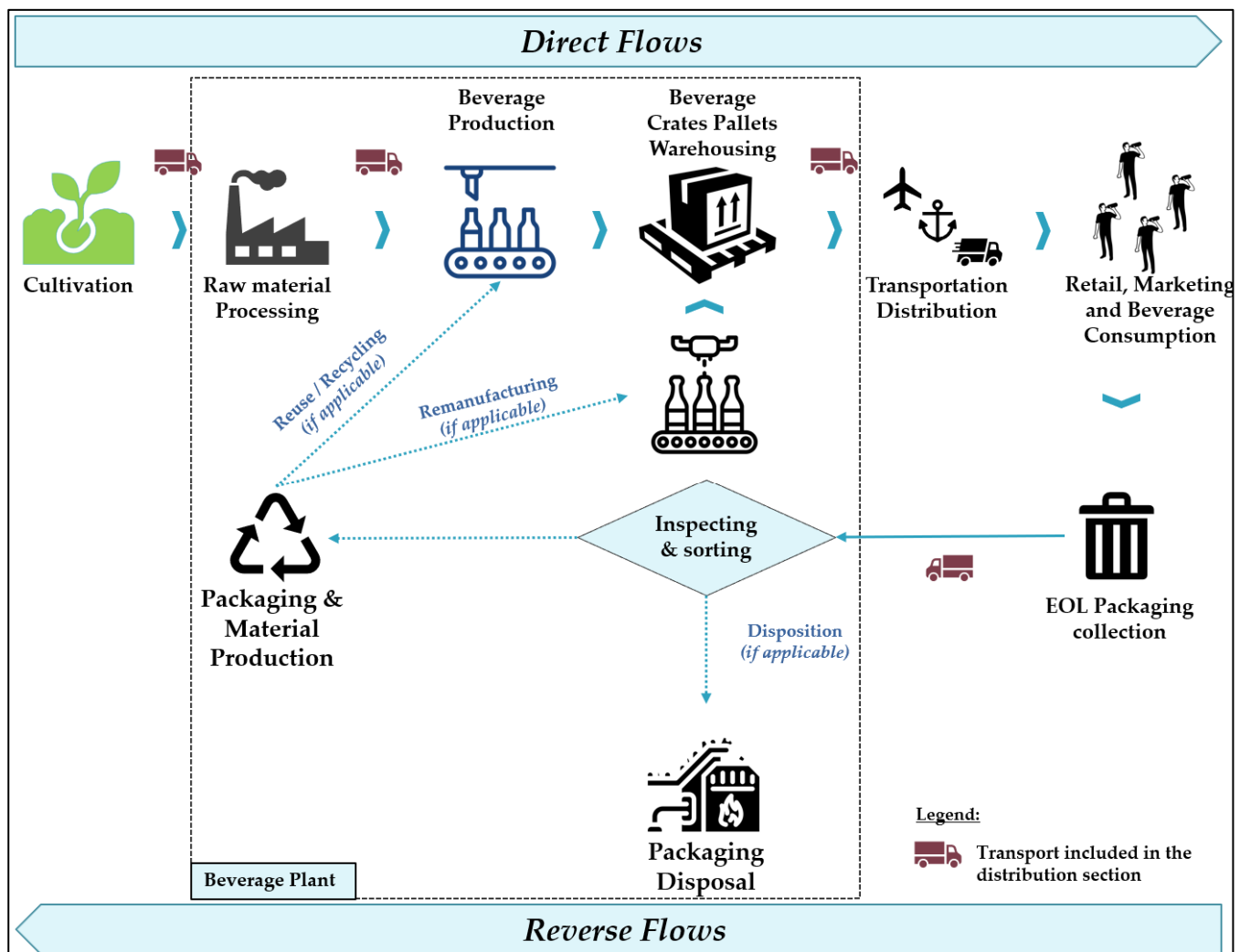

Figure 1. Typical beverage industry direct-reverse supply chain.

The integration of the production-inventory-distribution triplet in the context of reverse logistics can generate considerable benefits both economically (e.g., reduction in operational costs) and environmentally (e.g., reduction in carbon emissions). Additionally, integrating production, inventory and distribution decisions into a single problem is relevant for certain types of goods, especially perishable or time-sensitive ones. Therefore, the integration of these decision problems has attracted research interest due to the benefits provided by the coordination of these operations. However, due to the presence of return recovery activities, reverse logistics imposes new constraints on the management of the overall CLSC system. Indeed, the introduction of the concept of reverse supply chains has created many challenges with respect to logistics network design problems, transportation, used products selection, supplier selection and evaluation, yield management, remanufacturing, disassembly, etc. The main inherent difficulties that arise in managing EOL product returns include uncertainty (e.g., EOL returns quantity and quality prediction, random for demand remanufactured products, processing times . . .), balancing returns and demand, reverse distribution, etc. Therefore, the need for a returns management system seems essential. This information system can support and respond to reverse logistics implications and should be able to deal with the information relevant to each of the required

activities, such as return management, inventory management, production (manufacturing and remanufacturing) planning, and product improvement.

Among the various studies existing in the literature, IPIRP consisting of jointly optimized decisions of production, inventory, distribution and routing has recently received considerable attention [1]. In an IPIRP, single or multiple plants are responsible for producing products (single or multiple) and delivering them to multiple retailers or customers over a multi-period time horizon by jointly considering all of these decisions [2]. A solution for the IPIRP consists of deciding for each period: (1) the amount to produce at the central plant; (2) how much to deliver to each customer; (3) the amount of goods to be held at the central plant and at each customer; and (4) how to organize vehicle routes for each scheduled delivery. These decisions typically have to be made within a finite planning horizon consisting of multiple time periods. The aim is to jointly minimize production inventory, setup, and routing costs ([3,4]).

The IPIRP generalizes the Inventory Routing Problem (IRP) when considering production decisions. It can also be seen as a generalization of the classical Lost-Sizing Problem (LSP) and the VRP. Although the IPIRP is NP-hard, it can enhance synchronization, save product costs, and improve customer service levels. This has been demonstrated in numerous studies and real-world applications.

Although the fact the classical IPIRP has been the subject of in-depth research for the past decades, certain aspects of the problem have not been tackled in the recent literature. In the context of CLSCs with remanufacturing option, this problem has not, as far as we can tell, received enough attention. As part of this study, our objective is to design mathematical models and develop optimization approaches to solve the integrated planning problem of production, inventory, and direct-reverse distribution operations in the context of closed-loop supply chains with remanufacturing option. This integrated planning problem is a generalization of the classical IPIRP insofar as it additionally integrates decisions relating to the recovery and remanufacturing of EOL products. We have answered the following questions:

- How can we jointly design and optimize the integrated planning problem of production, inventory, and distribution operations with reverse logistics consideration and remanufacturing option?
- Given the complexity of his problem, would it be possible to solve the medium and large instances? up to what problem sizes can optimal/near-optima solutions be found?
- Finally, how could the aspect of remanufacturing EOL products contribute to the reduction in operational costs and consequently contribute to the economic performance of CLSCs?

Thus, the contributions of our study to the literature can be summarized as follows:

(a) It provides a variant of the classical IPIRP with direct and reverse flows as well as remanufacturing option. The direct-reverse distribution with simultaneous pickups and deliveries is now coupled with the Capacitated VRP (CVRP), which has been addressed in the recent IPIRP literature.

(b) The study offers a new mathematical formulation for the IPIRP-R problem with reverse logistics considerations. In contrast to most existing modeling approaches on the IPIRP, the IPIRP-R model covers additional costs related to remanufacturing EOL products at the total cost function level, as well as new constraints related to returns management.

(c) This study designs and implements a modified iterative decomposition heuristic inspired by [5]. To the best of our knowledge, the decomposition heuristic-based algorithm has not been adopted yet to tackle the IPIRP with remanufacturing option.

(d) The study provides extensive computational experiments to assess the efficiency and limits of the proposed solution approach. A set of randomly generated instances were used to test the proposed heuristic against cutting-edge optimization software. According to numerical results on benchmark instances, the decomposition heuristic

provides competitive solutions for the smaller instances and is capable of finding good feasible solutions in competitive computational times for the medium instances, which exceed the current capabilities of the solver.

(e) Finally, this study highlights the effects of remanufacturing parameters on the balance between manufacturing and remanufacturing operations through a sensitivity analysis and relevant management information provided. Possible industrial applications of the solutions outlined in this paper could be, but are not limited to, the production and distribution of newspapers, returnable and reusable packaging products, and beverage and perishable products industries.

Following is the outline of this paper. We provide a brief overview of related works on the IPIRP problem in Section 2. Section 3 formally describes the problem studied. Section 4 provides the mathematical formulation of the IPIRP-R problem. The developed solution approach is then presented in Section 5. Computational experiments and obtained results are provided in Section 6. In Section 7, we conclude with discussions on future research directions.

2. Related Literature

Since it was first introduced by [6], the IPIRP has received extensive research interest and finds its application in several areas and finds its application in several areas. Table 1 shows the most interesting application areas of the IPRIP occurring in the context of forward supply chains. In particular, this problem has been used extensively in the newspaper production and distribution industry, as well as in the food and perishable products industries. Furthermore, recent research works and real-life case studies have demonstrated that effective production, inventory, and routing activity coordination can increase synchronization, save product costs, and enhance service levels [7]. In comparison to the conventional sequential approach, where routing decisions are taken into account after developing the production plan, [8] notes that integrated production, inventory, and routing planning may result in savings ranging from 3% to 20% [9]. Another example illustrating the economic benefits that would be expected from effective coordination of production, inventory and distribution activities, can be found in [10] where authors indicate that the Kellogg company used the Kellogg Planning System (KPS), an integrated planning system, and was able to save $4.5 million in 1995. Moreover, by integrating these operational activities, customers can benefit from a high level of service with low stockout risk [11,12]. Despite the fact that the coordination of supply chain decisions has constantly progressed and attracted the attention of researchers, it remains a major challenge for industrials and researchers, especially in vendor-managed inventory (VMI) and Just-In-Time (JIT) systems, where it often arises [13].

After addressing the economic importance of the joint integration of production, inventory and routing decisions, a significant number of works have addressed several mathematical formulations and solving approaches for different variants of the IPIRP problem. In the literature, the IPIRP can be characterized according to the following specifications: (a) number of products (single or multiple); (b) number of plants (single or multiple); (c) production and inventory systems capacity at the plant (limited or unlimited); and (d) back-ordering (allowed or forbidden). Furthermore, most IPIRP variants addressed in the past decade have been focused on multiple plants and heterogenous capacitated vehicles [14], demand uncertainty [15,16], multi-item back-order [17], perishable products [18–20], multi-scale production [21], multiple products and outsourcing [7], sequence-dependent setup times and limited heterogeneous vehicles [15] and multi-item with a heterogenous fleet of vehicles and multiple time windows and customers deadlines [9].

Furthermore, the IPIRP can be perceived as an integration of two-level lot-sizing and VRP problems, both of which are quite difficult to solve [16]. Given the fact that the IPIRP is NP-hard [22], there are only a few exact algorithms such as branch-and-bound [23], branch-and-price [24–26], branch-and-cut [1,27], branch-and-cut guided search [13], mixed integer formulation [5,6,25], Lagrangian relaxation [27] and benders decomposition [28] available

in the literature and can solve small and medium-sized problems. Heuristics and metaheuristics are most preferred for the IPIRP, e.g., scatter search algorithm [29], approximation algorithms [12,14,16], decoupled/iterative heuristic [24,30], greedy randomized neighborhood searches [31], iterative mixed-integer programming [5], particle-swarm optimization [32], multiphase heuristic [13,26,33], ant colony optimization [34], memetic algorithm [35], genetic algorithms [36], tabu search algorithms [37,38], relax-and-fix heuristic [39] and mathematical programming heuristic [40] have been used to solve several variants of the IPIRP. In addition, the majority of these approximate methods have been tested on the benchmark instances generated by [41] and [22]. To the best of our knowledge, the recent and most successful one is the two-phase iterative heuristic proposed by [5] providing the best solutions for all available instances. Readers can refer to [37] for a recent review.

On the other hand, the fierce competition that persists today between companies as well as the growth of environmental awareness towards industrial activities have motivated companies and policymakers to collaborate and jointly improve their operational and environmental performance by incorporating the concept of reverse logistics within their regular production-inventory-distribution decision systems. This can be achieved by solving the IPIRP through CLSC. In contrast, the IPIRP has received less attention when it comes to reverse logistics, with the exception of a few research works such as [3–6,13]. According to [38], managing reverse flows of EOL products through a typical CLSC network, involves usually (1) EOL product pickup from costumers; (2) reverse logistics to return the EOL products collected; (3) screening, matching and disposal to specify the most economical reuse alternatives; (4) remanufacturing; and (5) remarketing to promote and reach new markets. Figure 2 illustrates the generic concept of a CLSC with the different key processes and steps of reverse logistics. Furthermore, refurbishment, repair or upgrade are considered to be the important processes involved in the transformation of EOL products through a remanufacturing operation [4]. In addition, the collection and remanufacturing of EOL products as practices to reduce emissions and to be environmentally friendly are best demonstrated through CLSCs ([3,4]). Furthermore, since [42] have emphasized the crucial role that remanufacturing plays in sustainable supply chains, in particular those with a closed-loop structure, several works that have addressed remanufacturing have mainly focused on inventory systems with remanufacturing [43], economic aspects of remanufacturing [44], and marketing considerations [3,4,45]. Unfortunately, routing decisions within inventory systems with remanufacturing option have received less attention.
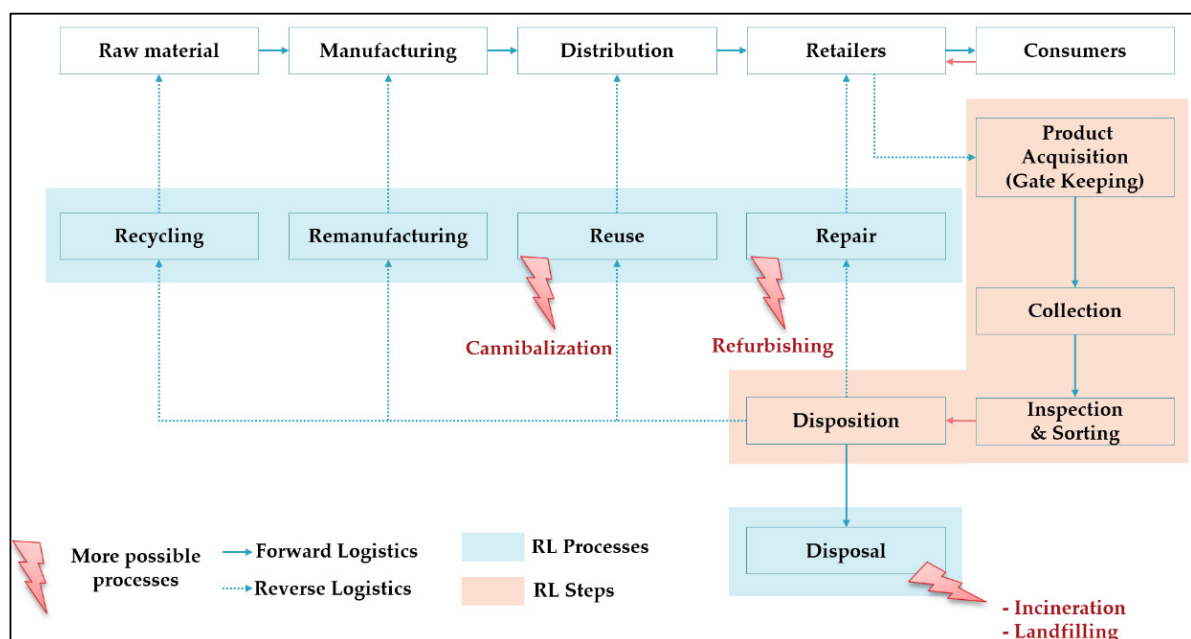

Figure 2. Basic Flows of Forward and Reverse Logistics Processes (adapted from [46]).

The IPIRP with reverse logistics, remanufacturing option and simultaneous delivery and pickup considerations, has been addressed recently by several studies [3,4,13]. The PRPRPD is the acronym given to this optimization problem, which was first presented by [13]. The PRPRPD problem has been defined as a MILP model by the authors, who have also developed a hybrid method that combines branch-and-cut with a search heuristic to find the best solution. Researchers have also paid particular attention to the PRPRPD due to its applications in the beverage, electrical appliance, and returnable/reusable transportation industries [47]. The works research of [48,49] are examples of vehicle routing issues with pickup and delivery.

Table 2 depicts the main characteristics of the selected IPIRP-related studies and compares them with the problem addressed in this article. In summary, we notice that the majority of research works related to the IPIRP literature conducted so far have focused on the optimization of this problem within classical supply chains rather than closed-loop ones. In addition, solution approaches such as decomposition heuristics, matheuristics (MIP-based heuristics) and metaheuristics are the most preferred methods used to tackle the complexity of solving IPIRP problems. In light of this, the purpose of our study is to contribute to the study of the IPIRP with remanufacturing (IPIRP-R) and bridge gaps related to the lack of studies dealing with the IPIRP with reverse logistics considerations, by extending our previous studies [5,6] since we provide a new mathematical formulation for the IPIRP with the option of remanufacturing (IPIRP-R) of returned EOL products throughout a typical CLSC network involving a fleet of homogeneous capacitated vehicles to perform simultaneous deliveries and pickups from customers.

The IPIRP-R we are studying in this article differs from the PRPRPD proposed by [13]. Indeed, authors have addressed the classic IPIRP in the context of CLSC with reverse logistics and remanufacturing considerations following a "many-to-many" direct-reverse distribution structure performed by a heterogeneous fleet of capacitated vehicles located at depots dedicated, respectively, for manufacturing new products and remanufacturing of EOL ones, and distinguish between pickup dedicated routes from those dedicated for deliveries. In addition, retailers require that their inventory levels should be managed by the supplier according to the Vendor Managed Inventory (VMI) policy. However, in our study, we have adopted a "one-to-many-to-one" direct-reverse distribution structure using a homogeneous fleet of vehicles with limited capacity available at the central plant to perform deliveries and collections simultaneously. Regarding the optimization method and given the NP-hardness of the IPIRP-R, we propose a decomposition heuristic inspired by [5] to deal with small and medium-scale size problems.

Table 1. Overview of the IPRP application areas.

Application areas		Food industry		[19,20,50,51]
		Supply Chain and logistics		[5,16,52]
	Process industries	Petrochemistry		[53]
		Gas production		[54,55]
		Furniture production		[9,39]
		Newspaper production/ distribution		[36,56–58]
		Forest production		[59,60]
	Emerging issues	Sustainable supply chains	Carbon emissions	[23,32]
			Energy efficiency	[61,62]
			Carbon emissions regulation policies	[19,20,24,63]
			Reverse logistics	[3,4,13,63–65]
		Perishability of products		[19,20,46,66]

Table 2. Summary of selected references on IPIRP.

Reference	Production			Inventory		Distribution		RL	Solution Method	
	#Plants	*#Product*	*Rem.*	*Policy*	*Cap.*	*Fleet*	*#Vehicle*		*Type*	*Approach*
[6]	Single	Multiple		ML		Hom.	Unlimited		H	Decomposition
[8]	Single	Multiple		ML		Hom.	Unlimited		H	Decomposition
[27]	Single	Multiple		ML		Hom.	Limited		H	Lagrangian relaxation
[36]	Single	Multiple				Hom.	Multiple		H	Genetic algorithms
[14]	Multiple	Single		ML	✓	Het.	Multiple		H	Decomposition
[41]	Single	Single		ML	✓	Hom.	Multiple		H	GRASP
[67]	Single	Single		ML	✓	Hom.	Multiple		H	Decomposition
[35]	Single	Single		ML	✓	Hom.	Multiple		H	Memetic
[25]	Single	Single		ML	✓	Hom.	Multiple		H	Tabu Search
[12,26]	Single	Single		ML	✓	Hom.	Multiple		H	Branch-and-price (B&P)
[68]	Single	Multiple		ML		Hom.	Multiple		H	Tabu Search
[33]	Single	Multiple		ML	✓	Hom.	Single		H	Tabu Search
[22]	Single	Single		ML/OU	✓	Hom.	Single		E/H	B&C/Decomposition iterative MIP based heuristic
[34]	Multiple	Multiple				Hom.	Multiple		H	Ant Colony Optimization (ACO)
[31]	Single	Single		ML	✓	Hom.	Multiple		H	ALNS
[1]	Single	Single		ML/OU	✓	Hom.	Multiple		E/H	B&C/ALNS
[5]	Single	Single		ML	✓	Hom.	Multiple		H	Decomposition iterative MIP based heuristic
[28]	Single	Single		ML	✓	Hom.	Multiple		E	Benders-based B&C
[29]	Single	Single		ML /OU	✓	Hom.	Multiple		H	Recherche de dispersion
[17]	Single	Multiple		ML	✓	Hom.	Single		H	Relax and Fix
[32]	Single	Single		ML	✓	Hom.	Multiple		H	Particle Swarm Optimization (PSO)
[24]	Single	Single		ML	✓	Hom.	Single		H	B&P
[21]	Multiple	Multiple		ML	✓	Hom.	Multiple		H	Decomposition iterative MIP based heuristic
[16]	Single	Single		ML	✓	Hom.	Multiple		H	Decomposition iterative MIP based heuristic
[40]	Single	Single		ML	✓	Hom.	Multiple		H	Decomposition iterative MIP based heuristic
[69]	Single	Single		ML	✓	Hom.	Multiple		H	Simulated annealing algorithm
[3]	Single	Single	✓		✓	Hom.	Single	✓	E	MILP
[13]	Multiple	Single	✓	ML	✓	Het.	Multiple	✓	E	B&C guided search
[9]	Single	Multiple				Het.	Multiple		E	Decomposition iterative MIP based heuristic

Table 2. *Cont.*

Reference	Production			Inventory		Distribution		RL	Solution Method	
	#Plants	*#Product*	*Rem.*	*Policy*	*Cap.*	*Fleet*	*#Vehicle*		*Type*	*Approach*
[39]	Single	Multiple				Hom.	Single		H	Relax and Fix
[15]	Single	Multiple				Het.	Multiple		E	MILP
[23]	Single	Single		ML	✓	Hom.	Single		E	B&B
[7]	Single	Multiple		ML	✓	Hom.	Multiple		H	Three-level decomposition MIP-based heuristic
[70]	Multiple	Multiple		ML	✓	Hom.	Multiple		H	Three-phase decomposition MIP-based heuristic
[4]	Single	Single	✓		✓	Hom.	Multiple	✓	E	MILP
Our study	Single	Single	✓		✓	Hom.	Multiple	✓	H	Two-phase decomposition MIP based heuristic

Note*:* **Rem.:** Remanufacturing; **OU:** Order-Up-to level policy; **ML:** Maximum Level policy; **Hom.:** Homogeneous; **Het.:** Heterogeneous; **RL:** Reverse Logistics; **E:** Exact; **H:** Heuristic.

3. Problem Description

In this section, a formal description of the IPIRP-R is provided.

Let $G = (N, A)$ define a complete directed graph network, where $N = \{0, 1, 2, \ldots, |N_C|\}$ is the set of nodes and $A = \{(i, j) : i, j \in N, i \neq j\}$ is the set of arcs. The central plant is denoted by node $\{0\}$ and the customers are denoted by the set $N_C = N \backslash \{0\}$. To satisfy the triangular inequality (i.e., $c_{ij} + c_{jk} \geq c_{ik}$), each $(i, j) \in A$ has a non-negative symmetric traveling cost denoted c_{ij}, which represents the cost of traveling to node j from node i. At each period $t \in T$ ($T = \{1, 2, \ldots, |T|\}$), each customer $i \in Nc$ requires a dynamic demand for delivery and pickup denoted, respectively, d_{it} and p_{it}. The delivery and pickup requirements of the central depot are supposed null ($d_{0t} = p_{0t} = 0, \forall t \in T$). Two production lines are located at the central plant for the manufacturing of serviceable products and remanufacturing of returned EOL ones. Each production line requires a separate setup cost and has a specific limited capacity. Furthermore, in order to satisfy all customer demands over the finite planning horizon, each production line must produce an economic quantity based on the single-item lot-sizing policy, either by manufacturing new products or by remanufacturing returned EOL ones or both. This implies that the remanufactured products are supposed *as new as* manufactured ones and backlogging is not allowed. We assume that returned EOL products cannot be fully remanufactured as some returned ones are beyond repair due to technical constraints. This means that remanufacturing cannot cover all returned EOL involving a remanufacturing rate $0 \leq \eta \leq 1$. In addition, a unit production cost is considered for manufacturing (resp., remanufacturing) a new product (resp., returned product) during each period. We also assume that all lead times for manufacturing and remanufacturing are extremely short. Furthermore, it is supposed that remanufacturing operations are less costly than those related to manufacturing in order to ensure manufacturing cost reduction by enhancing remanufacturing. Customers' requests for pickups and deliveries are expected to be deterministic and known throughout the entire planning horizon. In addition, we suppose that the initial inventory level of serviceables is set to a minimum safety stock. However, it is assumed that the returns inventory's initial stock level is zero (zero initial stock condition). Furthermore, a holding cost occurs for each inventory, and new manufactured (resp., remanufactured) products are stored in the serviceables (resp., returns) inventory without going over its maximum storage level. A fleet of capacitated homogeneous vehicles is available at the central plant to perform deliveries and pickups simultaneously. Moreover, we assume that customers

requiring no requests should not be visited (e.g.: $d_{it} > 0$ and $p_{it} \leq Q$), and each vehicle performs a single tour in a given time period, visiting multiple customers only once, and then returns to the central plant, making sure that its capacity Q limit is not exceeded. Split pickups and deliveries are supposed to be prohibited. The considered closed-loop network is shown in Figure 3.
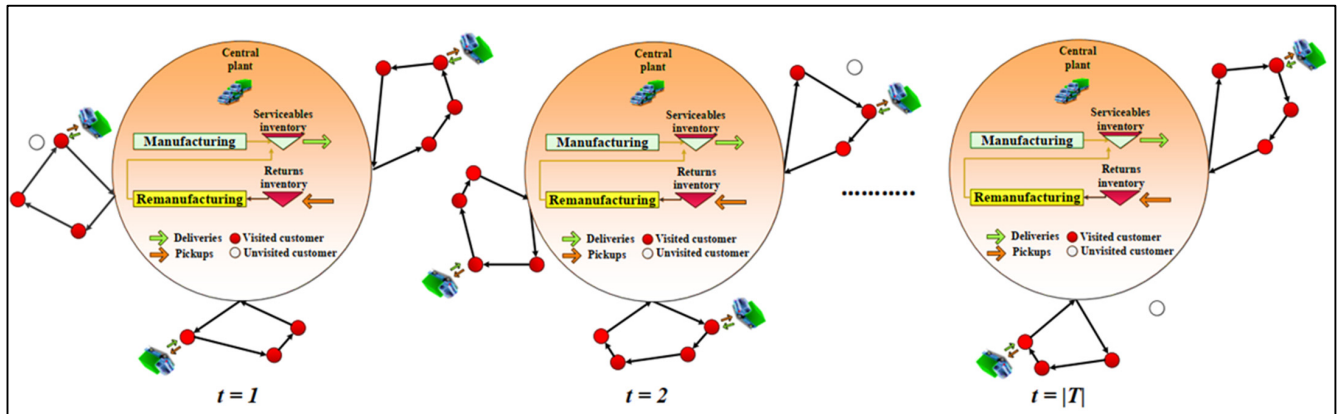

Figure 3. Illustration of the studied CLSC network.

The IPIRP-R problem seeks to find the conjoint optimal planning which minimizes the combined total cost of fixed and variable costs of manufacturing, remanufacturing, inventory, and routing operations in order to satisfy simultaneously customer demand for delivery and pickup. There are several decisions that must be made jointly for each period of the planning horizon:

1. When and how many items should be manufactured;
2. When and how many items should be remanufactured;
3. When and how many items to hold at both serviceables and returns inventories;
4. How to organize the vehicles tours visits to simultaneously perform delivery and pickup from customers.

4. Mathematical Formulation for the IPIRP-R

The main notation as well as the mathematical formulation of the IPIRP-R are provided in this section. We first introduce the following additional notation just before describing the MILP model:

(a)   Indexing sets

N:	set of nodes, with $N = \{0, 1, 2, \ldots, |N_C|\}$, where node $\{0\}$ denotes the central plant;
N_C:	set of customers, with $N_C = \{1, 2, \ldots, |N_C|\}$ is indexed by i and j;
T:	set of planning horizon periods, with $T = \{1, 2, \ldots, |T|\}$ is indexed by t;
K:	set of fleet vehicles, with $K = \{1, 2, \ldots, |K|\}$ is indexed by v.

(b)   Parameters

p^m:	unit cost of manufacturing a serviceable product;
K^m:	fixed setup cost for manufacturing;
h^m:	unit cost of holding new manufactured product at serviceables inventory;
C_t^m:	manufacturing system's maximum production rate at period t;
U_t^m:	serviceable inventory's maximum stock level at t;
p^r:	cost per unit for remanufacturing an EOL-returned product;
K^r:	fixed setup cost for remanufacturing;
h^r:	cost per stocking unit of returned EOL products in returns inventory;
C_t^r:	remanufacturing system's maximum production rate at period t;
U_t^r:	returns inventory's maximum stock level at period t;

I_0^m:	initial stock level of serviceables inventory;
I_0^r:	initial stock level of returns inventory;
d_{it}:	delivery demand of customer $i \in N_C$ at period t;
p_{it}:	pickup demand of customer $i \in N_C$ at period t;
η:	remanufacturing rate satisfying $0 \leq \eta \leq 1$;
Q:	capacity of each vehicle;
fc:	fixed cost of using a vehicle;
c_{ij}:	transportation costs over arc $(i, j) \in A$ (assume $c_{ij} = c_{ji}$ and $c_{ii} = 0$).

(c) Decision variables

x_t^m:	amount of new products manufactured in period t;
δ_t^m:	0–1 variable which equals 1, if manufacturing occurs at period t ($x_t^m > 0$), 0 otherwise;
I_t^m:	serviceable inventory's level at the end of period t;
x_t^r:	amount of remanufactured products in period t;
δ_t^r:	0–1 variable which equals 1, if remanufacturing occurs at period t ($x_t^r > 0$), 0 otherwise;
I_t^r:	returns inventory's level at the end of period t;
z_{ijt}:	demand delivered to node i and transported in arc (i, j) at period t;
w_{ijt}:	demand collected from node i and transported in arc (i, j) at period t;
y_{ijvt}:	binary variable which is 1 or 0 depending on whether vehicle v reaches node j after node i during period t.

(d) Objective function and constraints

The integrated model IPIRP-R can be formulated using the parameters and variables previously introduced as follows:

$$\text{(IPIRP} - \text{R)}: \quad \min z = \sum_{t \in T} (K^m . \delta_t^m + p^m . x_t^m + h^m . I_t^m + K^r . \delta_t^r + p^r . x_t^r + h^r . I_t^r) +$$

$$\sum_{j \in N_C} \sum_{v \in K} \sum_{t \in T} fc . y_{0jvt} + \sum_{\substack{i \in N}} \sum_{\substack{j \in N \\ j \neq i}} \sum_{v \in K} \sum_{t \in T} c_{ij} . y_{ijvt} \tag{1}$$

subject to

$$I_t^m = I_{t-1}^m + x_t^m + x_t^r - \sum_{j \in N_C} z_{0jt}; \ \forall t \in T \tag{2}$$

$$I_t^r = I_{t-1}^r - x_t^r / \eta + \sum_{i \in N_C} w_{i0t}; \ \forall t \in T \tag{3}$$

$$0 \leq I_t^m \leq U_t^m; \ \forall t \in T \tag{4}$$

$$0 \leq I_t^r \leq U_t^r; \ \forall t \in T \tag{5}$$

$$x_t^m \leq \min\left\{ \sum_{i \in N_C} \sum_{l=t}^{T} d_{il}, C_t^m \right\} . \delta_t^m; \ \forall t \in T \tag{6}$$

$$x_t^r \leq min\left\{ \min\left\{ \sum_{i \in N_C} \sum_{l=t}^{T} d_{il}, C_t^r \right\} . \delta_t^r, \eta . I_{t-1}^r \right\}; \ \forall t \in T \tag{7}$$

$$\sum_{j \in N_C} y_{0ivt} \leq 1; \ \forall v \in K, \ \forall t \in T \tag{8}$$

$$\sum_{i \in N_C} y_{i0vt} \leq 1; \ \forall v \in K, \ \forall t \in T \tag{9}$$

$$\sum_{\substack{i \in N \\ i \neq j}} \sum_{v \in K} y_{ijvt} \leq 1; \ \forall j \in N_C, \ \forall t \in T \tag{10}$$

$$\sum_{\substack{j \in N \\ j \neq i}} y_{ijvt} - \sum_{\substack{j \in N \\ j \neq i}} y_{jivt} = 0; \ \forall i \in N_C, \ \forall v \in K, \ \forall t \in T \tag{11}$$

$$(z_{ijt} + w_{ijt}) \leq Q . \sum_{v \in K} y_{ijvt}; \ \forall (i, j : i \neq j) \in N, \ \forall t \in T \tag{12}$$

$$\sum_{\substack{i \in N \\ i \neq j}} z_{ijt} - \sum_{\substack{i \in N \\ i \neq j}} z_{jit} = d_{jt}; \; \forall j \in N_C, \forall t \in T \tag{13}$$

$$\sum_{\substack{i \in N \\ i \neq j}} w_{jit} - \sum_{\substack{i \in N \\ i \neq j}} w_{ijt} = p_{jt}; \; \forall j \in N_C, \forall t \in T \tag{14}$$

$$w_{0jt} = 0; \; \forall j \in N_C, \forall t \in T \tag{15}$$

$$z_{i0t} = 0; \; \forall i \in N_C, \forall t \in T \tag{16}$$

$$y_{ijvt} \in \{0, 1\}; \; \forall \, (i,j : \, i \neq j) \in N, \; \forall v \in K, \forall t \in T \tag{17}$$

$$x_t^m \in \mathbb{Z}^+, \; x_t^r \in \mathbb{Z}^+, \; I_t^m \in \mathbb{Z}^+, \; I_t^r \in \mathbb{Z}^+, \; \delta_t^m \in \{0, 1\}, \; \delta_t^r \in \{0, 1\}; \; \forall t \in T \tag{18}$$

$$z_{ijt} \in \mathbb{Z}^+, \; w_{ijt} \in \mathbb{Z}^+; \; \forall \, (i,j : \, i \neq j) \in N, \; \forall t \in T \tag{19}$$

Objective function (1) seeks to minimize the total manufacturing, remanufacturing, setup, inventory, and routing costs. The inventory flow balance equations are represented by the constraints (2)–(3). Inequalities (4)–(5) limit, respectively, the inventory level of serviceables and returns inventories without exceeding their maximum capacity at the end of each period. The manufacturing and remanufacturing capacity constraints are represented by the inequalities (6)–(7), which ensure that a manufacturing operation would only take place in a period t if the manufacturing production system (or the remanufacturing production system) is set up at the beginning of the period t. The vehicle routing constraints are indicated by constraints (8)–(11). Constraint (10) ensures that the vehicle capacity should not be exceeded by the transported total load. Notably, the combination of constraints (10) and (12) guarantees that if a customer's requests for deliveries and pickups are not null, then that customer should be visited. Equations (13) and (14) ensure that customer requests, respectively, for delivery and collection are fulfilled. The vehicle must leave the central plant with no pickup loads and return after performing all deliveries, as required by constraints (15) and (16). Finally, constraints (17)–(19) provide the domain of the variables.

From Equations (1)–(19) described above, we can easily notice that the integrated model IPIRP-R is similar to the classic IPIRP with an additional structure of costs related to the remanufacturing operation as well as constraints related to the pickup and holding of EOL products. Because the IPIRP-R generalizes the classical IPIRP, it follows that the IPIRP-R is also *NP-hard* [13]. Given the fact that the IPIRP-R model includes a considerable number of decision variables and constraints, which explain the long-running execution times that this model took on medium-sized instances, it seemed logical to us to lean towards approximate optimization methods based on heuristics (see Section 5). In fact, the literature review carried out has revealed that heuristics based on mathematical models (matheuristics) have been widely used to solve this type of optimization problem. In particular, Iterative MILP-based heuristics have demonstrated their performance and flexibility when solving the classical IPIRP. This type of method often combines heuristics with exact approaches to turn the original problem into easier-to-solve sub-problems, while trying to lose as little information about the original problem as possible.

Note that jointly optimizing production, inventory, and distribution operations in the context of reverse logistics with the remanufacturing option of EOL products could have a positive impact on how well the economy and environment perform in the considered closed supply chain network. Indeed, we believe that the recovery of the residual value of EOL products through remanufacturing process could contribute to minimizing the total integrated cost of the involved operations, especially since we assume that the fixed and variable costs of remanufacturing activities are less costly than manufacturing ones given the fact that customer requests can be satisfied either by new or remanufactured products or both. This means that remanufacturing activities can reduce production costs as long as manufacturing activities can be replaced by remanufacturing ones given there are enough

EOL products to pick up at customer locations and to hold in the returns inventory. Furthermore, it may be possible to cut CO_2 emissions by collecting and remanufacturing returned EOL products and therefore contribute to improving the environmental performance of the closed-loop logistics network taking into account product lifecycle management after exceeding its lifespan.

5. Solution Approach

This section deals with the description of the approximate optimization method developed to solve the IPIRP-R problem. Thus, a modified two-phase iterative decomposition heuristic (denoted by *CST*) inspired by the work of [5] is proposed to solve small and medium-sized problems and obtain good quality solutions in terms of total integrated cost within reasonable computational time.

5.1. Overview

To solve the IPIRP-R, we consider a method based on some feature ideas similar to [5] who proposed a two-phase iterative method (IM) to solve the classic IPIRP. The idea is to split the IPIRP-R into two smaller sub-problems which are solved iteratively until no better solution is possible for a given stopping criterion (e.g., number of iterations) is reached.

The first phase addresses the capacitated single-item lot-sizing problem with remanufacturing, denoted CLSP-R, where the IPIRP-R model is restricted to the manufacturing, remanufacturing, and inventory management decisions. This model decides when and how many new products to manufacture and to hold, when and how many EOL products to remanufacture and to hold, how many new products to deliver and how many EOL ones to pick up simultaneously from visited customers. In addition, the model also allocates customers to vehicles without considering explicitly routing decisions by using approximate visiting costs SC_{ivt} which are updated throughout the algorithm. The capacity constraints on production and inventory systems as well as the capacity constraint of each vehicle are considered in this phase. The objective of the CLSP-R model is to minimize the total cost of manufacturing, remanufacturing, setup, holding and the approximate visiting costs. In the second phase, we optimize the routing decisions for the used vehicles in each period vehicle by solving a restricted version of the IPIRP-R model namely the VRPSPD taking into consideration the customers allocated for each vehicle arising from the first phase.

The method is started by setting the current solution as an empty solution and the best solution found as infinity. The value of SC_{ivt} is initially set to $c_{0i} + c_{i0}$ (Algorithm 1, line 3). Then, the first phase minimizes the restricted CLSP-R model using the initialized approximate visiting cost SC_{ivt} ($\forall\, i \in N_C$, $\forall v \in K$, $\forall t \in T$) instead of transportation cost c_{ij}.

Note that the approximate visiting cost SC_{ivt} has a crucial role as it creates a connection between the first and second phases. Moreover, this cost is updated in each iteration of the method using the information provided by the solution of the second phase so that solutions of the first phase are driven to better solutions in terms of customer clustering. Whenever a feasible solution is found, it is then used to update the approximate visiting costs for the next iteration of the CLSP-R model. This iterative procedure stops whenever the overall solution is not improved for a given number of iterations or the execution time limit is exceeded. When the iterative procedure stops, a diversification mechanism is applied and the whole process will be repeated until the number of restarts is reached.

Algorithm 1 shows the main steps of our decomposition heuristic method, and the sections that follow provide more specific descriptions of each step.

Algorithm 1. Optimization method—A Decomposition heuristic for the IPIRP-R

Inputs: *Nbmax_Iter*; *Nbmax_Rest*
Output: *Best_Sol*
1: // Parameters Initialization
2: $Sol \leftarrow \varnothing$; $Best\_Sol \leftarrow +\infty$; $Restart \leftarrow 0$; $No\_Improv \leftarrow 0$;
3: $SC_{ivt} = c_{0i} + c_{i0}$, $\forall i \in N_C$; $\forall v \in K$; $\forall t \in T$
4: **Repeat**
5:   **Repeat**
6:     Solve the restricted **CLSP-R** model and get γ_{ivt}, $\forall i \in N_C$, $\forall v \in K$, $\forall t \in T$;
7:     Solve the restricted **VRPSPD** model for all $v \in K, t \in T, \exists i \in N_C$ such as: γ_{ivt}=1
8:       // Update the solution and parameters if necessary
9:       **If** $Sol < Best\_sol$ **then**
10:         $Best_{Sol} \leftarrow Sol$;
11:         $No\_Improv \leftarrow 0$;
12:       **Else**
13:         $No\_Improv \leftarrow No\_Improv + 1$;
14:       **End If**
15:     Update approximate visiting costs SC_{ivt}, $\forall i \in N_C$, $\forall v \in K$, $\forall t \in T$ using **Algorithm 2**;
16:   **Until** $No\_Improv \geq Nbmax\_Iter$
17:     *Diversification*;
18:     $Restart \leftarrow Restart + 1$;
19: **Until** $Restat \geq Nbmax\_Rest$
20: **Return** *Best_Sol*

5.2. First Phase: A Restricted Capacitated Lot-Sizing Problem with Remanufacturing Model

In this section, we describe the mathematical model used in the first phase of our heuristic approach. As already mentioned, this phase decides when and how many new products to manufacture, when and how many EOL products to remanufacture, when to visit customers, how many new products to deliver and how many EOL products to pick up simultaneously by solving the restricted CLSP-R model. In this model the parameter SC_{ivt} represents the cost of visiting customer $i \in N_C$ at period t with vehicle v. The aim of the objective function is to minimize the total integrated cost related to manufacturing (resp. remanufacturing) of new products (reps. re-manufactured products), to setup (manufacturing and remanufacturing) and to inventory holding of serviceables products (resp. returned products) as well as costs related to inserting customers into vehicle routes.

To formulate the CLSP-R model (Algorithm 1, line 6), the following additional parameter and decision variables are defined:

(a)   Parameter

SC_{ivt}: approximate visiting cost for serving customer i from the central plant by vehicle v in period t;

(a)   Decision variables:

ξ_{vt}:   binary variable which equals 1, if vehicle v is selected to be used in period t, 0 otherwise;
γ_{ivt}:   binary variable which equals 1, if customer i is served by vehicle v in period t, 0 otherwise;
a_{ivt}:   quantity of new products delivered to customer i visited by the vehicle v in period t;
b_{ivt}:   quantity of returned products collected from customer i visited by the vehicle v in period t;

The CLSP-R restricted model can be described as follows:

$$(\text{CLSP} - \text{R}): \quad min\ z = \sum_{t=1}^{|T|} \left(K^m.\delta_t^m + p^m.x_t^m + h^m.I_t^m + K^r.\delta_t^r + p^r.x_t^r + h^r.I_t^r \right)$$
$$+ \sum_{i \in N_C} \sum_{v=1}^{|K|} \sum_{t=1}^{|T|} SC_{ivt}.\gamma_{ivt} + \sum_{v=1}^{|K|} \sum_{t=1}^{|T|} \xi_{vt} \tag{20}$$

subject to

- *Maximum inventory capacity limits constraints (4)–(5);*
- *Manufacturing and remanufacturing capacity constraints (6)–(7);*

subject to

$$I_t^m = I_{t-1}^m + x_t^m + x_t^r - \sum_{i \in N_C} \sum_{v \in K} a_{ivt}; \ \forall t \in T \tag{21}$$

$$I_t^r = I_{t-1}^r - x_t^r/\eta + \sum_{i \in N_C} \sum_{v \in K} b_{ivt}; \ \forall t \in T \tag{22}$$

$$\sum_{v \in K} a_{ivt} = d_{it}; \ \forall i \in N_C, \ \forall t \in T \tag{23}$$

$$\sum_{v \in K} b_{ivt} = p_{it}; \ \forall i \in N_C, \ \forall t \in T \tag{24}$$

$$\sum_{i \in N_C} a_{ivt} \le Q; \ \forall v \in K, \ \forall t \in T \tag{25}$$

$$\sum_{i \in N_C} b_{ivt} \le Q; \ \forall v \in K, \ \forall t \in T \tag{26}$$

$$a_{ivt} \le Q.\gamma_{ivt}; \ \forall i \in N_C, \ \forall v \in K, \ \forall t \in T \tag{27}$$

$$b_{ivt} \le Q.\gamma_{ivt}; \ \forall i \in N_C, \ \forall v \in K, \ \forall t \in T \tag{28}$$

$$\sum_{v \in K} \gamma_{ivt} \le 1; \ \forall i \in N_C, \ \forall t \in T \tag{29}$$

$$\gamma_{ivt} \le \xi_{vt}; \ \forall i \in N_C, \ \forall v \in K, \ \forall t \in T \tag{30}$$

$$\gamma_{ivt} \in \{0, 1\}; \ \forall i \in N_C, \ \forall v \in K, \ \forall t \in T \tag{31}$$

$$\xi_{vt} \in \{0, 1\}; \ \forall v \in K, \ \forall t \in T \tag{32}$$

$$a_{ivt} \in \mathbb{Z}^+ \tag{33}$$

$$b_{ivt} \in \mathbb{Z}^+ \tag{34}$$

and (18).

The objective function (20) seeks to minimize the total integrated manufacturing, remanufacturing, setup, inventory and visit costs as well as the number of vehicles to be used. The inventory flow balance at the serviceable inventory and returns inventory is guaranteed, respectively, by constraints (21) and (22). Equation (23) ensures that the amount of quantities delivered using selected vehicles corresponds to the delivery request of the visited customer. Constraint (24) ensures that the amount of quantities collected from the visited customer using selected vehicles corresponds to his pickup request. Constraints (25) and (26) guarantee that the capacity of the vehicles is not exceeded whatsoever by the quantities to be delivered or collected, respectively. Constraints (27) and (28) guarantee that the amount of delivery goods (reps. pickup amount) is equal to 0 if no customer is visited. Constraint (29) prevents split deliveries and pickups. Constraint (30) indicates that a customer is visited by a vehicle in a given period, only if this vehicle is used in the same period. Finally, constraints (31)–(34) bound the ranges of the decision variables.

Note that solving the CLSP-R model without estimating visiting costs would result in a solution entirely driven by manufacturing or remanufacturing costs, for which it may be quite challenging to find a feasible routing plan. It is therefore a matter of roughly considering transport costs to find solutions with a better compromise between manufacturing (resp., remanufacturing) and transport costs. Consequently, the CLSP-R model is aware that routing decisions are significant when determining the total cost even though it does not explicitly reflect routing decisions [3,33].

The second phase will use the assignment of customers per vehicle from the first phase to determine the delivery-collection routing program for customers served in each period by each vehicle.

5.3. Second Phase: A Restricted VRP with Simultaneous Pickup and Delivery MODEL

In this section, the routing part with simultaneous delivery and pickup of our heuristic approach will be described. Therefore, in virtue of the decisions made in the first phase, the routing problem seeks to solve the constrained VRPSPD model. The second phase's

objective consists in finding the optimal routes for the direct-reverse distribution of a set of customers served by a group of vehicles during each period.

Let $S_{vt} = \{i \in N_C : \gamma_{ivt} = 1\}$ represent the cluster of customers i assigned to vehicle v in period t from the first phase. For all vehicles, we solve the following restricted VRPSPD formulation (Algorithm 1, line 7):

$$\text{(VRPSPD)} : \min z = \sum_{j \in S_{ivt}} \sum_{v \in K} \sum_{t \in T} fc.y_{0jvt} + \sum_{\substack{i \in S_{ivt} \cup \{0\} \\ j \neq i}} \sum_{j \in S_{jvt} \cup \{0\}} \sum_{v \in K} \sum_{t \in T} c_{ij}.y_{ijvt} \tag{35}$$

subject to

$$\sum_{j \in S_{jvt}} y_{0jvt} \leq 1; \; \forall v \in K, \; \forall t \in T \tag{36}$$

$$\sum_{i \in S_{ivt}} y_{i0vt} \leq 1; \; \forall v \in K, \; \forall t \in T \tag{37}$$

$$\sum_{\substack{i \in S_{ivt} \cup \{0\} \\ i \neq j}} \sum_{v \in K} y_{ijvt} \leq 1; \; \forall j \in S_{jvt}, \; \forall t \in T \tag{38}$$

$$\sum_{\substack{j \in S_{jt} \cup \{0\} \\ j \neq i}} y_{ijvt} - \sum_{\substack{j \in S_{jt} \cup \{0\} \\ j \neq i}} y_{jivt} = 0; \; \forall i \in S_{ivt}, \; \forall v \in K, \; \forall t \in T \tag{39}$$

$$z_{ijt} + w_{ijt} \leq Q. \sum_{v \in K} y_{ijvt}; \; \forall(i,j : i \neq j) \in S_{ivt} \cup \{0\}, \; \forall v \in K, \; \forall t \in T \tag{40}$$

$$\sum_{\substack{i \in S_{ivt} \cup \{0\} \\ i \neq j}} z_{ijt} - \sum_{\substack{i \in S_{ivt} \cup \{0\} \\ i \neq j}} z_{jit} = d_{jt}; \; \forall j \in S_{jvt}, \; \forall t \in T \tag{41}$$

$$\sum_{\substack{i \in S_{ivt} \cup \{0\} \\ i \neq j}} w_{jit} - \sum_{\substack{i \in S_{ivt} \cup \{0\} \\ i \neq j}} w_{ijt} = p_{jt}; \; \forall j \in S_{jvt}, \; \forall t \in T \tag{42}$$

$$w_{0jt} = 0; \; \forall j \in S_{jvt}, \; \forall t \in T \tag{43}$$

$$z_{i0t} = 0; \; \forall i \in S_{ivt}, \; \forall t \in T \tag{44}$$

$$y_{ijvt} \in \{0, 1\}; \; \forall(i,j : i \neq j) \in S_{ivt} \cup \{0\}, \; \forall v \in K, \; \forall t \in T \tag{45}$$

$$z_{ijt} \in \mathbb{Z}^+, \; w_{ijt} \in \mathbb{Z}^+; \; \forall(i,j : i \neq j) \in S_{ivt} \cup \{0\}, \; \forall t \in T \tag{46}$$

Objective function (35) seeks to minimize the total fixed and variable routing costs. Constraints (36) and (37) ensure that each vehicle must leave and return to the central depot, respectively. Constraint (38) denotes that each customer is visited once. Constraints (39) guarantee flow conservation. Constraints (40) guarantee the vehicle's capacity. Constraints (41) and (42) impose that customers' requests need to be fulfilled. Constraints (43) and (44) ensure that the vehicle leaves the central plant with no pickup load and returns after fulfilling all delivery demands. Domain constraints for decision variables are provided via constraints (45) and (46).

5.4. Updating Visiting Costs SC_{ivt}

As mentioned earlier, the approximate visit costs play an important role as they allow the connection between the first and the second phase. This section outlines the steps involved in upgrading these costs.

Let R_{vt} the route assigned to vehicle v at period t, provided by solving the second phase. R_{vt} is defined on set $S_{vt} \cup \{0\}$. Let i^+ and i^- represent, respectively, the predecessor and successor of customer i in route $r \in R_{vt}$ for $v \in V, t \in T$ and $i \in S_{vt}$.

Likewise, for $v \in V, t \in T$ and $i \notin S_{vt}$, Δ_{ivt} is the least expensive cost to insert customer i in route r. Algorithm 2, based on [5], shows how to update the approximate visiting cost SC_{ivt} (Algorithm 1, line 15).

Algorithm 2. Update approximate visit costs S_{ivt}

1: For all $v \in K$ and $t \in T$ **do**
2:     **For all** $\forall i \in N_C$ **do**
3:         **If** $i \in S_{vt}$ **then**
4:             $SC_{ivt} \leftarrow (c_{i,i^-}) + (c_{i,i^+}) - (c_{i^-,i^+})$
5:         **Else**
6:            $SC_{ivt} \leftarrow \Delta_{ivt}$
7:         **End If**
8:     **End For**
9: **End For**

5.5. Diversification Mechanism

To guide the first phase to explore a new space of new solutions, the approximate visit costs should be diversified. Through this section, the mechanism of diversification of approximate visit costs is presented.

A feasible solution for the IPIRP-R problem is guaranteed whenever the LP solver finds feasible routes for all subproblems in phase 2. Afterwards, whenever the best solution already found has not been improved, we use a multi-start procedure to restart the whole procedure (Algorithm 1, line 17). The idea is to restart iteratively the process by randomly generating approximate visiting cost SC_{ivt} by multiplying it by a random number ρ_{ivt}. Its value is chosen in the interval [0.5, 1.5] and the approximate visiting cost SC_{ivt} is set to $\rho vit \times (c_{0i} + c_{i0})$. The stopping criterion of this mechanism is met when a predetermined number of restarts is reached (Algorithm 1, line 19).

6. Computational Study

In this section, we compare the obtained results from CPLEX and our *CST* heuristic by performing in-depth computational experiments on the generated instances described in Section 5.1. All mathematical formulations were built in Java Eclipse utilizing Concert Technology and solved by CPLEX 12.7 using the default parameters on a 64-bit Windows 10 Professional PC with a processor Intel® CoreTM i7-4790 CPU 2.9 GHz and 12 G RAM. In the following subsections, instances generation and numerical results are presented.

6.1. Instances Generation

In order to generate instances for the IPIRP-R, we use a similar test bed to [4] and adapt some data sets from the ideas of [13,22], along with the parameters related to remanufacturing and pickup requests. All the tests were conducted on three classes of randomly generated problem instances.

Test instances of the first class (standard class) were generated according to the parameters mentioned in Table 3.

Table 3. Parameters used for the generation of instances of the standard class.

Parameter	Possible Values
Time horizon $\|T\|$	$\{3, 6\}$
Number of costumers $\|N_C\|$	$5k$, with $k = \{1, 2, 3, 4, 7, 10\}$
Delivery demand of customer $i \in N_C$ at period $t \in T$, d_{it}	Integer number randomly generated in the range [5, 30]
Pickup demand of customer $i \in N_C$ at period $t \in T$, p_{it}	- Low pickups scenario: $p_{it} \leq \left(\frac{d_{it}}{2}\right)$ - High pickups scenario: $d_{it}/2 \leq p_{it} \leq (d_{it} * 3/4)$
Manufacturing system's maximum production rate at period $t \in T$, C_t^m	$2 \cdot \dfrac{\sum_{t \in T} \sum_{i \in N_C} d_{it}}{T}$
Remanufacturing system's maximum production rate at period $t \in T$, C_t^r	$2 \cdot \dfrac{\sum_{t \in T} \sum_{i \in N_C} p_{it}}{T}$

Table 3. *Cont.*

Parameter	Possible Values
Serviceable inventory's maximum stock level $t \in T$, U_t^m	$2 \cdot \dfrac{\sum_{t \in T} \sum_{i \in N_C} d_{it}}{T} \cdot \lvert N_C \rvert$
Initial stock level of serviceables inventory, I_0^m	$\left\lceil \dfrac{\sum_{t \in T} \sum_{i \in N_C} d_{it}}{T} \right\rceil$
Initial stock level of returns inventory	0
Unit cost of manufacturing a serviceable product, p^m	$10h^m$
Unit cost of holding new manufactured product at serviceables inventory, h^m	8
Unitary production cost of a new remanufactured product, p^r	$\sigma \times p^m$ where $\sigma = \{0.1,\ 0.3,\ 0.5,\ 0.7,\ 0.9\}$
Unitary inventory holding cost for return product, h^r	$h^r = h^m$
Fixed manufacturing setup cost, K^m	$100p^m$
Fixed remanufacturing setup cost, K^r	$0.01K^m$
Remanufacturing rate, η	$\eta = 0,9$
Transportation cost, c_{ij}	$\left\lfloor \sqrt{\left(X_i - X_j\right)^2 + \left(Y_i - Y_j\right)^2 + 0.5} \right\rfloor$, where each coordinate is randomly generated as an integer number in the interval [0, 500] to obtain the points $\left(X_i,\ Y_i\right)$ and $\left(X_j,\ Y_j\right)$
Transportation capacity, Q	100
Vehicle fixed cost for using a vehicle, fc	500
Maximum number of vehicles over the planning horizon, $\lvert K \rvert_{max}$	$\left\lceil \dfrac{\max\left(\sum_{t \in T} \sum_{i \in N_C} d_{it};\sum_{t \in T} \sum_{i \in N_C} p_{it}\right)}{Q} \right\rceil$

The only difference between the second class of instances and the first one is that the parameter p^m is changed from $10h^m$ to $100h^m$. This class mimics scenarios where manufacturing and remanufacturing production system costs have a greater impact than inventory costs. Finally, the coordinates of the consumers and central plant are multiplied by a factor of 10 in the third class, which is otherwise identical to the first. In this class, scenarios with high transportation costs are examined. In all cases, random selections have been drawn using a uniform distribution.

A total of 20 combinations are generated for a given number of customers $\lvert N_C \rvert$ based on the various parameter values: a time horizon with two values for $\lvert T \rvert = \{3, 6\}$, two pickup scenario possibilities {Low, High}, and a unit remanufacturing cost rate with five different values $\sigma = \{0.1,\ 0.3,\ 0.5,\ 0.7,\ 0.9\}$. Four instances with different requests and node coordinates were generated for each combination of these parameters, totaling $20 * 4 * 6 = 480$ instances for each instance type.

6.2. Algorithm Parameters Setting

The values of the parameters impacting the performance of the algorithm must be carefully chosen. To do so, we have adjusted the values of the parameters *Nbmax_Iter* (maximum number of iterations) and *Nbmax_Rest* (maximum number of restarts), where $NB\_Iterations = Nbmax\_Iter \times Nbmax\_Rest$ (total number iterations). We firstly adjusted these two parameters simultaneously and a second time we fixed the parameter *Nbmax_Iter* and we varied the parameter *Nbmax_Rest*. Figure 4 shows the simulated results in terms of the percentage of the average deviation from the solution of CPLEX ($Gap\%$) and the average execution time of the algorithm in seconds ($CPU(s)$) for problems with 15 customers and 3 periods over the three classes of data sets. Note that the average results were obtained based on four instances. Furthermore, the average $Gap\%$ was calculated according to the best solutions obtained from the algorithm compared with the upper bounds obtained by CPLEX in a certain amount of time. Note that whenever our

CST heuristic finds better solutions than the upper bounds found by CPLEX, the *Gap*% will be negative.
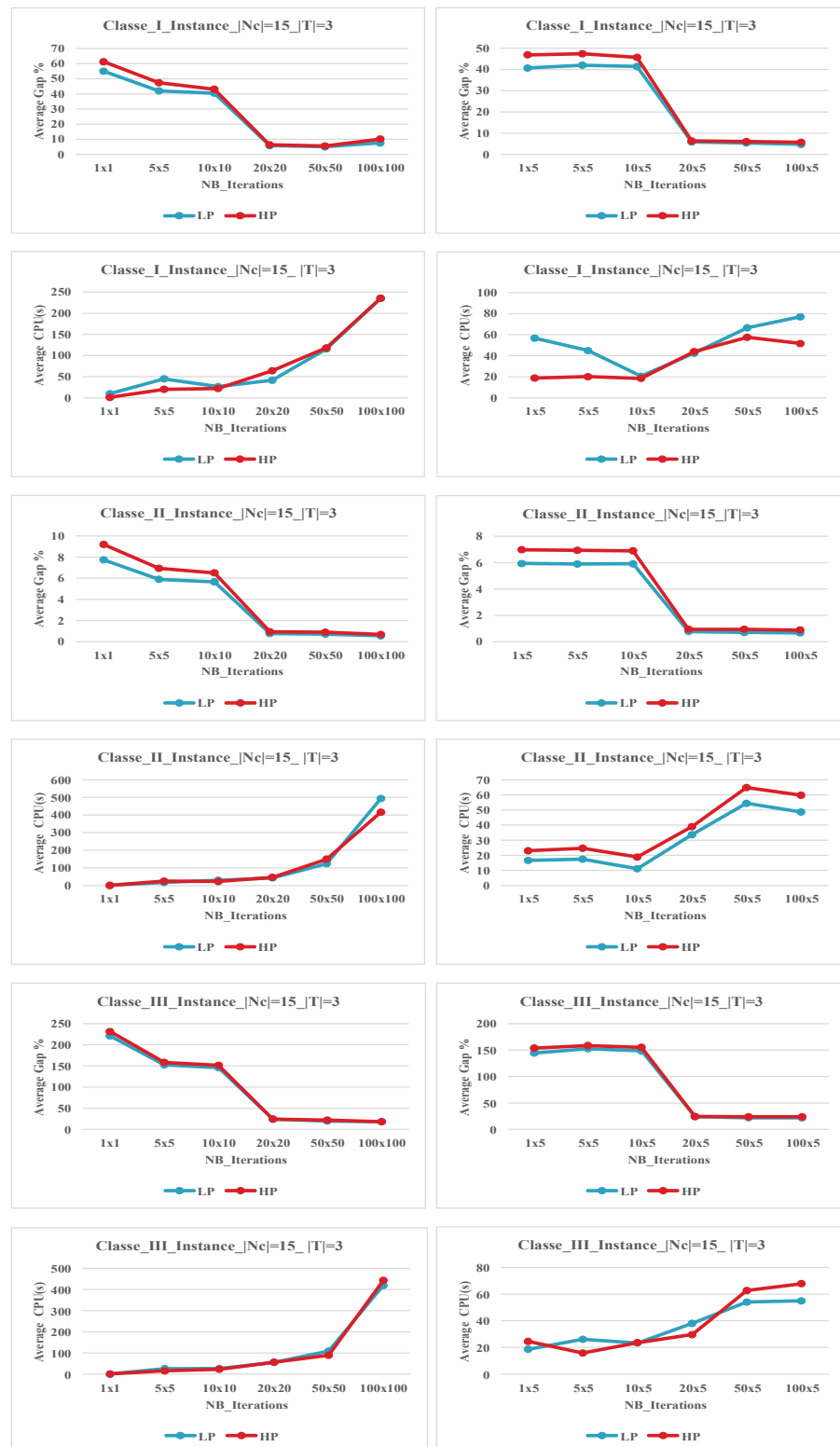

Figure 4. Sensitivity analysis of the performance of the proposed algorithm with respect to parameters *Nbmax_Iter* and *Nbmax_Rest* on problems with $|N_C| = 15$ and $|T| = 3$: three classes of data sets.

Obviously, adjusting the values of *Nbmax_Iter* and *Nbmax_Rest* has an important effect on the performance of the algorithm. For the case, we have adjusted the two parameters at the same time, we can notice that average *Gap*% decreases drastically when $NB\_Iterations = 50 \times 50$ for all cases with an average execution time $CPU(s)$ around 100 s. We observe the same results for the second scenario where we fix the first parameter and adjust the second. In fact, as can be seen from $NB\_Iterations = 50 \times 5$ the average *Gap*% drops enormously with an average execution time around 60 s. Moreover, to obtain good quality solutions, we have opted to set a number of iterations of $NB\_Iterations = 100 \times 5$ when solving all instances.

6.3. Computational Results

In order to assess the effectiveness of our CST heuristic and understand how changes in important remanufacturing-related parameters affect solutions, computational experiments were conducted. These experiments are summarized in this section.

By using IBM Concert Technology, the integrated model and the decomposition heuristic were coded in Java and solved using CPLEX 12.7. We used the default CPLEX parameters for the integrated model, and a two-hour maximum calculation time limit was imposed for each instance.

For the heuristic, we have solved the mathematical formulations in the first and the second phases (i.e., CLSP-R and VRPSPD) using CPLEX 12.7 with its default settings for up to 1200 s (20 min).

6.3.1. Performance Assessment of CST Heuristic on Random Instances

This section presents computational results obtained from performed tests on the random instances described in Section 5.1, achieved by the MILP model and our decomposition heuristic.

Note that all tests were carried out on the basis of a remanufacturing rate fixed at $\eta = 0.9$, a unit remanufacturing cost fixed at $p^r = 0.3 \times p^m$, with a number of iterations fixed at $Nbma\_Iter = 100$ and a number of restarts of value $Nbmax_{Rest} = 5$. These considerations are valid for all three classes of instances. In Tables 3–5, we report the results from both CPLEX and the *CST* heuristic. Each row corresponds to the calculation result carried out based on four instances.

For CPLEX, columns "#*Opt*." and "#*Feas*." indicate, respectively, the number of instances solved to optimality and those for which CPLEX has been capable to determine a feasible solution within the given time frame. Column "$Av.\ Opt.\ CPU(s)^{CPLEX}$" shows the average of the total computing time in seconds spent by CPLEX until solving the optimal instances, and "$Av.\ Gap\%\ (LB)^{CPLEX}$" represents the average optimality deviation calculated on the basis of feasible solutions. This optimality deviation was calculated as follows: $Gap\%\ (LB)^{CPLEX} = 100 \times \left(\frac{UB^{CPLEX} - LB^{CPLEX}}{LB^{CPLEX}} \right)$, where "$UB^{CPLEX}$" and "$LB^{CPLEX}$" represent the upper and lower bounds found by CPLEX after 2 h of calculation.

For the *CST* heuristic, column "$Av.\ CPU(s)^{CST}$" shows the average of the total computing time in seconds spent by the *CST* heuristic calculated based on all instances. The column $Av.\ Gap\%\ (Opt.)^{CST}$ indicates the average optimality deviation calculated based on instances solved to optimality. The optimality deviation, in this case, was calculated as follows: $Gap\%\ (Opt.)^{CST} = 100 \times \left(\frac{Sol^{CST} - OPT^{CPLEX}}{OPT^{CPLEX}} \right)$, with Sol^{CST} indicates the solution value found by the heuristic *CST* based on instances solved to optimality. The column $Av.\ Gap\%\ (LB)^{CST}$ indicates the average optimality deviation calculated based on instances for which CPLEX could not find an optimal solution and we take its lower bound. In this case, the optimality deviation is calculated as follows: $Gap\%\ (LB)^{CST} = 100 \times \left(\frac{Sol^{CST} - LB^{CPLEX}}{LB^{CPLEX}} \right)$. Finally, the column $Av.\ Gap\%\ (UB)^{CST}$ indicates the average optimality deviation calculated based on instances for which CPLEX was able to find an optimal feasible solution or not and we take its upper bound (which

is equal to the optimal solution in some cases). In this case, the optimality deviation is calculated as follows: $Gap\% \, (UB)^{CST} = 100 \times \left(\frac{Sol^{CST} - UB^{CPLEX}}{UB^{CPLEX}} \right)$.

In Table 4, we assess the performance of the algorithm against the commercial solver CPLEX on the instances of the second class of data set. Obtained results indicate that CPLEX was able to optimally solve all instances of up to five customers regardless of the number of period times considered. However, CPLEX was unable to optimally solve instances with more than 15 customers regardless of the number of period times during 2 h of execution time. These results are observed for both pickup scenarios (high and low). This implies that the integrated IPIRR-R model is sensitive to the instance size (number of both customers and time periods) and becomes unable to solve problems of medium and large sizes. This can be explained by the fact that the model contains a large number of variables and that it jointly optimizes the CLSP-R and the VRPSPD each of them is known as NP-hard. These results also confirm that integrated operations planning problems arising in the context of CLSCs with remanufacturing option are even more complex to solve, which means that the approximate methods are suitable for solving medium and large-sized instances of this class of problems.

Further, the proposed algorithm can obtain near-optimal solutions for the majority of instances with an average deviation from optimal solutions of 1% (with a minimum of 1% and a maximum of 2%) when pickup requests are relatively low, requiring very short average computation times of approximately 35 s (with a minimum of 10 s and a maximum of 105 s). However, when pickup requests are relatively high, the average deviation from optimal solutions is 1% (with a minimum of 1% and a maximum of 2%), requiring very short average computation times of approximately 44 s (with a minimum of 20 s and a maximum of 117 s).

For larger size problems for which CPLEX cannot find the optimal solution, the average deviation from the optimal solutions is 3% (with a minimum of 1% and a maximum of 9% when pickup requests are relatively low, requiring very short average computation times of approximately 35 s (with a minimum of 10 s and a maximum of 105 s). However, when the pickup requests are relatively high, the average deviation from optimal solutions is 4% (with a minimum of 1% and a maximum of 11, requiring very short average computation times of approximately 44 s (with a minimum of 20 s and a maximum of 117 s).

We can also highlight cases where CPLEX failed to find feasible solutions while the *CST* heuristic did, this applies to the case of problems with ($|N_C| = 35; |T| = 3$) and ($|N_C| = 50; |T| = 3$) with a pickup scenario, respectively, high and low. Moreover, negative values of the average $Gap(\%)$ correspond to cases where the heuristic exceeds the upper bound of CPLEX by finding better solutions. This can be observed in problems with $|N_C| > 35$ regardless of period times and pickup scenarios.

The obtained results from numerical tests on the standard class (Class 1) and the one with greater impact on transportation costs (Class 3) are depicted in Tables 5 and 6, respectively. We notice that the obtained results are consistent with those obtained from the performance assessment of CPLEX versus the CST heuristic on the second class of instances (Table 4).

Table 4. Computational results on the second class of instances: greater impact of manufacturing and remanufacturing systems.

		Low Pickups							
		CPLEX Results				CST Results			
$|N_C|$	$|T|$	#Opt	#Feas.	Av. Opt.CPU(s)^{CPLEX}	Av. Gap% (LB)^{CPLEX}	Av. CPU(s)^{CST}	Av. Gap% (Opt.)^{CST}	Av. Gap% (LB)^{CST}	Av. Gap% (UB)^{CST}
5	3	4	0	1	-	22	1	-	1
	6	4	0	132	-	10	1	-	1
10	3	2	2	8	0	27	1	1	1
	6	2	2	210	0	22	2	2	1
15	3	0	4	-	0	49	-	1	1
	6	0	4	-	0	24	-	1	1
20	3	0	4	-	0	28	-	1	1
	6	0	4	-	0	28	-	1	1
35	3	0	1	-	0	57	-	1	0
	6	0	4	-	0	23	-	9	6
50	3	0	0	-	-	105	-	-	-
	6	0	2	-	0	29	-	8	−8
		Min	1	0		10	1	1	−8
		Max	210	0		105	2	9	6
		Average	81	**0**		35	1	3	**1**

		High Pickups							
		CPLEX Results				CST Results			
$|N_C|$	$|T|$	#Opt	#Feas.	Av. Opt.CPU(s)^{CPLEX}	Av. Gap% (LB)^{CPLEX}	Av. CPU(s)^{CST}	Av. Gap% (Opt.)^{CST}	Av. Gap% (LB)^{CST}	Av. Gap% (UB)^{CST}
5	3	4	0	0	-	40	1	-	1
	6	4	0	2	-	21	1	-	1
10	3	3	1	26	0	27	1	1	1
	6	3	1	259	0	20	2	2	2
15	3	1	3	319	0	60	1	1	1
	6	0	4	-	0	29	-	2	2
20	3	0	4	-	0	36	-	1	1
	6	0	4	-	0	46	-	1	1
35	3	0	0	-	-	65	-	-	-
	6	0	4	-	0	24	-	11	9
50	3	0	1	-	0	117	-	1	−2
	6	0	3	-	0	38	-	10	−12
		Min	0	0		20	1	1	−12
		Max	319	0		117	2	11	9
		Average	79	**0**		44	1	4	**1**

Table 5. Computational results on the standard class of instances.

		Low Pickups							
$|N_C|$	$|T|$	**CPLEX Results**				**CST Results**			
		#Opt	**#Feas.**	*Av.* $Opt.CPU(s)^{CPLEX}$	*Av.* $Gap\% \, (LB)^{CPLEX}$	*Av.* $CPU(s)^{CST}$	*Av.* $Gap\% \, (Opt.)^{CST}$	*Av.* $Gap\% \, (LB)^{CST}$	*Av.* $Gap\% \, (UB)^{CST}$
5	3	4	0	0	-	30	9	-	9
	6	4	0	2	-	11	6	-	6
10	3	2	2	4	0	30	6	5	4
	6	4	0	48	-	28	11	-	11
15	3	1	3	312	0	77	3	6	5
	6	0	4	-	0	18	-	9	9
20	3	1	3	204	0	37	8	9	6
	6	0	4	-	0	38	-	7	5
35	3	0	0	-	-	61	-	-	-
	6	0	4	-	0	170	-	25	−3
50	3	0	1	-	0	188	-	5	−14
	6	0	2	-	0	323	-	5	−20
		Min		0	0	11	3	5	−20
		Max		312	0	323	11	25	11
		Average		45	0	84	8	10	6

		High Pickups							
$|N_C|$	$|T|$	**CPLEX Results**				**CST Results**			
		#Opt	**#Feas.**	*Av.* $Opt.CPU(s)^{CPLEX}$	*Av.* $Gap\% \, (LB)^{CPLEX}$	*Av.* $CPU(s)^{CST}$	*Av.* $Gap\% \, (Opt.)^{CST}$	*Av.* $Gap\% \, (LB)^{CST}$	*Av.* $Gap\% \, (UB)^{CST}$
5	3	4	0	0	-	30	10	-	10
	6	4	0	1	-	19	8	-	8
10	3	2	2	2	0	26	6	5	4
	6	4	0	44	-	21	14	-	14
15	3	1	3	229	0	52	4	7	6
	6	0	4	-	0	32	-	11	11
20	3	0	4	-	0	43	-	9	7
	6	0	4	-	0	34	-	8	6
35	3	0	0	-	-	59	-	-	-
	6	0	4	-	0	254	-	7	−15
50	3	0	2	-	0	158	-	6	−20
	6	0	2	-	0	296	-	7	−32
		Min		0	0	19	4	5	−32
		Max		229	0	296	14	11	14
		Average		28	0	85	9	8	8

Table 6. Computational results on the third class of instances: greater impact of transportation costs.

		Low Pickups							
		CPLEX Results				CST Results			
$|N_C|$	$|T|$	#Opt	#Feas.	Av. Opt.CPU(s)^{CPLEX}	Av. Gap% (LB)^{CPLEX}	Av. CPU(s)^{CST}	Av. Gap% (Opt.)^{CST}	Av. Gap% (LB)^{CST}	Av. Gap% (UB)^{CST}
5	3	4	0	0	-	10	37	-	37
	6	4	0	3	-	10	28	-	28
10	3	2	2	4	0	10	21	24	20
	6	4	0	24	-	9	44	-	44
15	3	1	3	276	0	55	16	26	24
	6	0	4	-	0	3	-	186	181
20	3	1	3	1814	0	18	24	28	20
	6	0	4	-	0	6	-	203	190
35	3	0	0	-	-	43	-	-	-
	6	0	4	-	0	17	-	262	136
50	3	0	1	-	0	13	-	256	113
	6	0	2	-	1	32	-	265	60
		Min		0	0	3	16	24	20
		Max		1814	1	55	44	265	190
		Average		138	0	19	32	157	21
		High Pickups							
		CPLEX Results				CST Results			
$|N_C|$	$|T|$	#Opt	#Feas.	Av. Opt.CPU(s)^{CPLEX}	Av. Gap% (LB)^{CPLEX}	Av. CPU(s)^{CST}	Av. Gap% (Opt.)^{CST}	Av. Gap% (LB)^{CST}	Av. Gap% (UB)^{CST}
5	3	4	0	0	-	9	39	-	39
	6	4	0	1	-	7	31	-	31
10	3	2	2	3	0	9	21	20	17
	6	4	0	17	-	8	49	-	49
15	3	1	3	1194	0	68	17	29	26
	6	0	4	-	0	9	-	200	195
20	3	1	3	2141	0	17	35	35	28
	6	0	4	-	0	3	-	217	201
35	3	0	0	-	-	39	-	-	-
	6	0	4	-	0	16	-	277	133
50	3	0	0	-	-	15	-	-	-
	6	0	2	-	1	32	-	289	59
		Min		0	0	3	17	20	17
		Max		2141	1	68	49	289	201
		Average		213	0	19	36	163	27

In Table 7, we report the performance of the proposed algorithm on the different classes of instances. This time, the column *Av. CPU(s)^{CPLEX}* shows the average of the total computing time in seconds from CPLEX calculated based on all instances.

We can notice that class 2 is the easiest to solve, followed by classes 1 and 3. This implies that the problem becomes easy to solve when the manufacturing and remanufacturing costs are significant. Moreover, from Tables 4–6, it should be noted that the average gaps for the low pickup scenario are always lower than those related to the high pickup scenario, which means that the instances are easy to solve when pickup requests are relatively low.

Table 7. Computational results on random instances with $|N_C| = \{5, 10, 15, 20, 35, 50\}$.

		Low Pickups			
Class	$|N_C|$	**CPLEX Solutions**	**CST Solutions**		
		$Av.CPU(s)^{CPLEX}$	$Av.CPU(s)^{CST}$	$Av.Gap\%\,(LB)^{CST}$	$Av.Gap\%\,(UB)^{CST}$
1	5	2 **(4)**	11	6	6
2		132 **(4)**	10	1	1
3		3 **(4)**	10	28	28
1	10	48 **(4)**	28	11	11
2		3710 **(2)**	22	2	1
3		24 **(4)**	9	44	44
1	15	- **(0)**	18	9	9
2		- **(0)**	24	1	1
3		- **(0)**	3	186	181
1	20	- **(0)**	38	7	5
2		- **(0)**	28	1	1
3		- **(0)**	6	203	190
1	35	- **(0)**	170	25	−3
2		- **(0)**	23	9	6
3		- **(0)**	17	262	136
1	50	- **(0)**	323	5	−21
2		- **(0)**	29	8	−8
3		- **(0)**	32	265	60
		High Pickups			
Class	$|N_C|$	**CPLEX Solutions**	**CST Solutions**		
		$Av.CPU(s)^{CPLEX}$	$Av.CPU(s)^{CST}$	$Av.Gap\%\,(LB)^{CST}$	$Av.Gap\%\,(UB)^{CST}$
1	5	1 **(4)**	19	8	8
2		2 **(4)**	21	1	1
3		1 **(4)**	7	31	31
1	10	44 **(4)**	21	14	14
2		1998 **(3)**	20	2	2
3		17 **(4)**	8	49	49
1	15	- **(0)**	32	11	11
2		- **(0)**	29	2	2
3		- **(0)**	9	200	195
1	20	- **(0)**	34	8	6
2		- **(0)**	46	1	1
3		- **(0)**	3	217	201
1	35	- **(0)**	254	7	−15
2		- **(0)**	24	11	9
3		- **(0)**	16	277	133
1	50	- **(0)**	296	7	−32
2		- **(0)**	38	10	−12
3		- **(0)**	32	289	59

Note: $|T| = 6$ **(i)** denotes the number of instances solved to optimality out of 4 instances, (-) indicates that the total computing time exceeds the time limit of 2 h.

To sum up, the numerical results showed that the IPIRP-R model becomes easy to solve when manufacturing and remanufacturing costs become important and when

pickup requests are relatively low. The same results revealed that CPLEX can effectively solve optimality instances involving up to five customers regardless of the number of time periods and pickup scenarios. For the same type of instances, the heuristic CST has provided solutions with a quality close to those found by the solver CPLEX. For instances with 35 up 50 customers and three periods, respectively, with a relatively high and low pickup scenario, our decomposition heuristic clearly outperformed the CPLEX solver, finding better solutions.

In all cases, the heuristic *CST* could usually find good feasible solutions within acceptable computing time. Furthermore, regardless of the number of time periods and pickup scenarios for which the solver failed to find a solution within two hours of computing, our solution approach successfully identified feasible solutions for instances with up to 50 consumers. These results are consistent for the three classes of instances.

6.3.2. Sensitivity Analysis

In order to assess and analyze the contribution of the remanufacturing process in terms of reducing operational costs, we present through this section a sensitivity analysis of the performance of the CLSC considered with respect to variations of the parameters related to the remanufacturing operation, in particular, the remanufacturing rate (η) and its unit cost ($p^r = \sigma \times p^m$). To do this, 200 new instances were generated on the basis of the instances from the standard class, each with $|N_C| = 5$ customers and $|T| = 6$ periods, by varying the parameters $\eta \in \{0.1; 0.3; 0.5; 0.7; 0.9\}$, $\sigma \in \{0.1; 0.3; 0.5; 0.7; 0.9\}$, two scenarios for pickups, and for each combination four instances with different requests and coordinates are considered.

In Table 8, we give a comparison example of costs for different values of the percentage of remanufacturing unit cost for a given remanufacturing rate. Note that each row is calculated based on four instances. In addition, to better understand other key information, we illustrate in Figure 5 the selected costs and parameters (for the high pickup scenario).

Table 8. Comparison of operational costs (on average) under different values of remanufacturing unit cost according to the pickup scenarios.

Pickup scenario		σ	TC	MC	RC	HNP	HRP	MS	RS	TCT
	High pickups	0.1	24,405	3822	946	278	1655	3114	3114	3800
		0.3	24,265	3318	2806	479	1478	3126	3126	3400
		0.5	26,479	3336	4440	346	1588	3120	3120	3400
		0.7	28,663	3480	6350	353	1420	3114	3114	3500
		0.9	31,734	3630	8473	625	1612	3114	3114	3600
	Low pickups	0.1	32,614	6978	630	322	1016	6096	6096	3800
		0.3	31,816	6444	1868	304	1065	6102	6102	3400
		0.5	33,252	6216	3000	371	919	6108	6108	3400
		0.7	34,990	6492	4242	244	988	6096	6096	3500
		0.9	37,307	6636	5767	467	1078	6096	6096	3600

Note: $|T| = 6$; $\eta = 0.9$; **TC:** total cost; **MC:** manufacturing total cost; **RC:** remanufacturing total cost; **HNP:** holding cost from storing new products; **HRP:** holding cost from storing return products; **MS:** manufacturing setup cost; **RS:** remanufacturing setup cost; **TCT:** transport total cost.

In Figure 5, we can observe, for a given remanufacturing unit cost, as the remanufacturing rate increases, the total remanufacturing cost increases, but the total manufacturing cost decreases. This suggests that remanufacturing activities are gradually replacing manufacturing ones. This observation is more significant when the remanufacturing rate is significant (for example, $\eta = 0.9$) because there is enough stock in terms of returned EOL to cover the delivery demands.
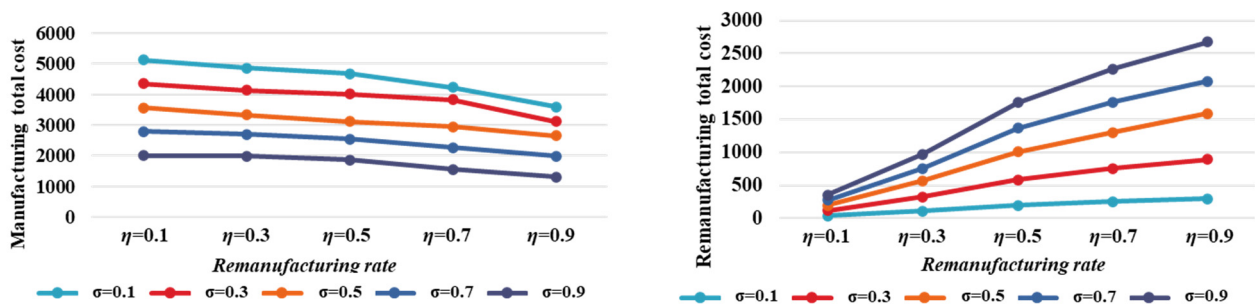
Figure 5. Sensitivity analysis on the rate and unit cost of remanufacturing.

7. Conclusions and Perspectives

In this paper, we proposed a two-phase decomposition heuristic to tackle the Integrated Production-Inventory-Routing Problem with Remanufacturing (IPIRP-R) that is a generalization of the classical single-item capacitated lot-sizing with the option of remanufacturing and the VRP with simultaneous pickup and delivery. Optimizing this problem within a CLSC network can have a positive impact on its economic and environmental performance. Indeed, recovering the residual value of returned EOL products through the remanufacturing process, leads to minimizing the total integrated cost of the involved operations, especially since we assume that the fixed and variable costs of the remanufacturing activities are less expensive than manufacturing and knowing that customer demands can be met with either new or remanufactured products or both. This means that remanufacturing activities can reduce production costs as long as manufacturing activities can be replaced by remanufacturing ones if there is a sufficient amount of EOL products collected from customers and stored at the returns inventory level. On the other hand, the collection and remanufacturing of EOL products can be considered a means of reducing greenhouse gas emissions and therefore contribute to improving the environmental performance of the network.

The proposed heuristic procedure splits the problem into two subproblems, which are subsequently solved iteratively. Based on approximative visiting costs, a restricted capacitated CLSP-R is solved in the first phase. Without expressly considering routing decisions, this phase decides when and how much to manufacture and to remanufacture at each period, when to visit customers and how much to deliver and pick up concurrently at each visit, as well as the assignment of customers to vehicles. Based on the assignment decisions of customers to vehicles from the first phase, the second phase decides the optimal sequence of visiting a set of customers served by a set of vehicles in each period by solving a restricted VRPSPD.

First, the effectiveness of the proposed solution approach has been assessed on a set of computational experiments performed on three classes of randomly generated problem instances against cutting-edge optimization software. Numerical results on benchmark instances showed that our method can find good quasi-optimal solutions in an acceptable computing time for small and medium-scale problems, characterized by a greater impact of manufacturing and remanufacturing costs, particularly when pickup requests are relatively low. The effects of remanufacturing parameters on the balance between manufacturing and remanufacturing are then captured through extensive computer analyses, from which pertinent management information has been derived. The obtained results demonstrate that the developed algorithm is operational and can thus be applied to many problems of common interest arising in the context of CLSs, particulary in the production and distribution of electrical and electronic components, returnable and reusable packaging products, food industry, (e.g., perishable products) and the beverage industry.

Despite the relevance and applicability of the approaches developed, the problems of integrated planning of operations in the context of sustainable supply chains are difficult to solve and require sophisticated approach methods to deal with their complexity. Given the obtained results and the contributions presented in this study, it would therefore

be relevant to continue our research work in order to improve the performance of the methods used. A notable perspective for future research would be to develop for each of the subproblems CLSP-R and VRPSPD, a fast heuristic procedure. In fact, each of these two sub-problems becomes difficult when the problem size increases and, therefore, developing effective techniques to solve it is necessary to reduce computation time and solve large-scale problems. Furthermore, better effective methods of approximating the visiting costs together with a more sophisticated diversification strategy (e.g., the update mechanism from [5]) can be used to drive the heuristic to more thoroughly explore the solution space, narrowing the gap to optimal solutions. Moreover, to assess the quality of heuristic solutions, another interesting perspective would be to compute good lower bounds by applying Lagrangian relaxation to the IPIRP-R model.

Future research work should also focus on expanding the IPIRP-R formulation by including carbon emissions exhausted from production (manufacturing and remanufacturing), remanufacturing, inventory, and routing activities. Moreover, investigating the IPIRP-R formulation with a heterogeneous fleet of capacitated vehicles and time windows to reduce transportation costs while guaranteeing a high service level. Moreover, it would be interesting to introduce multiple plants with multiple products as well as demands uncertainty (e.g.: quality of EOL products returned, amounts of delivery and pickup requests, etc.). Furthermore, it would be interesting to incorporate the cost for permanent assets capital to assess the effect of the tangible depreciation of permanent assets on both the total integrated cost and the involved operations' decisions. Finally, it would also be wise to develop a decision support system that integrates our solution approaches developed in this study with a returns management information system to help industrial decision-makers cope with the complexity imposed by the reverse logistics processes throughout a closed-loop network. By considering all these dimensions, we come closer to the real-life situations that occur in remanufacturing processes.

References

1. Adulyasak, Y.; Cordeau, J.F.; Jans, R. Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS J. Comput.* **2014**, *26*, 103–120. [CrossRef]
2. Avci, M.; Yildiz, S.T. A matheuristic solution approach for the production routing problem with visit spacing policy. *Eur. J. Oper. Res.* **2019**, *279*, 572–588. [CrossRef]
3. Chekoubi, Z.; Trabelsi, W.; Sauer, N. The integrated production-inventory-routing problem in the context of reverse logistics: The case of collecting and remanufacturing of EOL products. In Proceedings of the 2018 4th International Conference on Optimization and Applications (ICOA), Mohammedia, Morocco, 26–27 April 2018. [CrossRef]
4. Chekoubi, Z.; Sauer, N.; Trabelsi, W. The Integrated Production-Inventory-Routing Problem of EOL Products with Simultaneous Delivery and Pickup. In Proceedings of the 7th IEEE International Conference on Advanced Logistics and Transport (ICALT), Marrakech, Morocco, 14–16 June 2019.
5. Absi, N.; Archetti, C.; Dauzère-Pérès, S.; Feillet, D. A two-phase iterative heuristic approach for the production routing problem. *Transp. Sci.* **2015**, *49*, 784–795. [CrossRef]
6. Chandra, P. A dynamic distribution model with warehouse and customer replenishment requirements. *J. Oper. Res. Soc.* **1993**, *44*, 681–692. [CrossRef]
7. Li, Y.; Chu, F.; Chu, C.; Zhu, Z. An efficient three-level heuristic for the large-scaled multi-product production routing problem with outsourcing. *Eur. J. Oper. Res.* **2019**, *272*, 914–927. [CrossRef]

8. Chandra, P.; Fisher, M.L. Coordination of production and distribution planning. *Eur. J. Oper. Res.* **1994**, *72*, 503–517. [CrossRef]
9. Miranda, P.L.; Cordeau, J.-F.; Ferreira, D.; Jans, R.; Morabito, R. A decomposition heuristic for a rich production routing problem. *Comput. Oper. Res.* **2018**, *98*, 211–230. [CrossRef]
10. Brown, G.; Keegan, J.; Vigus, B.; Wood, K. The Kellogg Company optimizes production, inventory, and distribution. *Interfaces* **2001**, *31*, 1–15. [CrossRef]
11. Thomas, D.J.; Griffin, P.M. Coordinated supply chain management. *Eur. J. Oper. Res.* **1996**, *94*, 1–15. [CrossRef]
12. Bard, J.F.; Nananukul, N. The integrated production-inventory-distribution-routing problem. *J. Sched.* **2009**, *12*, 257–280. [CrossRef]
13. Qiu, Y.; Ni, M.; Wang, L.; Li, Q.; Fang, X.; Pardalos, P.M. Production routing problems with reverse logistics and remanufacturing. *Transp. Res. Part E Logist. Transp. Rev.* **2018**, *111*, 87–100. [CrossRef]
14. Lei, L.; Liu, S.; Ruszczynski, A.; Park, S. On the integrated production, inventory, and distribution routing problem. *IIE Trans. Inst. Ind. Eng.* **2006**, *38*, 955–970. [CrossRef]
15. Miranda, P.L.; Morabito, R.; Ferreira, D. Mixed integer formulations for a coupled lot-scheduling and vehicle routing problem in furniture settings. *INFOR* **2019**, *57*, 563–596. [CrossRef]
16. Solyalı, O.; Süral, H. A multi-phase heuristic for the production routing problem. *Comput. Oper. Res.* **2017**, *87*, 114–124. [CrossRef]
17. Brahimi, N.; Aouam, T. Multi-item production routing problem with backordering: A MILP approach. *Int. J. Prod. Res.* **2016**, *54*, 1–18. [CrossRef]
18. Vahdani, B.; Niaki, S.T.A.; Aslanzade, S. Production-inventory-routing coordination with capacity and time window constraints for perishable products: Heuristic and meta-heuristic algorithms. *J. Clean. Prod.* **2017**, *161*, 598–618. [CrossRef]
19. Alkaabneh, F.; Diabat, A.; Gao, H.O. Benders decomposition for the inventory vehicle routing problem with perishable products and environmental costs. *Comput. Oper. Res.* **2020**, *113*, 104751. [CrossRef]
20. Alkaabneh, F.M.; Lee, J.; Gómez, M.I.; Gao, H.O. A systems approach to carbon policy for fruit supply chains: Carbon tax, technology innovation, or land sparing? *Sci. Total Environ.* **2021**, *767*, 144211. [CrossRef] [PubMed]
21. Zhang, Q.; Sundaramoorthy, A.; Grossmann, I.E.; Pinto, J.M. Multiscale production routing in multicommodity supply chains with complex production facilities. *Comput. Oper. Res.* **2017**, *79*, 207–222. [CrossRef]
22. Archetti, C.; Bertazzi, L.; Paletta, G.; Speranza, M.G. Analysis of the maximum level policy in a production-distribution system. *Comput. Oper. Res.* **2011**, *38*, 1731–1746. [CrossRef]
23. Darvish, M.; Archetti, C.; Coelho, L.C. Trade-offs between environmental and economic performance in production and inventory-routing problems. *Int. J. Prod. Econ.* **2019**, *217*, 269–280. [CrossRef]
24. Qiu, Y.; Qiao, J.; Pardalos, P.M. A branch-and-price algorithm for production routing problems with carbon cap-and-trade. *Omega* **2017**, *68*, 49–61. [CrossRef]
25. Bard, J.F.; Nananukul, N. Heuristics for a multiperiod inventory routing problem with production decisions. *Comput. Ind. Eng.* **2009**, *57*, 713–723. [CrossRef]
26. Bard, J.F.; Nananukul, N. A branch-and-price algorithm for an integrated production and inventory routing problem. *Comput. Oper. Res.* **2010**, *37*, 2202–2217. [CrossRef]
27. Fumero, F.; Vercellis, C. Synchronized development of production, inventory, and distribution schedules. *Transp. Sci.* **1999**, *33*, 330–340. [CrossRef]
28. Adulyasak, Y.; Cordeau, J.-F.; Jans, R. Benders decomposition for production routing under demand uncertainty. *Oper. Res.* **2015**, *63*, 851–867. [CrossRef]
29. Moin, N.H.; Yuliana, T. Three-Phase Methodology Incorporating Scatter Search for Integrated Production, Inventory, and Distribution Routing Problem. *Math. Probl. Eng.* **2015**, *2015*, 1–11. [CrossRef]
30. Adulyasak, Y.; Cordeau, J.-F.; Jans, R. The production routing problem: A review of formulations and solution algorithms. *Comput. Oper. Res.* **2015**, *55*, 141–152. [CrossRef]
31. Ydulyasak, Y.; Cordeau, J.-F.; Jans, R. Optimization-based adaptive large neighborhood search for the production routing problem. *Transp. Sci.* **2014**, *48*, 20–45. [CrossRef]
32. Kumar, R.S.; Kondapaneni, K.; Dixit, V.; Goswami, A.; Thakur, L.; Tiwari, M. Multi-objective modeling of production and pollution routing problem with time window: A self-learning particle swarm optimization approach. *Comput. Ind. Eng.* **2016**, *99*, 29–40. [CrossRef]
33. Armentano, V.; Shiguemoto, A.; Løkketangen, A. Tabu search with path relinking for an integrated productiondistribution problem. *Comput. Oper. Res.* **2011**, *38*, 1199–1209. [CrossRef]
34. Calvete, H.I.; Galé, C.; Oliveros, M.-J. Bilevel model for productiondistribution planning solved by using ant colony optimization. *Comput. Oper. Res.* **2011**, *38*, 320–327. [CrossRef]
35. Boudia, M.; Prins, C. A memetic algorithm with dynamic population management for an integrated production-distribution problem. *Eur. J. Oper. Res.* **2009**, *195*, 703–715. [CrossRef]
36. Van Buer, M.G.; Woodruff, D.L.; Olson, R.T. Solving the medium newspaper production/distribution problem. *Eur. J. Oper. Res.* **1999**, *115*, 237–253. [CrossRef]
37. Díaz-Madroñero, M.; Peidro, D.; Mula, J. A review of tactical optimization models for integrated production and transport routing planning decisions. *Comput. Ind. Eng.* **2015**, *88*, 518–535. [CrossRef]

38. Iassinovskaia, G.; Limbourg, S.; Riane, F. The inventory-routing problem of returnable transport items with time windows and simultaneous pickup and delivery in closed-loop supply chains. *Int. J. Prod. Econ.* **2017**, *183*, 570–582. [CrossRef]

39. Miranda, P.L.; Morabito, R.; Ferreira, D. Optimization model for a production, inventory, distribution and routing problem in small furniture companies. *TOP* **2018**, *26*, 1–38. [CrossRef]

40. Russell, R.A. Mathematical programming heuristics for the production routing problem. *Int. J. Prod. Econ.* **2017**, *193*, 40–49. [CrossRef]

41. Boudia, M.; Louly, M.; Prins, C. A reactive GRASP and path relinking for a combined production-distribution problem. *Comput. Oper. Res.* **2007**, *34*, 3402–3419. [CrossRef]

42. Savaskan, R.C.; Bhattacharya, S.; Van Wassenhove, L.N. Closed-Loop Supply Chain Models with Product Remanufacturing. *Manage. Sci.* **2004**, *50*, 239–252. [CrossRef]

43. DeCroix, G.A. Optimal policy for a multiechelon inventory system with remanufacturing. *Oper. Res.* **2006**, *54*, 532–543. [CrossRef]

44. Geyer, R.; Van Wassenhove, L.N.; Atasu, A. The economics of remanufacturing under limited component durability and finite product life cycles. *Manage. Sci.* **2007**, *53*, 88–100. [CrossRef]

45. Atasu, A.; Sarvary, M.; Van Wassenhove, L.N. Remanufacturing as a marketing strategy. *Manage. Sci.* **2008**, *54*, 1731–1746. [CrossRef]

46. Manouchehri, F.; Nookabadi, A.S.; Kadivar, M. Production routing in perishable and quality degradable supply chains. *Heliyon* **2020**, *6*, e03376. [CrossRef]

47. Battarra, M.; Cordeau, J.-F.; Iori, M. Chapter 6: Pickup-and-Delivery Problems for Goods Transportation. In *Vehicle Routing*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2014; pp. 161–191. [CrossRef]

48. Subramanian, A.; Uchoa, E.; Pessoa, A.A.; Ochi, L.S. Branch-cut-and-price for the vehicle routing problem with simultaneous pickup and delivery. *Optim. Lett.* **2013**, *7*, 1569–1581. [CrossRef]

49. Qu, Y.; Bard, J.F. A branch-and-price-and-cut algorithm for heterogeneous pickup and delivery problems with configurable vehicle capacity. *Transp. Sci.* **2015**, *49*, 254–270. [CrossRef]

50. Çetinkaya, S.; Üster, H.; Easwaran, G.; Keskin, B.B. An integrated outbound logistics model for frito-lay: Coordinating aggregate-level production and distribution decisions. *Interfaces* **2009**, *39*, 460–475. [CrossRef]

51. Neves-Moreira, F.; Almada-Lobo, B.; Cordeau, J.-F.; Guimarães, L.; Jans, R. Solving a large multi-product production-routing problem with delivery time windows. *Omega* **2019**, *86*, 154–172. [CrossRef]

52. Qiu, Y.; Wang, L.; Xu, X.; Fang, X.; Pardalos, P.M. A variable neighborhood search heuristic algorithm for production routing problems. *Appl. Soft Comput. J.* **2018**, *66*, 311–318. [CrossRef]

53. Schenekemberg, C.M.; Scarpin, C.T.; Pécora, J.E.; Guimarães, T.A.; Coelho, L.C. The two-echelon production-routing problem. *Eur. J. Oper. Res.* **2020**, *288*, 436–449. [CrossRef]

54. Marchetti, P.A.; Gupta, V.; Grossmann, I.E.; Cook, L.; Valton, P.-M.; Singh, T.; Li, T.; André, J. Simultaneous production and distribution of industrial gas supply-chains. *Comput. Chem. Eng.* **2014**, *69*, 39–58. [CrossRef]

55. Zamarripa, M.; Marchetti, P.A.; Grossmann, I.E.; Singh, T.; Lotero, I.; Gopalakrishnan, A.; Besancon, B.; André, J. Rolling Horizon Approach for Production-Distribution Coordination of Industrial Gases Supply Chains. *Ind. Eng. Chem. Res.* **2016**, *55*, 2646–2660. [CrossRef]

56. Hurter, A.; Van Buer, M. The Newspaper Production/Distribution Problem. *J. Bus. Logist.* **1996**, *17*, 85–107.

57. Russell, R.; Chiang, W.-C.; Zepeda, D. Integrating multi-product production and distribution in newspaper logistics. *Comput. Oper. Res.* **2008**, *35*, 1576–1588. [CrossRef]

58. Chiang, W.-C.; Russell, R.; Xu, X.; Zepeda, D. A simulation/metaheuristic approach to newspaper production and distribution supply chain problems. *Int. J. Prod. Econ.* **2009**, *121*, 752–767. [CrossRef]

59. Aydinel, M.; Sowlati, T.; Cerda, X.; Cope, E.; Gerschman, M. Optimization of production allocation and transportation of customer orders for a leading forest products company. *Math. Comput. Model.* **2008**, *48*, 1158–1169. [CrossRef]

60. Troncoso, J.J.; Garrido, R.A. Forestry production and logistics planning: An analysis using mixed-integer programming. *For. Policy Econ.* **2005**, *7*, 625–633. [CrossRef]

61. Zeddam, B.; Belkaid, F.; Bennekrouf, M. Bi-objective optimization for the Production-Routing problem Cost vs. Environment with energy consideration. In Proceedings of the 2019 International Conference on Advanced Electrical Engineering (ICAEE), Algiers, Algeria, 19–21 November 2019. [CrossRef]

62. Nouiri, M.; Bekrar, A.; Trentesaux, D. An energy-efficient scheduling and rescheduling method for production and logistics systems. *Int. J. Prod. Res.* **2020**, *58*, 3263–3283. [CrossRef]

63. Fang, X.; Du, Y.; Qiu, Y. Reducing carbon emissions in a closed-loop production routing problem with simultaneous pickups and deliveries under carbon cap-and-trade. *Sustainability* **2017**, *9*, 2198. [CrossRef]

64. Zhang, Y.; Alshraideh, H.; Diabat, A. A stochastic reverse logistics production routing model with environmental considerations. *Ann. Oper. Res.* **2018**, *271*, 1023–1044. [CrossRef]

65. Shuang, Y.; Diabat, A.; Liao, Y. A stochastic reverse logistics production routing model with emissions control policy selection. *Int. J. Prod. Econ.* **2019**, *213*, 201–216. [CrossRef]

66. Ghasemkhani, A.; Tavakkoli-Moghaddam, R.; Shahnejat-Bushehri, S.; Momen, S. An integrated production inventory routing problem for multi perishable products with fuzzy demands and time windows. *IFAC-PapersOnLine* **2019**, *52*, 523–528. [CrossRef]

67. Boudia, M.; Louly, M.A.O.; Prins, C. Fast heuristics for a combined production planning and vehicle routing problem. *Prod. Plan. Control* **2008**, *19*, 85–96. [CrossRef]
68. Shiguemoto, A.L.; Armentano, V.A. A tabu search procedure for coordinating production, inventory and distribution routing problems. *Int. Trans. Oper. Res.* **2010**, *17*, 179–195. [CrossRef]
69. Agra, A.; Requejo, C.; Rodrigues, F. An adjustable sample average approximation algorithm for the stochastic production-inventory-routing problem. *Networks* **2018**, *72*, 5–24. [CrossRef]
70. Chitsaz, M.; Cordeau, J.-F.; Jans, R. A unified decomposition matheuristic for assembly, production, and inventory routing. *INFORMS J. Comput.* **2019**, *31*, 134–152. [CrossRef]