A gradient-descent-based framework for solving a stochastic two-echelon delivery problem with cargo-bikes

Fatima Ezzahra Achamrah, Jakob Puchinger

A gradient-descent-based framework for solving a stochastic two-echelon delivery problem with cargo-bikes

Fatima Ezzahra Achamrah[a,b,*], Jakob Puchinger[c,d]

[a] *Sheffield University Management School, The University of Sheffield, Conduit Road, Sheffield, S10 1FL, United Kingdom*
[b] *Mines Paris, PSL University, Centre for management science (CGS), i3 UMR9217 CNRS, 75006 Paris, France*
[c] *EM Normandie Business School, Métis Lab, 92110, Clichy, France*
[d] *Université Paris-Saclay, CentraleSupélec, Laboratoire Génie Industriel, 91190, Gif-sur-Yvette, France*

Abstract

In this paper, we examine a stochastic two-echelon vehicle routing problem (2e-VRP) using cargo bikes within hyperconnected networks. The focus is on the integration of both delivery and pickup of reusable containers, incorporating transshipment operations, time windows, and stochastic demand constraints. The model also considers the flow consolidation for empty and full containers at the satellites and allows for load splitting. Moreover, this study introduces an innovative gradient-descent-based optimization framework to handle the combinatorial complexity of the proposed model, opening new avenues in stochastic programming. Furthermore, the performance of this novel method is compared against the sample average approximation method, evaluating both solution quality and computational efficiency. Experimental results demonstrate the model's advanced integration and flexibility, significantly enhancing urban delivery systems and advancing logistics and transportation optimization research.

Keywords: Hyperconnected Logistics Networks, Two-Echelon Vehicle Routing, Cargo Bikes, Stochastic Demand, Stochastic Gradient Descent

1. Introduction

Urbanization is rapidly transforming city logistics, presenting complex challenges requiring innovative solutions. The rise of people residing in urban areas leads to a consistent increase in road freight transport, contributing to a significant portion of transport-induced greenhouse gas emissions (European-Commission, 2023). This trend is further intensified by the growing e-commerce market, which increases smaller, more frequent home deliveries. Recent regulatory efforts to reduce urban truck traffic and promote eco-friendly vehicles for last-mile delivery are only partially successful and insufficient to address the evolving complexities of urban logistics (Anderluh et al., 2019). In response, our research proposes a new model that combines two-echelon distribution, containerization, and consolidation of forward and reverse flows, all based on the concept of hyperconnected networks (Pan et al., 2019). This approach differs

*Corresponding author
Email addresses: f.e.achamrah@sheffield.ac.uk (Fatima Ezzahra Achamrah), jpuchinger@em-normandie.fr (Jakob Puchinger)

from traditional models by fostering horizontal collaboration among multiple suppliers and customers, crucial for achieving efficiency in urban logistics (Cleophas et al., 2019).

The presented research addresses the challenges of inner-city deliveries by integrating innovative last-mile delivery methods within a two-echelon distribution system. In this system, goods are initially transported to intermediary hubs, known as satellites, using traditional feeder vehicles like trucks (Bakach et al., 2021). At these satellites, the cargo is transferred to zero-emission vehicles, such as cargo bikes, which are well-suited for densely populated areas. The planning and execution of this system involve coordinating the interaction between the two distinct vehicle fleets, ensuring seamless transition and delivery efficiency.

Facilitated within the hyperconnected network's operational framework, the proposed model is based on smart, standardized, modular, shared, and reusable containers that allow for efficient handling and transfer of goods (Ballot et al., 2014). Containerization allows managing the rapid transit of small-sized orders, guarantees swift and secure transitions of goods from feeder vehicles to last-mile vehicles like cargo bikes, and supports the co-loading of products from diverse carriers within the same vehicle and satellite hubs. Business models exemplifying these strategies can be observed in GS1, Amazon, ES3, Flexe.com, and Darkstore.com (Kim et al., 2021).

Our model integrates the effective consolidation of both forward and reverse flows at the satellite nodes, which double as cross-docking terminals. This consolidation, including transshipment between satellites for inventory sharing, is not limited to the forward flow of containers to customers; it also encompasses the reverse flow of containers, ensuring their reuse, continuity of the distribution operations under stochastic demands, and supporting sustainable logistics practices.

The challenges of implementing such a model are manifold. Aside from the inherent complexity of two-echelon vehicle routing problems (2e-VRP), routing decisions become very complex, as the forward distribution to customers and the return of empty containers to the satellites for further reuse need to be optimized (Pan et al., 2021; Achamrah et al., 2023). This also involves consolidating empty and loaded container flows, managing transshipment operations between satellites, and accommodating split deliveries. These elements and other constraints are critical, such as time windows, service times, and stochastic demands. These factors are critical gaps in the 2e-VRP field, as highlighted in a recent survey by Sluijk et al. (2022). Therefore, the integration of all these constraints makes the problem and its modeling complex and requires the development of efficient and robust solution algorithms.

By addressing these key areas, our research introduces significant contributions to the field of logistics and transportation optimization.

- We address a previously unexplored research area by modeling a stochastic 2e-VRP within hyperconnected networks, encompassing both deliveries and pickups of reusable containers while incorporating transshipment operations, time windows, and service time constraints. This comprehensive model also includes the optimization of flow consolidation for both empty and loaded containers, as well as load splitting, ensuring efficient resource utilization and preventing product or packaging unavailability.

- The paper proposes an innovative gradient-descent-based optimization framework, which represents a pioneering approach in stochastic programming. By formulating the problem as a differentiable program and employing a continuous algorithm, this approach opens new avenues for solving complex logistics optimization challenges. We assess the

performance of this novel solution method against the sample average approximation method in terms of solution quality and computational efficiency.

- We conduct experiments to provide valuable insights into how the advanced integration and flexibility of the proposed model can enhance urban delivery systems, ultimately contributing to the advancement of logistics and transportation optimization research.

The remainder of this paper is structured as follows. In Section 2, related works are presented. Sections 3, 4, and 5 describe the problem, formulations, and solution method, respectively. Section 6 provides reports on the experimental results. Finally, Section 7 highlights the main conclusions and perspectives.

2. Related literature

This section identifies gaps in existing literature regarding two-echelon vehicle routing problems (2e-VRP), with a specific emphasis on the integration of pickup and delivery flows, as well as cross-docking in hyperconnected networks. It also underscores the paper's objective in addressing these gaps, particularly through advanced mathematical modeling and tackling combinatorial complexity.

2.1. 2e-VRP

The literature on the 2e-VRP has significantly evolved, driven by the need to address complex and congested urban logistics. Initially, the focus was optimizing routes from central depots to final destinations via intermediate satellite nodes. Gonzalez-Feliu et al. (2008) and Perboli et al. (2011) provided foundational categorizations of 2e-VRP, ranging from basic models without time considerations to more complex forms incorporating time windows and synchronization at satellites.

Recent research has expanded the scope and complexity of 2e-VRP models. Zhang et al. (2023) introduced a stochastic formulation of 2e-VRP integrated with loading bay reservations, addressing the uncertainty in urban logistics and the coordination of multiple courier companies. (Yu et al., 2021) proposed a 2e-VRP variant that incorporates covering locations and occasional drivers, exploring new flexibility in last-mile delivery systems.

Recent trends in 2e-VRP research have further emphasized incorporating sustainable practices, particularly by integrating zero-emission vehicles. Significant contributions in this area have been made to explore the use of electric and autonomous vehicles, and cargo bikes for eco-friendly urban deliveries (Anderluh et al., 2021; Bakach et al., 2021; Jie et al., 2019; Breunig et al., 2019; Yu et al., 2020). This shift is motivated not only by environmental concerns but also by the aim to optimize operational efficiencies, including the optimization of locations and vehicle synchronization in urban peripheries, as elaborated by Anderluh et al. (2021) and Enthoven et al. (2020).

2.2. 2e-VRP with pickup and delivery

In parallel, the aspect of reverse logistics in Vehicle Routing Problem (VRP), particularly in VRP with mixed and simultaneous Pickup and Delivery (VRPPD), has gained attention (Guide and Van Wassenhove, 2009; Berbeglia et al., 2007; Battarra et al., 2014). The objective here is to manage both product deliveries and returns within the same vehicle system (Guide and Van Wassenhove, 2009), addressing challenges like simultaneous pickup and delivery of

reusable containers, collaborative pickup and delivery of reusable containers, time window constraints, product incompatibilities, and scheduling issues among linehaul and backhaul customers (Iassinovskaia et al., 2017; Achamrah et al., 2022b, 2020; Kassem and Chen, 2012; Hu et al., 2015; Tarantilis et al., 2013). Nevertheless, notable gaps exist, especially regarding the 2e-VRP framework and the handling of reusable containers under uncertainty and time windows, integrating flow consolidation, and transshipment operations.

2.3. 2e-VRP with cross-docking

Cross-docking, a critical aspect in the evolution of VRP and particularly in VRP with Cross-Docking (VRPCD), has been explored to optimize shipment consolidations from various sources. Diverse facets of VRPCD have been examined, including single product and lone cross-dock setups (Lee et al., 2006), multi-product and single cross-dock VRPCD (Benjamin, 2020), and multi cross-docks VRPCD with split deliveries (Wang et al., 2017). Studies have also integrated time window constraints and promoted transshipment between cross-docks (Ma et al., 2011; Lim et al., 2005; Chen et al., 2006; Marjani et al., 2012). Further advancing the integration of cross-docking in 2e-VRP, Qiu et al. (2021) combined production routing with 2e-VRP and cross-docking satellites. However, only a few papers address VRPCD with both pickup and delivery, integrating reverse flows of products into a single cross-dock VRPCD (Zuluaga et al., 2016; Kheirkhah and Rezaei, 2015; Kaboudani et al., 2020). Cross-docking integration within the 2e-VRP framework, particularly in efficiently managing both forward and reverse flows and transshipment operations under uncertainty, remains underexplored with limited comprehensive models addressing this aspect.

2.4. 2e-VRP in hyperconnected networks

Building on the foundation of 2e-VRP, this section delves into its evolution within the framework of hyperconnected networks. From a forward-looking perspective, hyperconnected networks, such as interconnected, intertwined networks, and the concept of the "Physical Internet" represent a transformative approach to addressing the complexities of modern logistics. Endorsed by the Alliance for Logistics Innovation through Collaboration in Europe, these paradigms advocate for a cost-effective, resilient transition towards Net-Zero logistics through enhanced connectivity and sustainability (Montreuil, 2011; Ballot et al., 2014). Central to our research is the role of hyperconnected networks in facilitating horizontal collaboration through the use of smart, standardized, modular, shared, and reusable containers for seamless interaction and coordination among diverse logistics operators (Pan et al., 2019; Kim et al., 2021). Within this framework, satellite nodes function as hubs, networks of logistics terminals, serving as crucial consolidation and transshipment points for a highly efficient, scalable, and flexible logistics infrastructure (Montreuil, 2011; Ballot et al., 2014; Pan et al., 2019; Kim et al., 2021).

In conclusion, while numerous studies have addressed 2e-VRP, focusing on the use of zero-emission vehicles for last-mile delivery and integrating flow consolidation through cross-docking, managing the return flow and reusable containers has received limited attention (Cuda et al., 2015; Cattaruzza et al., 2017). Additionally, the incorporation of 2e-VRP with forward and reverse flows in hyperconnected systems is yet to be fully considered. Most existing literature on hyperconnected networks primarily deals with functional design for a road-based transit center (Meller et al., 2012), tracking technology for the management of reusable containers (Roch et al., 2014), VRP with a single echelon (Fazili et al., 2017; Gansterer and Hartl, 2018), stochastic VRP with transshipment (Achamrah et al., 2023), rule-based

simulation models for inventory control and collaborative transport network (Pan et al., 2014; Lafkihi et al., 2020), a single product inventory control with transshipment (Yang et al., 2017), and Physical Internet-enabled integrated production inventory and distribution (Ji et al., 2019; Peng et al., 2021). Therefore, this research aims to fill these gaps.

2.5. Solving 2e-VRP

The integration of all the aforementioned constraints presents substantial challenges in problem-solving. One of the most promising research areas promotes using stochastic and model-based methods to solve optimization problems with discrete decision variables (Vishnoi, 2021; Chen et al., 2018). Unlike solution-based approaches such as LNS and genetic algorithms, which derive new candidate solutions by exploring the vicinity of preceding solutions, model-based techniques produce candidate solutions from a probabilistic model, specifically a parameterized sampling distribution. This model's parameters are subsequently updated, drawing upon the function evaluations of earlier candidate solutions (Zhang et al., 2022; Chen, 2015). These algorithms are not only faster but also exhibit superior computational efficiency. They have been demonstrated to converge to the optimum with a probability of one (Vishnoi, 2021; Zhang et al., 2022). For instance, Chen et al. (2018) and Chen (2015) proposed a framework for reformulating discrete problems into continuous ones and then using gradient-based stochastic search to solve them. In addition, the authors applied their framework to solve selected benchmarks of deterministic TSP.

Of the few applications on stochastic and discrete problems, Zhang et al. (2022) proposed gradient-based simulation-optimization algorithms that solve large-scale discrete convex problems under uncertainties. More particularly, the algorithms are applied to solve an optimal allocation problem under the stochastic arrival of customers and a separable convex minimization problem. To take full advantage of model-based methods and gradient-based search in terms of fast convergence and global exploration search, our paper is among the first to use such an approach to address a concrete problem in logistics optimization. For this, we propose a new gradient-descent-based optimization framework that allows us to reformulate our stochastic and discrete problem as a differentiable program and then apply the first-order gradient to optimize the stochastic version of the reformulated program.

3. Problem description

This paper aims to optimize transportation costs for delivery and pickup in a closed-loop supply chain. It consists of multiple suppliers, customers, and satellites to consolidate, through cross-docking, the forward and reverse flows of reusable containers. The focus is on the cross-docking terminals that transfer containers between the echelons' vehicles. Furthermore, the distribution network is turned into a hyperconnected network of satellites using conventional interfaces. Products are stored and distributed in containers. As a result, the suppliers and customers have significantly more options: suppliers can distribute their products throughout the network, while retailers have more supply options as requested (see Figure 1). For example, such a setting is used by La Poste in France and Velove in Sweden to merge several actors and optimize their network as one single entity (Leveque et al., 2021; Velove, 2019).

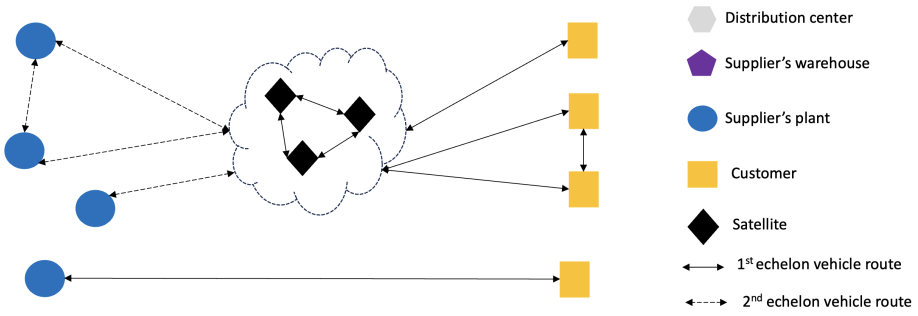

Figure 1: Containers flows in the closed loop supply chain under study

The two-echelon distribution network functions as outlined below. In the first echelon, routes connect suppliers to designated satellites with set capacities, where cross-docking services cater to both full and empty container flows. Traditional vehicles, like trucks, transport full containers from suppliers to satellites while picking up empty ones from satellites for return to suppliers. In the second echelon, zero-emission vehicles, such as cargo bikes, move full containers from satellites to customers. They also collect empty containers from these customers and return them to the satellites. Furthermore, the transportation infrastructure accommodates direct shipping for full and empty containers.

Vehicles have the flexibility of revisiting a source node multiple times within a single tour. Also, some routes may visit multiple suppliers and be managed by one vehicle. Additionally, destinations might receive their product needs through several split shipments, necessitating multiple vehicles. This leads to a potential series of operations at every site. A vehicle's stop is not solely dictated by the node it visits; it also involves a predetermined set of activities at each location. Hence, pickup and delivery can be facilitated by multiple vehicles at a single site, with the possibility of vehicles making multiple stops at identical locations. Therefore, the visited node and the associated time event define each vehicle stop.

Regarding inventory, an initial stock of full and empty containers is available at supplier locations. Our mathematical model accounts for stochastic customer demand in terms of full containers. Satellites are visited to cater to these customer demands. A similar strategy applies to empty containers. Additionally, both supplier and satellite nodes feature two distinct storage sections: one for full containers and another for empty ones. Each storage section comes with its initial inventory level and maximum capacity. Moreover, satellites can transship between one another, authorizing them to meet each other's inventory requirements. This facilitates sharing empty containers, preventing potential shortages and averting excessive spending on new container purchases.

Finally, in addition to the previously mentioned assumptions, we consider the following:

1. Players are not required to share any information. The assumption is that there is an orchestrator, an impartial coordinator, who coordinates and manages operations. For instance, this orchestrator can be a 4PL or a "trustee," also responsible for allocating profits. A control tower may also handle logistics activities, including management, execution, and monitoring. Several industrial applications can be found in (Pan et al., 2019).
2. Time window restrictions for delivering and picking up loaded/empty containers should be satisfied.
3. Each vehicle's maximum capacity and service time must not be exceeded.

4. The system has enough empty containers to satisfy demand.
5. All damaged containers are either repaired or replaced.

4. A deterministic mathematical formulation

For clarity, this section presents a deterministic formulation of the 2e-VRP under consideration, substantially extending the problem formulation of previous works (Dondo et al., 2009, 2011) in which a deterministic multi-level VRP with cross-docking and direct shipments is addressed. New variables and constraints are added to integrate pickup and delivery of loaded and empty containers at each supplier, satellite, and customer, flows consolidations, routing constructions, and transshipment operations at satellite nodes.

The problem is modeled on a graph $G = (L, A)$, where $L = S \cup K \cup N$ encompasses suppliers, satellites (cross-docks), and customer locations, respectively. The arc set A represents routes connecting suppliers to satellites and satellites to customer zones. Nodes in S denote suppliers distributing products to satellites represented by K and subsequently to customer locations N. Each arc $(i, j) \in A$ connecting nodes (i, j) carries an associated transportation cost c_{ij} based on distance and a time of travel d_{ij}.

Further contextualizing the problem, we have the set P, denoting the products encased in containers. These products travel from suppliers, are consolidated at satellites, and finally reach customers. The set V stands for the vehicles, with trucks or similar vehicles employed for the first echelon and cargo bikes utilized for the second echelon, facilitating product delivery to their final destinations. Given that any cumulative shipment should not eclipse a vehicle's carrying capacity, both the weight uw_p and volume uvo_p of each container p are imperative to our calculations, just as the weight qw_v and volume capacities qvo_v of each vehicle v are.

Furthermore, vehicles are stationed at and operate from specific depots at supplier sites or satellite locations. Let Dep_v designate the possible depot locations for a particular vehicle v. Also, specific customer zones, represented as i, are generally serviced by predefined satellites and are accessed by cargo bikes, represented by $(V_i \subset V)$. These bikes embark on their delivery routes from these designated satellite locations.

Using the notations, parameters, and decision variables reported in Table 1, the deterministic 2e-VRP with forward and reverse flows is formulated as follows:

$$\min \sum_{v \in V} Ct_v + \sum_{v \in V} \sum_{i,j \in L} \sum_{t \in T_i} \alpha_v z_{tijv} \tag{1}$$

Subject to:

$$\sum_{i \in Dep_v} \sum_{t \in T_i} y_{tiv} \leq 1 \quad \forall v \in V \tag{2}$$

$$\sum_{v \in V_i} y_{tiv} \leq 1 \quad \forall t \in T_i, i \in L \tag{3}$$

$$\sum_{v \in V_i} y_{tiv} \geq \sum_{v \in V_i} y_{t'iv} \quad \forall t, t' \in T_i, t < t', i \in L, \tag{4}$$

Table 1: Model Notations

Sets	
L	Nodes (all echelons)
V	Vehicles (all echelons)
L_v	Nodes serviceable by vehicle v (all echelons)
V_i	Vehicles able to visit node i (all echelons)
Dep_v	Possible depots for vehicle v (all echelons)
S	Suppliers ($S \subset L$)
K	Satellites ($K \subset L$)
N	Customers ($N \subset L$)
T	Events (i.e., vehicle stops)
T_i	Events specific to node $i \in N$
P	Containers
Parameters	
c_{ij}	Cost of transportation from node i to node j
α_v	Cost of deploying vehicle v, associated with the actual movement or operation of the vehicle
M_c	Upper bounds on transportation cost
D_{pi}	Demand for loaded container p at customer node i
$I_{pi}^{0,l}$	Initial inventory of loaded container p at source i
$I_{pi}^{f,l}$	Final inventory of loaded container p at source i
$I_{pi}^{i,e}$	Initial inventory of empty container p at source i
$I_{pi}^{f,e}$	Final inventory of empty container p at source i
w_v	Weight capacity of vehicle v
vo_v	Volume capacity of vehicle v
$utmax_{ip}$	Load/unload time for container p at node i
uw_p^l	Loaded container p weight
uvo_p^l	Loaded container p volume
uw_p^e	Empty container p weight
uvo_p^e	Empty container p volume
M_L	Upper bounds on load - capacity
a_i, l_i	Node i's earliest and latest service times
β_i	Stop time at node i
M_v	Upper bounds on the number of stops for v
$tmax_v$	Maximum route time for vehicle v
d_{ij}	Travel time between nodes i and j
$utmax_{ip}$	Load/unload time for container p at node i
Variables	
y_{tiv}	1 if vehicle v visits node i at event t, 0 otherwise
z_{tijv}	1 if arc (i,j) is traversed by vehicle v during event t, 0 otherwise
q_{tip}^l	Additional loaded containers p received at node i after event t
q_{tip}^e	Additional empty containers p received at node i after event t
Q_{tipv}^l	Loaded containers p on vehicle v at stop (t,i) from source i
Q_{tipv}^e	Empty containers p on vehicle v at stop (t,i) from source i
A_{tipv}^l	Loaded containers p delivered by vehicle v to node i at stop (t,i)
A_{tipv}^e	Empty containers p delivered by vehicle v to node i at stop (t,i)
Qt_{tipv}^l	Cumulative loaded containers p on vehicle v up to stop (t,i)
Qt_{tipv}^e	Cumulative empty containers p on vehicle v up to stop (t,i)
At_{tipv}^l	Cumulative loaded containers p delivered by vehicle v up to stop (t,i)
At_{tipv}^e	Cumulative empty containers p delivered by vehicle v up to stop (t,i)
C_{ti}	Travel cost to node i from start up to stop (t,i)
Ct_v	Total travel cost for vehicle v
T_{ti}	Travel time to node i from start up to stop (t,i)
Tt_v	Total travel time for vehicle v

$$\sum_{j\in L_v, i\neq j} z_{tijv} + \sum_{j\in L_v, i\neq j} z_{tjiv} = 2y_{tiv} \qquad \forall i \in L_v, v \in V, t \in T_i \tag{5}$$

$$\sum_{i\in\mathscr{E},\, j\in\mathscr{E}, i\neq j} x_{tijv} \leq \sum_{i\in\mathscr{E}} y_{tiv} - y_{t\iota v} \quad \forall \mathscr{E} \subseteq L_v, \iota \in \mathscr{E}, v \in V, t \in T_i \tag{6}$$

$$\sum_{i\in L_v} \sum_{t\in T_i} y_{tiv} \leq M_v \sum_{i'\in Dep_v} \sum_{t\in T_{i'}} y_{ti'v} \quad \forall v \in V \tag{7}$$

$$C_{ti} \geq \sum_{i'\in L, i'\neq i} \sum_{t'\in T_{i'}} c_{i'i} y_{t'i'v} - M_c(1 - y_{tiv}) \quad \forall v \in V_i, t \in T_i, i \in L \tag{8}$$

$$C_{t'i} \geq C_{ti} - M_c(2 - y_{tiv} - y_{t'iv}) \quad \forall v \in V_i, t, t' \in T_i, t < t', i \in L \tag{9}$$

$$Ct_v \geq C_{ti} + \sum_{i'\in Dep_v} \sum_{t'\in T_{i'}} c_{i'i} y_{t'i'v} - M_c(1 - y_{tiv}) \quad \forall v \in V, t \in T_i, i \in L \tag{10}$$

$$\begin{aligned} T_{ti} \geq & \sum_{i'\in L, i'\neq i} \sum_{t'\in T_{i'}} d_{i'i} y_{t'i'v} + \beta_{i'} + \sum_{p\in P} utmax_{i'p}(Q^l_{ti'pv} + Q^e_{ti'pv}) - M_c(1 - y_{tiv}) \\ & \forall v \in V, i \in L, t \in T_i \end{aligned} \tag{11}$$

$$\begin{aligned} T_{t'i} \geq & T_{ti} + \beta_i + \sum_{p\in P} utmax_{ip}(Q^l_{tipv} + A^l_{tipv} + Q^e_{tipv} + A^e_{tipv}) \\ & - M_c(2 - y_{tiv} - y_{t'iv}) \quad \forall v \in V_i, t, t' \in T_i, t < t', i \in L \end{aligned} \tag{12}$$

$$\begin{aligned} Tt_v \geq & T_{ti} + \beta_i + \sum_{p\in P} utmax_{ip}(Q^l_{tipv} + A^l_{tipv} + Q^e_{tipv} + A^e_{tipv}) \\ & + \sum_{i'\in L, i'\neq i} \sum_{t'\in T_{i'}} d_{ii'} y_{t'i'v} - M_c(1 - y_{tiv}) \quad \forall v \in V, t \in T_i, i \in L \end{aligned} \tag{13}$$

$$a_i \leq T_{ti} \leq b_i \quad \forall t \in T_i, i \in L \tag{14}$$

$$Tt_v \leq tmax_v \quad \forall v \in V \tag{15}$$

$$\sum_{v\in V_i} \sum_{t\in T_i} Q^l_{tipv} \leq I^{0,l}_{pi} \quad \forall i \in S, p \in P \tag{16}$$

$$\sum_{v\in V_i} \sum_{t\in T_i} Q^e_{tipv} \leq I^{0,e}_{pi} \quad \forall i \in S, p \in P \tag{17}$$

$$\sum_{v\in V_i} \sum_{t'\in T_i t' \leq t} Q^l_{t'ipv} \leq I^{0,l}_{pi} + q^l_{tip} \quad \forall t \in T_i, i \in K, p \in P \tag{18}$$

$$\sum_{v\in V_i}\sum_{t'\in T_i t'\le t} Q^e_{t'ipv} \le I^{0,e}_{pi} + q^e_{tip} \quad \forall t\in T_i, i\in K, p\in P \tag{19}$$

$$\sum_{v\in V_i}\sum_{t\in T_i} Q^l_{tipv} \le I^{0,l}_{pi} + \sum_{v\in V_i}\sum_{t\in T_i} A^l_{tipv} - I^{f,l}_{pi} \quad \forall i\in K, p\in P \tag{20}$$

$$\sum_{v\in V_i}\sum_{t\in T_i} Q^e_{tipv} \le I^{0,e}_{pi} + \sum_{v\in V_i}\sum_{t\in T_i} A^e_{tipv} - I^{f,e}_{pi} \quad \forall i\in K, p\in P \tag{21}$$

$$\sum_{v\in V_i}\sum_{t\in T_i} A^l_{tipv} \ge D_{pi} \quad \forall i\in N, p\in P \tag{22}$$

$$Q^l_{tipv} \le M_L y_{tiv} \quad \forall t\in T_i, i\in (S\cup K), p\in P, v\in V_i \tag{23}$$

$$Q^e_{tipv} \le M_L y_{tiv} \quad \forall t\in T_i, i\in (K\cup N), p\in P, v\in V_i \tag{24}$$

$$A^l_{tipv} \le M_L y_{tiv} \quad \forall t\in T_i, i\in S, p\in P, v\in V_i \tag{25}$$

$$A^e_{tipv} \le M_L y_{tiv} \quad \forall t\in T_i, i\in S, p\in P, v\in V_i \tag{26}$$

$$A^l_{tipv} \le D_{pi} y_{tiv} \quad \forall t\in T_i, i\in N, p\in P, v\in V_i \tag{27}$$

$$\sum_{i\in S\cup K}\sum_{t\in T_i} Q^l_{tipv} = \sum_{i\in N\cup K}\sum_{t\in T_i} A^l_{tipv} \quad \forall p\in P, v\in V \tag{28}$$

$$\sum_{i\in N\cup K}\sum_{t\in T_i} Q^e_{tipv} = \sum_{i\in S\cup K}\sum_{t\in T_i} A^e_{tipv} \quad \forall p\in P, v\in V \tag{29}$$

$$Qt^l_{t'ipv} + Qt^e_{t'ipv} \ge Qt^l_{tipv} + Q^l_{t'ipv} + Qt^e_{tipv} + Q^e_{t'ipv} - M_L(2 - y_{tiv} - y_{t'iv})$$
$$\forall v\in V_i, t, t'\in T_i, t<t', i\in L, p\in P \tag{30}$$

$$At^l_{t'ipv} + At^e_{t'ipv} \ge At^l_{tipv} + A^l_{t'ipv} + At^e_{tipv} + A^e_{t'ipv} - M_L(2 - y_{tiv} - y_{t'iv})$$
$$\forall v\in V_i, t, t'\in T_i, t<t', i\in L, p\in P \tag{31}$$

$$Q^l_{tipv} \le Qt^l_{tipv} \le \sum_{i'\in S\cup K}\sum_{t'\in T'_i} Q^l_{t'i'pv} \quad \forall v\in V, t\in T_i, i\in L_v, p\in P \tag{32}$$

$$Q^e_{tipv} \le Qt^e_{tipv} \le \sum_{i'\in N\cup K}\sum_{t'\in T'_i} Q^e_{t'i'pv} \quad \forall v\in V, t\in T_i, i\in L_v, p\in P \tag{33}$$

$$A^l_{tipv} \le At^l_{tipv} \le \sum_{i'\in S\cup K}\sum_{t'\in T'_i} A^l_{t'i'pv} \quad \forall v\in V, t\in T_i, i\in L_v, p\in P \tag{34}$$

$$A^e_{tipv} \le At^e_{tipv} \le \sum_{i'\in S\cup K}\sum_{t'\in T'_i} A^e_{t'i'pv} \quad \forall v\in V, t\in T_i, i\in L_v, p\in P \tag{35}$$

$$\sum_{p\in P} uw_p^l(Qt_{tipv}^l - At_{tipv}^l) + uw_p^e(Qt_{tipv}^e - At_{tipv}^e) \leq w_v$$
$$\forall v \in V, t \in T_i, i \in L_v, p \in P \tag{36}$$
$$\sum_{p\in P} uvo_p^l(Qt_{tipv}^l - At_{tipv}^l) + uvo_p^e(Qt_{tipv}^e - At_{tipv}^e) \leq vo_v$$
$$\forall v \in V, t \in T_i, i \in L_v, p \in P \tag{37}$$
$$Qt_{tipv}^l - At_{tipv}^l \geq 0 \quad \forall v \in V, t \in T_i, i \in L_v, p \in P \tag{38}$$
$$Qt_{tipv}^e - At_{tipv}^e \geq 0 \quad \forall v \in V, t \in T_i, i \in L_v, p \in P \tag{39}$$

$$q_{t'ip}^l \geq q_{tip}^l + \sum_{v\in V_i} A_{t'ipv}^l \quad \forall t, t' \in T_i, t < t', i \in K, p \in P \tag{40}$$
$$q_{t'ip}^e \geq q_{tip}^e + \sum_{v\in V_i} A_{t'ipv}^e \quad \forall t, t' \in T_i, t < t', i \in S \cup K, p \in P \tag{41}$$

$$\sum_{v\in V_i} A_{tipv}^l \leq q_{tip}^l \leq \sum_{v\in V_i} \sum_{t'\in T_i'} A_{t'ipv}^l \quad \forall t \in T_i, i \in K, p \in P \tag{42}$$
$$\sum_{v\in V_i} A_{tipv}^e \leq q_{tip}^e \leq \sum_{v\in V_i} \sum_{t'\in T_i'} A_{t'ipv}^e \quad \forall t \in T_i, i \in S \cup K, p \in P \tag{43}$$

The objective function (1) is to minimize the overall transportation cost, encompassing both fixed and variable components. Constraints (2) ensure that every vehicle v, when utilized, begins and ends its route at the designated base node $i \in Dep_v$. Constraints (3) specify that a predefined event t at node i, represented by the vehicle stop (t, i), can only be assigned to one vehicle v. Constraints (4) mandate that the stop (t', i) cis allocated to vehicle v only if all earlier stops (t, i) at node i, with (t<t′), are already assigned to vehicles. Constraints (5) and (6) define degree and sub-tour elimination constraints. Constraints (7) allow vehicle v to serve multiple stops (t, i) on different nodes provided it has previously been assigned to a base node i'. Constraints (8) dictate that the minimal cost to reach any node i should at least equal the direct travel cost from node i' to node i, given by $c_{i'i}$. Constraints (9) ensure vehicle v can make consecutive stops at node i, while event t always occurs before t′ (t<t′). Constraints (10) account for the cost from the last visited node i on the v-trip to the base depot i'. Constraints (11) determine the minimum time needed to arrive at the first node, combining the travel time for arc (i', i) and the total service time at the base node i'. Constraints (12) demand that for multiple stops by vehicle v at node i, event t should precede t′ when t<t′. Constraints (13) define the duration each vehicle v spends on its journey. Constraints (14) and (15) define the time window and maximum service time. Constraints (16) and (17) restrict the total container p quantity suppliers to not exceed the initial inventory at node i. Constraints (18) and (19) ensure that all containers p taken from satellite i up to event t do not exceed the sum of the initial inventory and the number of containers p received from other sources until event t. Constraints (20) and (21) define the overall container balance at each satellite i. Constraints (22) stipulate the delivery of container p to every customer node i satisfies their demand. Constraints (23) and (24) guarantee that a vehicle v can only pick up at source node i during stop (t, i), if $y_{tiv} = 1$. Constraints (25)– (27) state that a delivery task by vehicle v at stop (t, i) is feasible only if this stop is allocated to v. Constraints (28) and (29) define

the overall container balance for each vehicle v. Constraints (30) quantify the cumulative containers p picked up by vehicle v up to the stop (t, i). Constraints (31) denote the total amount of container p delivered by vehicle v up to (t, i). Constraints (32)– (35) define bounds on variables Qt^l_{tipv}, Qt^e_{tipv}, At^l_{tipv} and At^e_{tipv}. Constraints (36)-(39) define vehicle capacity constraints. Constraints (40) and (41) specify the added inventory of container p received at cross-dock i up to stop (t',i). Constraints (42) and (43) define bounds on variables q^l_{tip} and q^e_{tip}.

5. Stochastic and continuous transformation of the deterministic model

The transformation of our optimization problem from a discrete framework, suitable for deterministic demands, to a continuous one, adaptable to stochastic variability, is a nuanced process. This section delves into the methodologies involved in this transition, ensuring that the continuous framework remains relevant and feasible for the original discrete problem.

5.1. Stochastic formulation of the problem

Central to our approach is the reformulation of the deterministic model to accommodate stochastic demands by introducing random variables $\mathcal{D}$. Accordingly, as shown in Expression (45), the objective function shifts from a deterministic focus to minimizing expected costs over various independent demand realization scenarios, noted Ω. Therefore, the stochastic model can be written as:

$$\hat{x} \in argmin_{x \in \chi} \mathcal{OF}(x) \tag{44}$$

With :

$$\mathcal{OF}(x) := \mathbb{E}_{\Omega}\left[OF(x, \mathcal{D})\right] \tag{45}$$

And $\hat{x}$ represents the optimal solution, $x = [x_1, x_2, ..., x_n]^T$ the solution vector and n the problem's dimensionality. The solution space, denoted as χ, is a non-empty and finite subset of $\mathbb{R}^n$. Specifically, each component x_i is confined to a finite set $x_i^1, x_i^2, ..., x_i^{m_i}$, with m_i marking the dimension of the solution space for the *ith* coordinate (i.e., a decision variable). Moreover, the objective function, $\mathcal{OF}(\cdot)$, is a deterministic function with real values, and it is consistently defined over the domain χ.

5.2. Model transformation

The methodology for transitioning to a continuous model involves several critical steps, guided by principles of gradient-based optimization and adaptive stochastic search (Zhou and Hu, 2014; Chen et al., 2018). We employ a robust optimization technique within the parameter space of a probability density model. This involves generating solutions tailored to each demand scenario and re-calibrating parameters based on the objective function evaluations through a stochastic gradient descent algorithm. The process includes converting the discrete problem into a continuous problem and iteratively refining potential solutions and parameters, as described in Algorithm (1).

Algorithm 1 Overview of the steps involved in the transformation

1. Initialize parameters within the boundaries of a probability density model.
2. **Repeat until convergence:**
   (a) Generate potential solutions based on the parameterized distribution.
   (b) Update parameters using a stochastic gradient descent algorithm to converge to the optimal solution.

In the following sections, we further detail each of these steps.

5.2.1. Probabilistic model formulation and parameter space

The introduction of a probabilistic model, parameterized by θ, is a pivotal step. This model effectively captures the potential solutions within the stochastic landscape, reflecting the inherent variability in each demand scenario. The treatment of solution variables as random variables within this model allows for a more nuanced approach to stochastic optimization.

To elaborate further, let us consider $f(x;\theta)$ which represents a family of probability density functions parameterized by θ, for a given realization scenario. Here, x is an element of the solution space χ, while the parameter θ belongs to a subset Θ of $\mathbb{R}^d$, and d denotes the dimensionality of the parameter space.

The parameter space Θ is defined to facilitate the application of this probabilistic model as follows:

$$\Theta = \left\{ \theta \in [0,1]^{\sum_{i=1}^{n} m_i}, \sum_{j=1}^{m_i} \theta^{ij} \ \forall i = 1..., n \right\} \tag{46}$$

Since it is simpler to sample from the independent distribution, this paper uses an independent probabilistic model on the solution space. This model has also been employed in many model-based approaches showing solid empirical performances (Hu et al., 2007). Another justification for employing an independent distribution is the principle of "localization", frequently invoked for dimension reduction across diverse fields such as filtering and communication networks (Chen et al., 2018). Accordingly, within the defined parameter space and for a given realization scenario, the probabilistic model $f(x;\theta)$ takes the form:

$$f(x;\theta) = \prod_{i=1}^{n} \left(\sum_{j=1}^{m_i} \theta^{ij} \mathbb{I}\{x_i = x_i^j\} \ \theta \in \Theta \right) \tag{47}$$

With θ^{ij} is the probability that the ith component of the problem's solution takes a value x_i^j, and $\mathbb{I}$ the indicator function. For example, an $f(y_{tiv};\theta^{tiv})$ can represent the probability θ^{tiv} that the vehicle v visits a node i at the event t (i.e., $y_{tiv} = 1$), and $f(q^l_{tip};\theta^{tip})$ the probability that transshipped quantity of loaded containers received at node i after the event t equals to a value q^l_{tip}.

5.2.2. Continuous expression of the optimization problem

With the probabilistic model and parameter space defined as continuous, the objective function $\mathcal{OF}(\hat{x})$ becomes bounded by the inequality (48), involving $f(x;\theta)$. This transformation leads to the derivation of a novel objective function that is both continuous and differentiable with respect to the parameter θ. It is worth noting that equality is attained when an

optimal parameter $\hat{\theta}$ exists, causing the probability function $f(x;\hat{\theta})$ to be concentrated solely on a subset of the optimal solution set.

$$\mathcal{OF}(\hat{x}) \leq \sum_{x\in\chi} \mathcal{OF}(x) f(x;\theta) \quad \forall\theta \in \Theta \tag{48}$$

Given the Expression (48), the minimization problem is reformulated into a continuous one, within the parameter space Θ, as follows:

$$\hat{\theta} \in argmin_{\theta\in\Theta} \sum_{x\in\chi} \mathcal{OF}(x) f(x;\theta) = argmin_{\theta\in\Theta} \mathbb{E}_{f(\cdot;\theta)}[\mathcal{OF}(\chi)] \tag{49}$$

Finally, we adapt a gradient-based optimization algorithm to our stochastic continuous model. This adaptation is crucial for facilitating the swift adjustment of parameter θ, thus leading to convergence towards optimal solutions. This approach leverages the strengths of stochastic gradient descent and robust optimization techniques, offering a comprehensive solution to complex stochastic problems. In the following, we delve into more details of the technique we use.

5.2.3. Stochastic gradient descent for parameter updating and solving the continuous minimization problem

To solve the reformulated version of the problem (noted $RF(\theta)$) under stochastic demands, we relied on the ADAptive Moment estimation-based algorithm ($ADAM$) developed in Kingma and Ba (2014) and supported by Google Deepmind. More specifically, this paper conducts the following two steps for each demand realization scenario:

1. Generate candidate solutions based on the probabilistic model $f(x;\theta)$.
2. Use $ADAM$ to update the parameter θ to enhance solutions.
3. Use a projection method to ensure the updated parameters remain within the feasible region (see Section(5.2.4))

The $ADAM$ algorithm optimizes stochastic objective functions solely based on first-order gradients, utilizing adaptive estimations of lower-order moments. $ADAM$ is easy to implement and computationally efficient with minimal memory demands. It remains invariant to diagonal re-scaling of the gradients, making it particularly adept for problems with extensive datasets or parameters. Moreover, the algorithm's hyper-parameters are intuitively comprehensible and generally require minimal fine-tuning (Kingma and Ba, 2014). Further, in diverse contexts, $ADAM$ has been applied and consistently demonstrates a regret bound on its convergence rate that aligns with, if not surpasses, the best-known outcomes within the online convex optimization framework. Finally, empirical results underscore $ADAM$'s compelling performance in real-world scenarios, showcasing its superior capabilities compared to other stochastic optimization techniques.

As described in Algorithm (2), our resolution algorithm allows us to minimize the expected value of $RF(\theta)$ under stochastic demands, $\mathbb{E}[RF(\theta)]$, with respect to the parameters θ. Let $RF_1(\theta), ...RF_\Omega(\theta)$ be the realization of the stochastic function for each scenario $1, ...\Omega$.

Moreover, for each realization scenario ω, we generate candidate solutions within a given number of iterations. The objective is to bring into play the advantages of population-based methods in exploring the solution space compared to the algorithms that use a single candidate solution for each iteration (Zhou and Hu, 2014).

Let $x_{k,\omega}$ be the candidate solutions to the original problem for a scenario ω, and in iteration k generated according to $f(x_{k,\omega}; \theta_{k,\omega})$. And let $g_{k,\omega} = \nabla_{k,\theta} RF_{k,\omega}(\theta_{k,\omega})$, be the gradient, i.e., the vector of partial derivatives of $RF_{k,\omega}$, with respect to $\theta_{k,\omega}$ evaluated for each scenario ω and iteration k.

Furthermore, in this algorithm, we use exponential moving averages for both the gradient, represented as $m_{k,\omega}$, and the squared gradient, denoted $v_{k,\omega}$. These averages are regulated by hyper-parameters $\phi_1; \phi_2$, both lying within the range $[0, 1]$, which determine the decay rates of the respective moving averages. Essentially, $m_{k,\omega}$ gives an estimate of the gradient's first moment (or the mean), while $v_{k,\omega}$ captures its second raw moment (essentially the uncentered variance).

Algorithm 2 The solution method employed in this paper

Require: Step size σ
Require: Exponential decay rates $\phi_1, \phi_2 \in [0, 1]$ for moment estimates
Require: Initial parameter vector $\theta_{0,\omega}$ for each scenario ω

$m_{0,\omega} \leftarrow 0$ ▷ Initialize 1st moment vector
$v_{0,\omega} \leftarrow 0$ ▷ Initialize 2nd moment vector
$k \leftarrow 0$ ▷ Initialize iteration counter
repeat
  repeat
    $k \leftarrow k + 1$
    Generate candidate solutions using $f(x_{k,\omega}; \theta_{k,\omega})$
    $g_{k,\omega} \leftarrow \nabla_\theta RF_{k,\omega}(\theta_{k-1,\omega-1})$ ▷ Get gradients for scenario ω at iteration k
    $m_{k,\omega} \leftarrow \phi_1 m_{k,\omega} + (1 - \phi_1) g_{k,\omega}$ ▷ Update biased first moment estimate
    $v_{k,\omega} \leftarrow \phi_2 v_{k-1,\omega-1} + (1 - \phi_2) g^2_{k,\omega}$ ▷ Update biased second moment estimate
    $\hat{m_{k,\omega}} \leftarrow m_{k,\omega} / (1 - \phi_1^{k,\omega})$ ▷ Correct bias for first moment estimate
    $\hat{v_{k,\omega}} \leftarrow v_{k,\omega} / (1 - \phi_2^{k,\omega})$ ▷ Correct bias for second moment estimate
    $\theta'_{k,\omega} \leftarrow \theta_{k-1,\omega-1} - \sigma \hat{m_{k,\omega}} / (\sqrt{\hat{v_{k,\omega}}} + \epsilon)$ ▷ Update $\theta'_{k,\omega}$, the parameter vector, not yet projected onto the feasible region (Θ')
    $\theta_{k,\omega} \leftarrow \text{Project}(\theta'_{k,\omega}, \Theta')$ ▷ Apply Algorithm (3) to project the intermediate vector onto Θ', and ensure it meets the problem's constraints
  until maximum iterations reached
  $\omega \leftarrow \omega + 1$ ▷ Advance scenario counter
until $\theta_{k,\omega}$ converges (i.e., any improved solution, in a certain number of sequential iterations, is returned) or maximum scenarios reached
$\theta_{best} \leftarrow \theta_{k,\omega}$ ▷ Set best parameter vector to final iteration's vector
return Solutions x derived from $f(x; \theta_{best})$

Ensuring the feasibility of solutions, derived from $f(x; \theta_{best})$, throughout the optimization process is a critical aspect of the stochastic and continuous transformation approach. Indeed, while *ADAM* iteratively improves the solution by moving in the direction of the steepest descent, it does not inherently account for the feasibility with respect to the defined constraints. To address this, as shown in Algorithm (2), we integrate a projection method immediately following each parameter update, $\theta'_{k,\omega}$, to maintain the integrity of our solutions within the feasible region of the parameter space.

When the stopping criteria are met, the algorithm stops, and the current best parameter,

denoted as θ_{best}, is retrieved, and the corresponding solution, x, is generated using f. Otherwise, the process proceeds to step 6. The stopping criteria are considered satisfied if either any improved solution is found within a specified number of consecutive iterations, or the maximum number of scenarios has been reached.

In the following, we describe the projection method.

5.2.4. Feasibility of solutions

In the optimization process using Algorithm (2), we encounter a crucial step where the updated parameter vector, denoted as $\theta'_{k,\omega}$, must be adjusted to meet the problem's constraints. In the case of the canonical simplex, these constraints may require that these parameters all be non-negative and sum to one.

The adjustment is made through a projection method. This projection, as described in (Chen, 2015) and presented in Algorithm (3), maintains the feasibility of solutions by adjusting $\theta'_{k,\omega}$ to ensure they stay within the feasible region, Θ', defined by the constraints. This step involves sorting the elements of the parameter vector and then adjusting them according to a computed threshold value (τ) that ensures all constraints are satisfied. The outcome is a feasible parameter vector $\theta_{k,\omega}$, ready for the next iteration and guaranteed to remain within the simplex's boundaries. Thus, ensuring that the solution, x, generated using θ_{best} will be the feasible solution that provides the best performance in terms of minimizing the objective function, taking into account the constraints and stopping criteria, as described in Algorithm (2).

For instance, let us consider Constraints (2). In our discrete model, these constraints ensure that each vehicle v can be assigned to at most one event at a depot. Translated into the continuous framework, this becomes a probabilistic representation where θ^{tiv} denotes the probability of vehicle v attending event v at depot i. Thus, ensuring feasibility in the continuous setting involves projecting parameter updates onto a canonical simplex, making sure that probabilities sum up to 1 (Chen, 2015).

In addition, when it comes to interactions with other constraints, such as Constraints (4), the projection method ensures that any adjustments made to satisfy Constraints (2) do not lead to the violation of Constraints (4). Indeed, if the adjustment for Constraints (2) pushes the parameters outside the feasible region of Constraints (4), the projection method must then reconcile this by finding the closest point within the feasible region that satisfies both constraints, as described in Algorithm (3).

Algorithm 3 Projection onto the Feasible Region

1: **Require** $\theta'_{k,\omega} \in \mathbb{R}^{m_i}$ ▷ Updated parameter vector from *ADAM* (Algorithm (2), Line 13)
2: **Require** Θ' ▷ Canonical simplex, representing the feasible region of the solution space
3: **Sort** the elements of $\theta'_{k,\omega}$ to obtain $\theta'^{,(1)}_{k,\omega}, \ldots, \theta'^{,(m_i)}_{k}$ in ascending order
4: **Compute** the cumulative sum $S_j = \sum_{l=j}^{m_i} \theta'^{,(l)}_{k,\omega}$ for $j = 1, \ldots, m_i$
5: **Find** the largest j such that $\theta'^{,(j)}_{k,\omega} + \frac{1}{m_i - j + 1}(1 - S_j) > 0$ ▷ To determine the threshold τ
6: **Compute** $\tau = \frac{1}{m_i - j + 1}(1 - S_j)$ ▷ Using the j found in the previous step
7: **Update** $\theta^{(i)}_{k,\omega} = \max\{\theta'^{,(i)}_{k,\omega} - \tau, 0\}, \quad \text{for } i = 1, \ldots, m_i$ ▷ The parameter vector
8: **Output:** $\theta_{k,\omega}$

6. Experimental design and results

In this section, we initially outline the data generation and parameter fine-tuning processes that were incorporated into our methodology. Subsequently, we provide a detailed description of the three experiments we conducted. The first experiment focuses on assessing the quality and speed of our proposed solution method, the second experiment examines the performance of the distribution settings and the resolution algorithm, and the third experiment involves assessing the robustness of each model to enable the generalization of our findings.

6.1. Experimental design

This section covers three integral components of our experimental design, namely the data generation process, the fine-tuning of parameters, and the experimental setup.

6.1.1. Data generation

We generated the dataset randomly as no benchmarks were available in the 2e-VRP and hyperconnected networks literature with the same constraints. To ensure a realistic dataset, we followed some of the standards for instances generated for the 2e-VRP, as described in (Sluijk et al., 2022; Dondo et al., 2009), and for the production-inventory-distribution problem in the hyperconnected network, as given in (Ji et al., 2019; Peng et al., 2021).

In our data generation process, we created a dataset consisting of 1000 instances, with the parameter N (denoting the number of customers) taking on values of 10, 20, 30, up to 100. This enabled us to create 10 sets, each corresponding to a specific N value (e.g., the first set with $N = 10$, the second with $N = 20$, and so on). Within each set, we generated 100 instances, varying the remaining parameters as outlined in Table 2.

Table 2: Data generation patterns

Sets	
Set of suppliers S	U(2,20)
Set of satellites K	U(3,10)
Set of vehicles V	100 (60 conventional vehicles, 40 cargo bikes)
Set of possible depots, for vehicle v, Dep_v	U(2,11)
Set of events, for each customer node i, T_i	2
Set of containers P	U(1,10)
Parameters	
Cost of transportation c_{ij}	U(1.5,3) ($ per km)
Cost of using vehicle α_v	U(20,500) ($)
Demand D_{pi}	Random variable associated to the stochastic demand following N(100, 20); Normal distribution $N(\mu, \sigma)$
Initial Inventory of loaded containers $I_{pi}^{0,l}$	U(0,2600), uniform distribution, per unit of loaded container
Final Inventory of loaded containers $I_{pi}^{f,l}$	U(0,600) per unit of loaded container
Initial Inventory of empty containers $I_{pi}^{0,e}$	U(0,2600) per unit of empty container
Final Inventory of empty containers $I_{pi}^{f,e}$	U(0,600) per unit of empty container
Weight capacity w_v	U(5, 20000) (kg)
Volume capacity vo_v	U(1,35) (m^3)
Maximum route time $tmax_v$	10 hours
Weight of loaded container uw_p^l	U(1,6) kg per unit of loaded container
Volume of loaded container uvo_p^l	U(0.005,0.01) m^3 per unit of loaded container
Weight of empty containeruw_p^e	U(1,4) kg per unit of empty container
Volume of empty containeruvo_p^e	U(0.005,0.01) m^3 per unit of loaded container
Load/unload time $utmax_{ip}$	U(200,300) of unit of loaded/empty container per hour
Earliest service time a_i	U(0.4,5) (hour)
Latest service time l_i	U(0.4,6) (hour)
Stop time β_i	U(0.5,1) (hour)
Average speed	U(25,90) km per hour
Distance	U(100,300)(km) between suppliers, between satellites, and between customers
Distance	U(300,500)(km) from suppliers to satellites, and from satellites to customers

6.1.2. Parameter fine-tuning

We used the Irace package developed by López-Ibáñez et al. (2016) for tuning the parameters of the robust optimization algorithm. The software package employs automated configuration procedures, including the iterated F-race algorithm. Furthermore, following the data generation process previously described, we generated a separate set of representative instances for the parameter tuning. Specifically, we created 4 sets with 15, 30, 50, and 100 customers each, resulting in 400 instances. These instances were exclusively used to tune the parameters to ensure the robustness and generalizability of our model. Table 3 reports the parameters tuning adopted.

Table 3: Parameters tuning for our solution method using Irace package

Parameters	Chosen value
Number of demand realization scenarios	100
Number of iterations	100
Step size σ	0.001
Exponential decay rate ϕ_1	0.9
Exponential decay rate ϕ_2	0.999

Regarding the parameterized distribution f, in practice, it should be chosen based on the prior knowledge of the structure of the objective function. Since no prior knowledge is available, we set the initial distribution $f(x_{0,\omega}, \theta_{0,\omega})$ to be the uniform distribution on the solution space, as in (Chen et al., 2018).

Finally, all the optimization steps are carried out on a personal computer (MacBook Pro, macOS Monterey, Apple M1 Pro, 32 GB of RAM). We implemented the proposed algorithms using CPLEX 12.9 (academic version) and Python 3.10.

6.1.3. Setup for the first experiment: Insights into the performance of the solution method

To get insights into the robustness of our solution method, we compare the results with those obtained using Sample Average Approximation (SAA) as an exterior sampling method. As described in (50), SAA approximates the expected objective function $\mathbb{E}[OF(x)]$ under stochastic demands by the sample average function (Kleywegt et al., 2002). Accordingly, the problem is solved using a deterministic optimization algorithm (in our case, the Branch-and-Cut solver of CPLEX).

$$\frac{1}{\Omega}\sum_{\omega=1}^{\Omega} OF(x,\omega) \tag{50}$$

In SAA, Ω independent samples, each of size L, are generated, and the related deterministic problem is solved (see Section (4)). Let $\eta_L^1, \eta_L^2, ..., \eta_L^\Omega$ and $\hat{x}^1, \hat{x}^1, ..., \hat{x}^\Omega$ be the objective function values and candidate solutions respectively. Let $\overline{\eta}_L$ be the average of the optimal objective function values. $\overline{\eta}_L$ provides a statistical estimate for a lower bound on the optimal value of the true problem (Kleywegt et al., 2002; Achamrah et al., 2022a). In addition, the variance of this estimator can be calculated as follows:

$$\hat{\nu}^2_{\overline{\eta}_L} = \frac{1}{(\Omega-1)\Omega}\sum_{\omega=1}^{\Omega}(\eta_L^\omega - \overline{\eta}_L)^2 \tag{51}$$

Moreover, for any feasible solution $\hat{x}$, the statistical upper bound, $\hat{\eta}_{L'}(\hat{x})$, for the optimal value is expressed as follows (Verweij et al., 2003; Mohammadi Bidhandi and Patrick, 2017):

$$\hat{\eta}_{L'}(\hat{x}) = \frac{1}{L'}\sum_{\omega=1}^{L'} OF(\hat{x},\omega) \tag{52}$$

where L' is a sample size chosen to be quite large, $L' > L$. In addition, the variance of this estimator can be calculated as follows:

$$\hat{\nu}^2_{\hat{\eta}_{L'}(\hat{x})} = \frac{1}{(L'-1)L'}\sum_{\omega=1}^{L'}(OF(\hat{x},\omega) - \hat{\eta}_{L'}(\hat{x}))^2 \tag{53}$$

For the entire problem, the optimal solution $\hat{x}^*$ is determined as follows:

$$\hat{x}^* \in argmin\{\hat{\eta}_{L'}(\hat{x}); \hat{x} \in \{\hat{x}^1, \hat{x}^1, ..., \hat{x}^{\Omega}\}\} \tag{54}$$

The quality of the solution $\hat{x}^*$ is evaluated by calculating the optimality gap estimate $\mathcal{O}$ (Kleywegt et al., 2002; Verweij et al., 2003; Achamrah et al., 2022a; Mohammadi Bidhandi and Patrick, 2017):

$$\mathcal{O} = \hat{\eta}_{L'}(\hat{x}^*) - \overline{\eta}_L \tag{55}$$

The estimate variance of $\mathcal{O}$ is computed as follows (Kleywegt et al., 2002; Verweij et al., 2003):

$$\hat{\nu}^2 = \hat{\nu}^2_{\hat{\eta}_{L'}(\hat{x}^*)} + \hat{\nu}^2_{\overline{\eta}_L} \tag{56}$$

In this experiment, we applied SAA to solve the proposed model, with the parameters $\Omega = 4; L = 50; L' = 100$. Without loss of generality, we run tests on ten instances generated as per the data generation process previously described. Furthermore, to compare the results with those obtained using our newly proposed algorithm, we computed the optimality gap estimate $\mathcal{O}$ and its estimate variance $\hat{\nu}^2$ for the objective values obtained using each solution method. It is worth noting that we used the lower bounds estimates computed with SAA. In addition, we report CPU time in seconds for each resolution algorithm.

6.1.4. Setup for the second experiment: Impact of the proposed model on the distribution network performance

In this experiment, we aim to better understand the potential advantages our proposed model, with its enhanced collaboration, flexibility, and openness, can bring to urban delivery systems (Cleophas et al., 2019). We evaluate its effectiveness against the dedicated system (DS) model described in (Yang et al., 2017). Within this system, each supplier independently manages delivery routes, either with their own fleet or through an exclusively contracted one. Figure (2) shows that this network represents a multi-echelon hierarchical configuration. It encompasses upstream levels (like plants and warehouses) overseen by suppliers and downstream sites (such as distribution centers and retail sales points) managed by retailers. It falls upon retailers to define and refine their segment of the logistics framework. Throughout the network, products are also shipped using reusable containers owned by suppliers. Similarly, the system also integrates the pickup of empty containers from customer areas, ensuring they are returned to the original suppliers for subsequent utilization. In addition, the DS model does not consider the costs associated with contracting fleets nor inventory sharing through transshipment. All other assumptions outlined in this paper remain valid within the DS model. Finally, the primary decisions in this model revolve around minimizing transportation costs, and determining the quantity and timing of deliveries and pickups under stochastic demand and time window constraints.

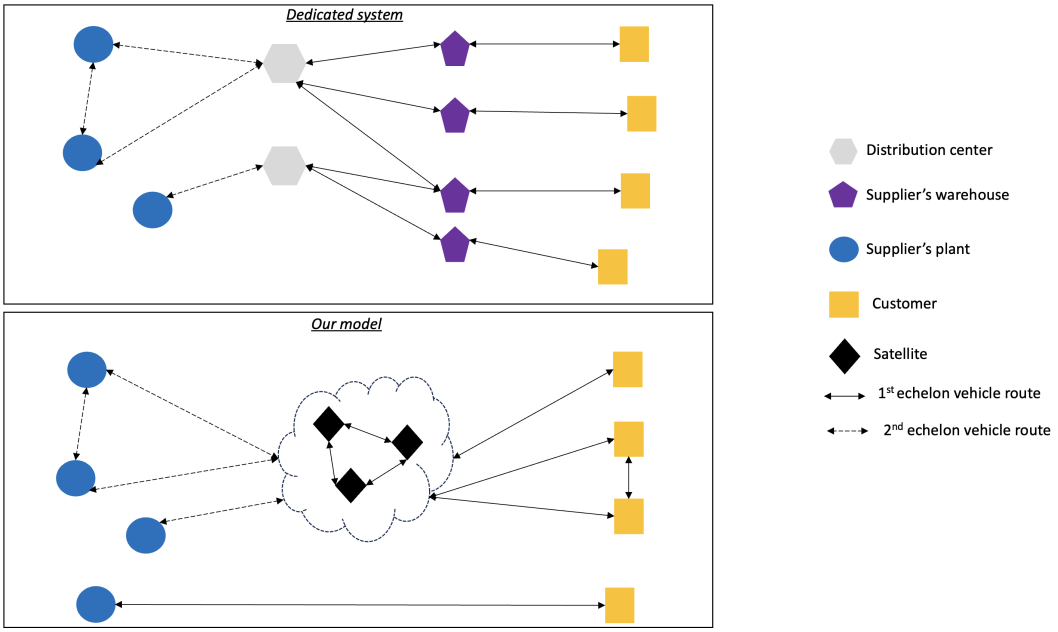

Figure 2: Dedicated system vs. our model

To compare the results, we run tests using 10 sets generated as described in Subsection (6.1.1). In addition, we used cost saving, capacity utilization, number of used vehicles, and service time as the primary key performance indicators (KPIs). The following ratios to represent the results are computed:

The performance ratio $\mathscr{C}$ related to cost saving:

$$\mathscr{C} = \frac{TotalCost_{DS} - TotalCost_{2eVRP}}{TotalCost_{2eVRP}} 100 \tag{57}$$

The performance ratio $\mathscr{U}$ related to the vehicle -capacity- utilization (VU):

$$\mathscr{U} = \frac{VU_{2eVRP} - VU_{DS}}{VU_{2eVRP}} 100 \tag{58}$$

The performance ratio $\mathscr{R}$ related to the number of vehicles used (NV):

$$\mathscr{R} = \frac{NV_{DS} - NV_{2eVRP}}{NV_{2eVRP}} 100 \tag{59}$$

The performance ratio $\mathscr{S}$ related to service time (ST):

$$\mathscr{S} = \frac{ST_{DS} - ST_{2eVRP}}{ST_{2eVRP}} 100 \tag{60}$$

To assess the performance of the resolution algorithm, we also report the CPU times for each model and instance under consideration.

6.1.5. Setup for the third experiment: Insights into the robustness of the model setting and the resolution algorithm

To generalize the findings, we assess the robustness of each model setting as per the data generation process described in Subsection (6.1.1). The objective is to evaluate each model

with regard to its responsiveness to uncertainty and sensitivity to demand profiles. This paper considers the following scenarios of demand realizations:

- Scenario S1 (best case): $D_{pi} \in [\mu - 2\sigma, \mu - \sigma]$
- Scenario S2: $D_{pi} \in [\mu - \sigma,\ \mu]$
- Scenario S3: $D_{pi} \in [\mu,\ \mu + \sigma]$
- Scenario S4 (worst/stressing case) : $D_{pi} \in [\mu + \sigma,\ \mu + 2\sigma]$

6.2. Experimental results and managerial insights

This section reports the results of the three experiments we conducted to assess the performance of the proposed algorithm and our model in terms of savings, flexibility, and openness.

6.2.1. Results of the first experiment: Insights into the performance of the solution method

Table 4 summarizes our algorithm's and SAA's comparison results for all instances under consideration. It reports the optimality gap estimate $\mathcal{O}$ and its estimate variance $\hat{\nu}^2$ computed for each solution method. It also reports CPU time in seconds.

Table 4: Comparison results of SAA and our resolution algorithm

Instance					$\mathcal{O}$ (%)		$\hat{\nu}^2$ (%)		CPU time (s)	
#	Suppliers	Satellites	Customers	Containers	SAA	Our algorithm	SAA	Our algorithm	SAA	Our algorithm
1	2	3	15	1	0.23	0	0.01	0	1322	12
2	2	3	15	2	0.24	0	0.01	0	5494	25
3	2	3	15	3	0.26	0.01	0.01	0	7692	38
4	2	3	15	4	0.88	0.01	0.03	0	12325	67
5	2	3	15	5	5.34	0.01	0.43	0	>18000	94
6	2	3	15	6	7.22	0.01	3.52	0	>18000	111
7	2	3	15	7	11.32	0.02	5.02	0.01	>18000	130
8	2	3	15	8	15.01	0.04	8.96	0.01	>18000	124
9	2	3	15	9	22.66	0.02	11.2	0.01	>18000	186
10	2	3	15	10	27.43	0.02	18.31	0.01	>18000	152

From Table 4, we observe that our solution method is more robust in providing high-quality solutions with less computational effort than SAA for all instances under consideration. The solutions obtained are within an average estimated optimality gap of less than 0.01% compared to 9.06% for SAA. The same goes for the estimated variance of this gap. Our algorithm provides estimates, on average, less than 0.01% compared to 4.75% for SAA. Moreover, the table shows that our algorithm requires less computational effort (on average 93.84 seconds) than SAA. The reason is that, unlike SAA, our solution method does not need more samples to converge as it combines the advantage of the fast convergence speed of continuous algorithms (i.e., gradient descents) with the insensitivity to the problem's size and the global exploration of stochastic search methods.

6.2.2. Results of the second experiment: Impact of the proposed model on the distribution network performance

Figure (3) reports the average total transportation costs computed for our model and DS settings. In addition, Figure (4) provides the average CPU time in seconds, computed for each setting. Finally, Figure (5) provides $\mathscr{C}$, $\mathscr{U}$, $\mathscr{R}$, and $\mathscr{S}$ in % computed with respect to DS for all instances under consideration. Finally, a t-test is performed to verify the statistical differences among the solutions computed for our and the DS models ($p - value < 0.0001$).

Results in Figures (3), (4), and (5) highlight that the average total costs obtained using the proposed model are the lowest: the average total cost is reduced by 25%. This implies that

this model, unlike DS, enables exploiting a high level of integration, flexibility, and openness. In other words, cost savings follow as the distribution becomes more flexible. Furthermore, the distribution flexibility is extended by enabling lateral transshipment of filled and empty containers, implying better service quality.

As for DS, frequent transportation is required to reach the same performance level as in our model, which implies increasing transportation costs. Indeed, because the DS network is hierarchical, decentralized, and fragmented, transportation is inefficient, even with lower quantities of products that must be shipped. That is, fulfilling stochastic demands efficiently and under time constraints would be much more challenging.

In addition, results show that the proposed model minimizes resource utilization $\mathscr{R}$ (on average by 16%, see Figure (5)) while increasing vehicle capacity utilization $\mathscr{U}$ (on average 21%), resulting in considerable cost savings $\mathscr{C}$ (on average 25%, see Figure (5)). Accordingly, our model allows for more efficient transportation of containers within the network while implicitly considering the negative impact on the environment. Also, at satellite levels, loaded/empty container flows from various streams can be re/deconsolidated multiple times. Therefore, both echelons' underused vehicle capacity may be better utilized without exerting unnecessary effort. Additionally, the cost savings increases with the network size.

Moreover, our model takes advantage of the interconnectedness and synergy among the network levels to consolidate the fragmented and overlapping transportation flows of both loaded and empty containers. As a result, more nodes are visited, and heterogeneous vehicle fleets are effectively managed. Also, given the small capacity of cargo bikes, compared to traditional vehicles, the routes are likely to be limited in length, restricting the service time (duration)-and distance of a route. Cargo bikes may, therefore, be utilized for multiple trips while keeping a lower carbon footprint. In addition, integrating warehousing, cross-docking, and transshipment operations at satellites enables efficient fulfillment of all actors' needs when initial stocks of full and empty containers are insufficient to meet customer demand within time windows. On the other hand, when the vehicle capacity is insufficient, the proposed model becomes less advantageous, implying the generation of several routes, including more than a single tour with multiple stops.

Furthermore, containers are reusable and may be shared among players to increase their utilization rates. Thus, containers are more effectively managed in terms of investment level and utilization rate compared to the DS model. Moreover, loaded/empty containers may be shipped to destinations by employing all open hyperconnected satellites and other logistical resources. This enhances transportation efficiency and stock turnover, reduces implicit inventories, and results in total cost savings. Accordingly, in our model, the service time $\mathscr{S}$ is reduced (on average 25%, see Figure (5)) since the loading and unloading of containers is faster due to their modular and standard characteristics. Also, when compared to DS, the consolidation of container flows and their (trans)shipment to destinations by vehicles occurs with little to no delay. Therefore, results stress that the model's flexibility and openness directly reduce order-cycle time and indirectly reduce storage space requirements.

Regarding the solution method's performance, we notice from Figure 4 that it is robust and well-suited for solving such combinatorial problems. It allows for consistently providing solutions on different instance sizes within a reasonable CPU time; for DS, on average, it took 356 seconds and 221 seconds for our model.

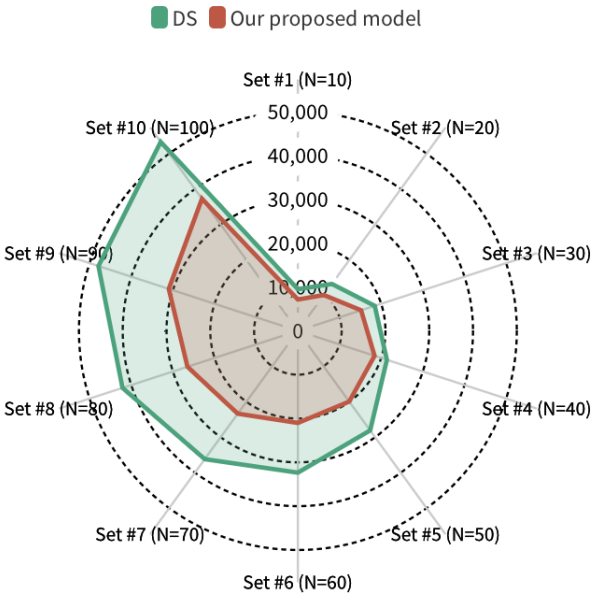

Figure 3: Average total cost computed for each distribution setting

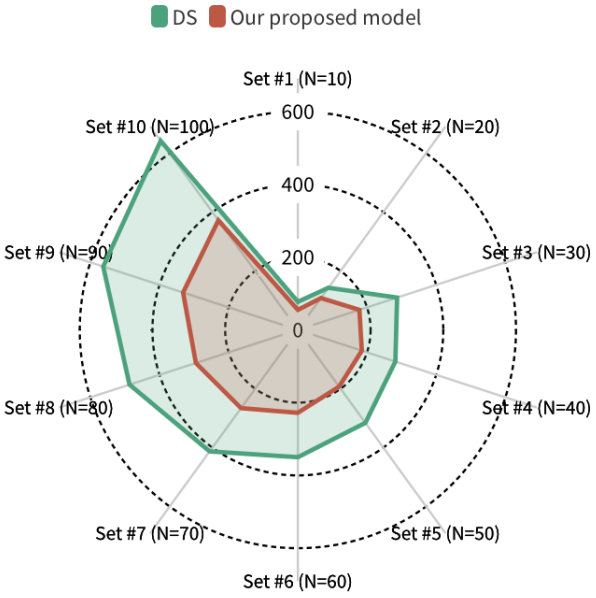

Figure 4: Average CPU time (s) computed for each distribution setting

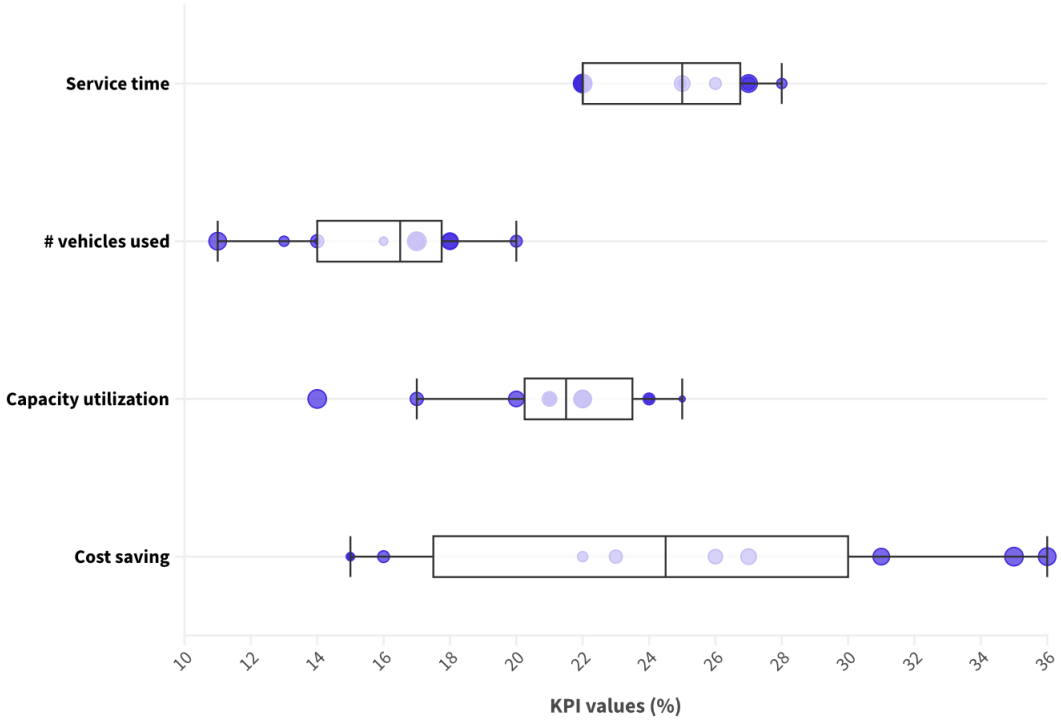

Figure 5: Key performance indicators computed with respect to DS, for all the sets under consideration

6.2.3. Results of the third experiment: Insights into the robustness of the model setting and the resolution algorithm

In this section, we compute the total cost for each of the four scenarios under consideration. Without loss of generality, we run tests on the first 3 sets. For each scenario, we compare the corresponding average total cost (TC) to the solution obtained in each scenario and each setting. Figure 6 presents the results of the comparison.

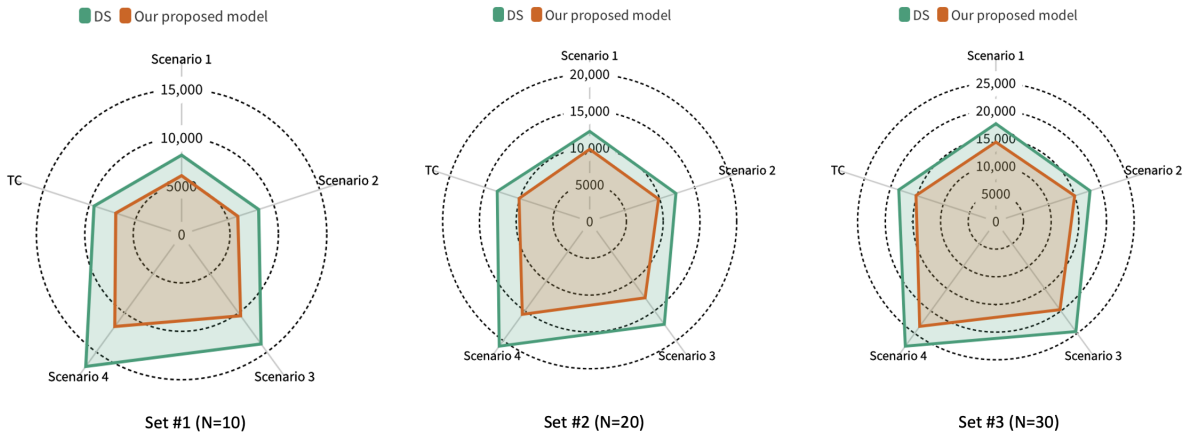

Figure 6: Results of comparison of the total costs

From Figure 6, we notice that the total cost in each scenario is highly sensitive to the de-

mand variation for both DS and our model. We also observe that the total cost in DS increases significantly with the size of the sets as the distribution network becomes highly stressed to keep up with high demand. This highlights the flexibility and robustness of our model, and enables us to navigate uncertainty thanks to its hyperconnected distribution network.

To demonstrate the actual performance of the solution method, let TC^r be a robust solution to our model and DS model, which corresponds to the minimum recovery distance to all optimal solutions of the expected scenarios (Goerigk and Schöbel, 2011). Also, the following metric noted Dev is computed as follows:

$$Dev = |TC - TC^r| \tag{61}$$

Figure 7 depicts the variation of TC and TC^r for each set.

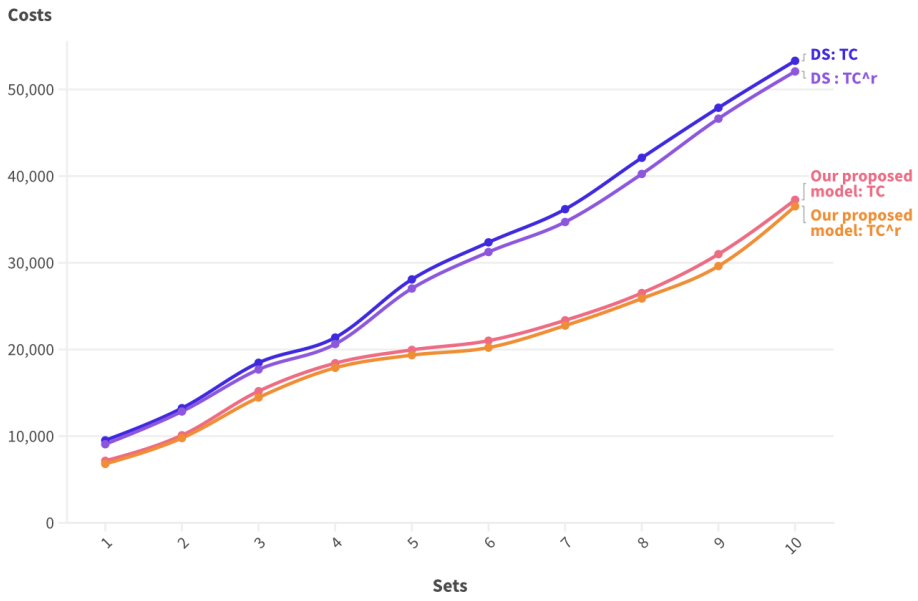

Figure 7: Robustness metric for each model setting

For the robustness assessment of each model setting, we notice that TC is slightly higher than TC^r for DS compared to our model, as each model's resolution does not require the same treatment in terms of optimality and feasibility. In addition, for both configurations, the proposed algorithm allows us to find near-robust solutions as it combines the advantage of good values with lower deviation. Therefore, the results show that our algorithm highlights the trade-off between good solution quality and robustness under uncertainty for DS and the proposed model.

7. Conclusions & perspectives

This paper investigates a two-echelon cargo bike distribution system with pickups and deliveries of reusable containers in hyperconnected networks. The paper introduces and models deterministic and stochastic 2e-VRPs with container forward and reverse flows. Also, it considers the consolidation of empty and loaded container flows at satellite nodes, transshipment operations, and load splitting. Moreover, it takes into account time windows and service

time constraints. Furthermore, to deal with the combinatorial complexity of the model, we propose a gradient-descent-based optimization-based framework to solve it. Furthermore, we conduct experiments to highlight the performance of our solution method and demonstrate the advantages of the proposed model compared to DS.

Experimental results highlight that, unlike DS in which suppliers optimize delivery routes served by their own or exclusively contracted fleet of vehicles, the proposed model allows for economies of scale thanks to its high level of integration, flexibility, and openness. In addition, our model allows for creating robustness against uncertainty for all instances under consideration. Moreover, the results stress that distribution flexibility is further enhanced by enabling lateral transshipment and flow consolidation of loaded and empty containers. On the other hand, due to the limited capacity of cargo bikes, the proposed model becomes less advantageous, as it implies the generation of several routes, including more than a single tour with multiple stops, to meet customers' demand within time windows and service time constraints. Finally, the results highlight the good performance of the proposed algorithm in terms of robustness and providing high-quality solutions with reduced CPU times.

The practical application of our proposed model can be realized through a systematic approach that aligns with the hyperconnected network paradigm described in our paper. Decision-makers can implement this model by first establishing a network of satellite nodes that function as consolidation and transshipment points, as outlined in Section (3). The stochastic nature of customer demand, a key feature of our model, can be addressed by utilizing historical data and advanced forecasting techniques to generate the demand scenarios described in Section (5.1). The gradient-descent-based optimization framework presented in Section (5.2) can be integrated into existing logistics management systems, allowing for real-time route optimization that accounts for both forward and reverse flows of containers. As demonstrated in our experimental results (Section (6.2.2)), the model's ability to reduce total costs by an average of 25% while improving vehicle utilization by 21% provides a clear incentive for practical adoption. Companies can implement this model incrementally, starting with basic two-echelon routing and gradually incorporating more complex features such as transshipment and flow consolidation. The robustness of the model to different demand scenarios, as shown in Section (6.2.3), further enhances its applicability in volatile urban logistics environments.

Since the 2e-VRP in hyperconnected networks is very complex, it leaves room for improvement regarding the integration of information and inventory management at each echelon, the limited storage capacity of satellites, uncertainty in travel time, and the design of efficient policies for costs and information sharing, and to deal with the case where demands are dynamic and stochastic. Other disruptive events, such as vehicle delays, maintenance breaks, and unserviceable satellites, could also be investigated. Finally, further supporting the findings would be interesting by applying the model and the resolution method to real data and using other parameterized distribution models.

Disclosure statement

No potential competing interest is reported by the authors.

Data availability statement

The data that support the findings of this study are available upon request.

Acknowledgement

The funding body will be acknowledged after peer review.

The authors thank the AE and reviewers for their constructive comments and encouragements that have helped improve the paper greatly.

References

Achamrah, F. E., Lafkihi, M., and Ballot, E. (2023). A dynamic and reactive routing protocol for the physical internet network. *International Journal of Production Research*, 0(0):1–19.

Achamrah, F. E., Riane, F., Bouras, A., and Sahin, E. (2020). Collaboration mechanism for shared returnable transport items in closed loop supply chains. In *ICORES*, pages 247–254.

Achamrah, F. E., Riane, F., and Limbourg, S. (2022a). Spare parts inventory routing problem with transshipment and substitutions under stochastic demands. *Applied Mathematical Modelling*, 101:309–331.

Achamrah, F. E., Riane, F., Sahin, E., and Limbourg, S. (2022b). An artificial-immune-system-based algorithm enhanced with deep reinforcement learning for solving returnable transport item problems. *Sustainability*, 14(10):5805.

Anderluh, A., Hemmelmayr, V. C., and Nolz, P. C. (2019). Chapter 8 - sustainable logistics with cargo bikes—methods and applications. In Faulin, J., Grasman, S. E., Juan, A. A., and Hirsch, P., editors, *Sustainable Transportation and Smart Logistics*, pages 207–232. Elsevier.

Anderluh, A., Nolz, P. C., Hemmelmayr, V. C., and Crainic, T. G. (2021). Multi-objective optimization of a two-echelon vehicle routing problem with vehicle synchronization and 'grey zone' customers arising in urban logistics. *European Journal of Operational Research*, 289(3):940–958.

Bakach, I., Campbell, A., and Ehmke, J. (2021). A two-tier urban delivery network with robot-based deliveries. *Networks*, 78.

Ballot, E., Montreuil, B., and Meller, R. D. (2014). *The Physical Internet: The Network of Logistics Networks*.

Battarra, M., Cordeau, J., and Iori, M. (2014). Pickup-and-delivery problems for goods transportation. In *Vehicle Routing*.

Benjamin, G. (2020). *Vehicle Routing Problem for Multi-Product Cross-Docking*.

Berbeglia, G., Cordeau, J.-F., Gribkovskaia, I., and Laporte, G. (2007). Static pickup and delivery problems: A classification scheme and survey. *TOP: An Official Journal of the Spanish Society of Statistics and Operations Research*, 15:1–31.

Breunig, U., Baldacci, R., Hartl, R. F., and Vidal, T. (2019). The electric two-echelon vehicle routing problem. *Computers & Operations Research*, 103:198–210.

Cattaruzza, D., Absi, N., Feillet, D., and González-Feliu, J. (2017). Vehicle routing problems for city logistics. *EURO Journal on Transportation and Logistics*, 6(1):51–79.

Chen, P., Guo, Y., Lim, A., and Rodrigues, B. (2006). Multiple crossdocks with inventory and time windows. *Computers & Operations Research*, 33:43–63.

Chen, X. (2015). *New model-based methods for non-differentiable optimization*. University of Illinois at Urbana-Champaign.

Chen, X., Zhou, E., and Hu, J. (2018). Discrete optimization via gradient-based adaptive stochastic search methods. *IISE Transactions*, 50(9):789–805.

Cleophas, C., Cottrill, C., Ehmke, J. F., and Tierney, K. (2019). Collaborative urban transportation: Recent advances in theory and practice. *European Journal of Operational Research*, 273(3):801–816.

Cuda, R., Guastaroba, G., and Speranza, M. G. (2015). A survey on two-echelon routing problems. *Computers & Operations Research*, 55:185–199.

Dondo, R., Méndez, C. A., and Cerdá, J. (2009). Managing distribution in supply chain networks. *Industrial & engineering chemistry research*, 48(22):9961–9978.

Dondo, R., Méndez, C. A., and Cerdá, J. (2011). The multi-echelon vehicle routing problem with cross docking in supply chain management. *Computers & Chemical Engineering*, 35(12):3002–3024.

Enthoven, D. L., Jargalsaikhan, B., Roodbergen, K. J., uit het Broek, M. A., and Schrotenboer, A. H. (2020). The two-echelon vehicle routing problem with covering options: City logistics with cargo bikes and parcel lockers. *Computers & Operations Research*, 118:104919.

European-Commission (2023). *Greening Freight Transport*. Publications Office of the European Union.

Fazili, M., Venkatadri, U., Cyrus, P., and Tajbakhsh, M. (2017). Physical internet, conventional and hybrid logistic systems: a routing optimisation-based comparison using the eastern canada road network case study. *International Journal of Production Research*, 55(9):2703–2730.

Gansterer, M. and Hartl, R. F. (2018). Collaborative vehicle routing: A survey. *European Journal of Operational Research*, 268(1):1–12.

Goerigk, M. and Schöbel, A. (2011). A scenario-based approach for robust linear optimization. pages 139–150.

Gonzalez-Feliu, J., Perboli, G., Tadei, R., and Vigo, D. (2008). The two-echelon capacitated vehicle routing problem.

Guide, D. and Van Wassenhove, L. (2009). The evolution of closed-loop supply chain research. *Operations Research*, 57:10–18.

Hu, J., Fu, M. C., and Marcus, S. I. (2007). A model reference adaptive search method for global optimization. *Operations research*, 55(3):549–568.

Hu, Z.-H., Sheu, J.-B., Zhao, L., and Lu, C.-C. (2015). A dynamic closed-loop vehicle routing problem with uncertainty and incompatible goods. *Transportation Research Part C: Emerging Technologies*, 55:273–297. Engineering and Applied Sciences Optimization (OPT-i) - Professor Matthew G. Karlaftis Memorial Issue.

Iassinovskaia, G., Limbourg, S., and Riane, F. (2017). The inventory-routing problem of returnable transport items with time windows and simultaneous pickup and delivery in closed-loop supply chains. *International Journal of Production Economics*, 183:570–582.

Ji, S.-f., Peng, X.-s., and Luo, R.-j. (2019). An integrated model for the production-inventory-distribution problem in the physical internet. *International Journal of Production Research*, 57(4):1000–1017.

Jie, W., Yang, J., Zhang, M., and Huang, Y. (2019). The two-echelon capacitated electric vehicle routing problem with battery swapping stations: Formulation and efficient methodology. *European Journal of Operational Research*, 272(3):879–904.

Kaboudani, Y., Ghodsypour, S. H., Kia, H., and Shahmardan, A. (2020). Vehicle routing and scheduling in cross docks with forward and reverse logistics. *Operational Research*, 20(3):1589–1622.

Kassem, S. and Chen, M. (2012). Solving reverse logistics vehicle routing problems with time windows. *The International Journal of Advanced Manufacturing Technology*, 68.

Kheirkhah, A. and Rezaei, S. (2015). Using cross-docking operations in a reverse logistics network design: a new approach. *Production Engineering*, 10.

Kim, N., Montreuil, B., Klibi, W., and Kholgade, N. (2021). Hyperconnected urban fulfillment and delivery. *Transportation Research Part E: Logistics and Transportation Review*, 145:102104.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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

Lafkihi, M., Pan, S., and Ballot, E. (2020). Rule-based incentive mechanism design for a decentralised collaborative transport network. *International Journal of Production Research*, 58(24):7382–7398.

Lee, Y. H., Jung, J., and Lee, K. (2006). Vehicle routing scheduling for cross-docking in supply chain. *Computers & Industrial Engineering*, 51:247–256.

Leveque, J., Klibi, W., and Stauffer, G. (2021). The price of hyperconnectivity in last mile delivery network design . In *8th International Physical Internet Conference PIC 2021*.

Lim, A., Miao, Z., and xu, Z. (2005). Transshipments through crossdocks with inventory and time windows. *Naval Research Logistics - NAV RES LOG*, 52:724–733.

López-Ibáñez, M., Dubois-Lacoste, J., Pérez Cáceres, L., Birattari, M., and Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43 – 58.

Ma, H., Miao, Z., Lim, A., and Rodrigues, B. (2011). Crossdocking distribution networks with setup cost and time window constraint. *Omega*, 39:64–72.

Marjani, M., moattar husseini, M., and Karimi, B. (2012). Bi-objective heuristics for multi-item freights distribution planning problem in crossdocking networks. *International Journal of Advanced Manufacturing Technology - INT J ADV MANUF TECHNOL*, 58.

Meller, R. D., Montreuil, B., Thivierge, C., and Montreuil, Z. (2012). Functional design of physical internet facilities: A road-based transit center. In *Progress in Material Handling Research*.

Mohammadi Bidhandi, H. and Patrick, J. (2017). Accelerated sample average approximation method for two-stage stochastic programming with binary first-stage variables. *Applied Mathematical Modelling*, 41:582–595.

Montreuil, B. (2011). Toward a physical internet: meeting the global logistics sustainability grand challenge. *Logistics Research*, 3(2):71–87.

Pan, S., Nigrelli, M., Ballot, E., Sarraj, R., and Yang, Y. (2014). Perspectives of inventory control models in the physical internet: A simulation study. *Computers & Industrial Engineering*, 84.

Pan, S., Trentesaux, D., Ballot, E., and Huang, G. Q. (2019). Horizontal collaborative transport: survey of solutions and practical implementation issues. *International Journal of Production Research*, 57(15-16):5340–5361.

Pan, S., Trentesaux, D., McFarlane, D., Montreuil, B., Ballot, E., and Huang, G. Q. (2021). Digital interoperability in logistics and supply chain management: state-of-the-art and research avenues towards physical internet. *Computers in Industry*, 128:103435.

Peng, X., Ji, S., Thompson, R. G., and Zhang, L. (2021). Resilience planning for physical internet enabled hyperconnected production-inventory-distribution systems. *Computers & Industrial Engineering*, 158:107413.

Perboli, G., Tadei, R., and Vigo, D. (2011). The two-echelon capacitated vehicle routing problem: Models and math-based heuristics. *Transportation Science*, 45(3):364–380.

Qiu, Y., Zhou, D., Du, Y., Liu, J., Pardalos, P. M., and Qiao, J. (2021). The two-echelon production routing problem with cross-docking satellites. *Transportation Research Part E: Logistics and Transportation Review*, 147:102210.

Roch, Y., Ballot, E., and Perraudin, X. (2014). A new framework for the management of returnable "containers" within open supply networks. *Studies in Computational Intelligence*, 594.

Sluijk, N., Florio, A. M., Kinable, J., Dellaert, N., and Van Woensel, T. (2022). Two-echelon vehicle routing problems: A literature review. *European Journal of Operational Research*.

Tarantilis, C. D., Anagnostopoulou, A. K., and Repoussis, P. P. (2013). Adaptive path relinking for vehicle routing and scheduling problems with product returns. *Transportation Science*, 47(3):356–379.

Velove (2019). The physical internet and containerised last mile delivery.

Verweij, B., Ahmed, S., Kleywegt, A. J., Nemhauser, G., and Shapiro, A. (2003). The sample average approximation method applied to stochastic routing problems: a computational study. *Computational optimization and applications*, 24(2):289–333.

Vishnoi, N. K. (2021). *Bridging Continuous and Discrete Optimization*, page 1–16. Cambridge University Press.

Wang, J., Jagannathan, A., Zuo, X., and Murray, C. (2017). Two-layer simulated annealing and tabu search heuristics for a vehicle routing problem with cross docks and split deliveries. *Computers & Industrial Engineering*, 112.

Yang, Y., Pan, S., and Ballot, E. (2017). Innovative vendor-managed inventory strategy exploiting interconnected logistics services in the physical internet. *International Journal of Production Research*, 55(9):2685–2702.

Yu, S., Puchinger, J., and Sun, S. (2020). Two-echelon urban deliveries using autonomous vehicles. *Transportation Research Part E: Logistics and Transportation Review*, 141:102018.

Yu, V. F., Jodiawan, P., Hou, M.-L., and Gunawan, A. (2021). Design of a two-echelon freight distribution system in last-mile logistics considering covering locations and occasional drivers. *Transportation Research Part E: Logistics and Transportation Review*, 154:102461.

Zhang, H., Zheng, Z., and Lavaei, J. (2022). Gradient-based algorithms for convex discrete optimization via simulation. *Operations Research*.

Zhang, L., Ding, P., and Thompson, R. G. (2023). A stochastic formulation of the two-echelon vehicle routing and loading bay reservation problem. *Transportation Research Part E: Logistics and Transportation Review*, 177:103252.

Zhou, E. and Hu, J. (2014). Gradient-based adaptive stochastic search for non-differentiable optimization. *IEEE Transactions on Automatic Control*, 59(7):1818–1832.

Zuluaga, J., Thiell, M., and Perales, R. (2016). Reverse cross-docking. *Omega*, 66.