Journal of Industrial and Systems Engineering
Vol. 17, No. 3, pp. 39-65
Summer (July) 2025

JISE

# Integrating Machine Learning and Optimization for Hierarchical Cross-Dock Scheduling in Online Retail Vehicle Routing with Time Windows

**Parastoo Heidarifar [a], Mohammad Mahdi Nasiri [b*], Fariborz Jolai [b], Matineh Ziari [b]**

[a] *Ph.D. Candidate, Department of Industrial Engineering, Kish International Campus, University of Tehran, Tehran, Iran*

[b] *School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran*

## Abstract

Advancements in technology, combined with rapid shifts in modern lifestyles and consumer behaviors, have driven a significant surge in online shopping, which now accounts for a substantial portion of the global market. To address the escalating online demands of today, online retailers are pursuing flexible and cost-effective supply chains that can be enhanced through cross-dock facilities. This study explores the application of the Vehicle Routing Problem (VRP) to schedule order deliveries for an online retailer, ensuring timely and accurate fulfillment to boost responsiveness and operational efficiency. Machine learning techniques, as a state-of-the-art tool, have advanced optimization in dynamic, real-time environments. The dataset employed in this research is derived from a large-scale online retailer. Using machine learning methods, a pairwise distance matrix is computed from the geographical longitude and latitude of all nodes, followed by dimensionality reduction via the K-Means clustering algorithm to form optimized regional groups. This refined matrix is then integrated into a newly developed Mixed Integer Programming (MIP) model, which handles supply and delivery scheduling, cross-dock operations, and vehicle assignments. The model incorporates customer-specified time frames, allowable delays, and varying service levels across customer segments. Exact solutions from the model demonstrate that adjustments to key parameters such as fleet size, vehicle capacity, and time-frame configurations yield outcomes that enhance profitability and overall success for the online retailer, as validated through sensitivity analyses.

**Keywords**: Clustering, Cross-docking, Machine learning, Online-retailer, Scheduling, Vehicle routing.

* Corresponding Author

## 1. Introduction

In the e-commerce market, adopting a Business-to-Customer (B2C) strategy that combines competitive pricing with efficient last-mile delivery services is crucial for achieving sustained success. The concept of cross-docking (CD), a logistics strategy, considerably reduces lead times and inventory costs by streamlining the flow of commodities from suppliers to customers without long-term storage. Cross-docking can reduce total transportation miles and the number of trucks required, enhancing operational efficiency (Ghomi et al., 2023). This approach allows for faster fulfillment of customer orders, making it particularly beneficial for online retailers facing rapid demand fluctuations. Amazon utilizes cross-docking to consolidate shipments, reduce transportation time and costs, and improve supply chain efficiency (Bodnár & Juhász, 2023). Cross-docking reduces last-mile costs by 8–12% due to fewer touch-points and optimized route planning (Pei et al., 2011). It is found that cross-docking reduces last-mile delivery time significantly by eliminating intermediate storage and handling delays (Rushton et al., 2014). The faster the order delivery, the higher the rate of demand and sales, while the increased demand raises the likelihood of delivery mistakes and consequently customer dissatisfaction (Parasuraman et al., 1988), along with cross-docking decreases last-mile delivery errors by 18% due to fewer handling stages (Boysen et al., 2019). Although cross-docking offers numerous advantages, it also presents challenges such as the need for precise coordination and scheduling of inbound and outbound shipments. The integration of routing decisions with cross-dock scheduling is complex and requires sophisticated mathematical models to optimize operations.

A significant advancement in logistics to optimize the flow of goods and address cross-dock operation to strategic complex challenges and leading to its optimal performance is Vehicle Routing Problem (VRP), its combination with CD and Time Windows (TW) called (VRPCDTW), this hybrid strategy is especially well-suited for the scheduling demands of online retail supply chains, which is the central focus of this research. The main objective of the VRPCDTW is to minimize overall expenses and time while adhering to restrictions and facilitating effective handling at the cross-dock (Hasani-Goodarzi & Tavakkoli-Moghaddam, 2012). By enhancing both cost efficiency and timely order fulfillment, VRPCDTW helps resolve the often-conflicting objectives of cost minimization and customer satisfaction, making it a valuable framework for addressing the following key research questions:

- How can ML techniques reduce the complexity of a large-scale real-time data of an online retail supply chain to make precise optimization decisions?
- Does a hierarchical cross-docking network enhance cost efficiency and delivery performance in large-scale, real-time e-commerce logistics systems?

- Does the implementation of unsupervised ML techniques lead to precise scheduling and optimization of VRPCDTW strategy applied in an online retail delivery process and maximize customer fulfillment?
- How does real-time adjustment of operational parameters, such as number of time frames, fleet size, and vehicle capacity, impact the feasibility and optimality of cross-docking scheduling decisions?

To address these research questions, this study proposes an integrated solution that combines Machine Learning (ML) techniques with a mathematical optimization model, aiming to overcome real-world logistics challenges of online retail supply chains and enhance these days competitiveness through cost effectiveness and improved customer satisfaction. The model focuses on the daily scheduling of an online retailer orders collected from multiple suppliers and transported by a homogeneous fleet of vehicles to a central cross-dock, where they are sorted, packed, and routed according to customer preferred time windows. In response to the complexity of questions related to order assignment, vehicle scheduling, capacity allocation, and distribution efficiency, ML techniques are employed to preprocess and structure large-scale spatial data by clustering geographically dispersed locations into representative groups, effectively simplifying computational complexity to achieve exact optimization solutions of the mathematical model within a short processing time.

The subsequence of paper is outlined as follows. In Section 2 an overview of existing scholarly works in the field is provided to shed light on the background of the problem based on the relevant subjects. The problem definition and mathematical formulation are detailed in Section 3. Section 4 is about the solution approach of this study, outlines the proposed clustering algorithm, while presents the results of computational experiments. Section 5 consists of a set of sensitivity analysis with numerical results following by highlighted key managerial insights. The paper closes in Section 6, where computational results are summarized and prospects for future researches are highlighted.

## 2. Literature review

In line with the growing demand for responsive and cost effective supply chains of online retailers and the increasing complexity of logistics networks; the efficiency of cross-docking is evident in its ability to optimize transportation, consolidate shipments, and improve supply chain responsiveness, as demonstrated by its successful implementation in companies like Amazon and Walmart (Bodnár & Juhász, 2023). The last decade has witnessed significant evolution in the integration of cross-docking with vehicle routing problems; the classical formulation of the VRP involves a single depot from which a fleet of vehicles departs to deliver goods to a set of customers. Since its inception by (Dantzig & Ramser, 1959), VRP has evolved through numerous variants. Among them, the integration with CD, introduced VRPCD by (Lee et al., 2006). The integration of customer satisfaction into VRPCD models, such as those by (Santos et al., 2020) and (Ghorbani et al., 2020), marked a shift from pure efficiency toward responsiveness and service quality with Time Windows (VRPTW) which has become increasingly relevant in recent decades. Solomon (1987) introduced hard time windows, which enforced strict scheduling constraints. Taillard et al. (1997) later proposed soft time windows, balancing service quality with operational cost.

Necula et al. (2017) laid a foundational step by applying Ant Colony Optimization for adaptive routing in VRPTW, enabling decentralized and heuristic adaptation to complex delivery constraints. This approach inspired further algorithmic refinements, such as the two-phased Genetic Algorithm employed by (Baniamerian et al., 2018) for efficient scheduling in cross-docking terminals, enhancing responsiveness under strict time constraints. Küçükoğlu and

Öztürk (2019) developed a hybrid algorithm combining Genetic Algorithms with Simulated Annealing, improving convergence rates and solution robustness in VRPTW instances. The literature began to shift focus from pure optimization to system-wide coordination, with (Qian et al., 2020) highlighting the logistical challenges of multi-modal coordination and real-time data integration. By 2020, research shifted toward the entrance of ML into VRPCD research. Poullet (2020) applied clustering and reinforcement learning to address scalability issues in large-scale routing. Deep reinforcement learning (DRL) was then introduced by (Sultana et al., 2021), achieving dynamic routing adaptability. Zhou et al. (2021) quantified these benefits, showing up to 22% improvement in on-time delivery using DRL-based strategies. Advancing from traditional heuristic and metaheuristic techniques to sophisticated ML solutions; Tirkolaee et al. (2021) integrated ML with heuristic methods to reduce environmental impacts and cost in VRPTW. While Gunawan et al. (2022) further advanced reverse logistics using a metaheuristic to optimize multi-period cross-docking, complementing the branch-and-price models of Wölck and Meisel (2022) which tackled soft time window constraints for improved delivery flexibility. This progression reflects a shift from cost-efficiency and constraint satisfaction to real-time adaptability, sustainability, and customer satisfaction, Ghasemi et al. (2022) used Graph Neural Networks (GNNs) for predictive scheduling, improving forecast precision in routing. Kamble et al. (2023), and Lima et al. (2023) expanded the application scope of ML across warehouse analytics, smart logistics, and inventory optimization, aligning efficiency with customer service. In 2024, the field explored more personalized and stochastic approaches. Huang et al. (2023) introduced a PSO–retrospective approximation model for time-windowed replenishment. Meanwhile, Iklassov et al. (2024) tackled uncertainty in VRPTW using DRL, while Zhang et al. (2025) examined scalable personalized delivery windows, and Nozari et al. (2025) analyzed the barriers of ML scalability in real-world logistics. Additionally, Sippel and Forbes (2024) presented fragment-based exact algorithms for VRPTW that show promise in generating strong linear relaxations. The most recent breakthroughs of 2025 push the boundaries with quantum computing in logistics problem-solving. Osaba et al. (2024) explored quantum approaches in pickup and delivery variants of the VRPTW, demonstrating the potential of quantum-based methods beyond classical formulations. Building on this, Holliday et al. (2025) applied hybrid quantum annealing to real-time VRPTW optimization, achieving approximately a 3.86% optimality gap while maintaining adherence to time window constraints.

This research addresses critical gaps identified in the existing literature on VRPCD, particularly in the context of large-scale, real-time e-commerce logistics networks. While prior studies have explored cross-docking models, vehicle routing with time windows, or machine learning-based clustering individually, few have provided an integrated, scalable solution that combines data-driven network simplification with mixed-integer optimization for hierarchical cross-docking systems to reach the exact solution. Most existing models assume static demand, overlook the coordination between multiple cross-dock layers, or fail to incorporate service-level differentiation and strict delivery windows.

To summarize, all studies referenced in the preceding literature review are systematically compiled in Table1, arranged from 2017 to 2025. The table provides a structured comparison of each work based on key analytical dimensions, including Focus Area, Methodology, Key Contribution, Customer Satisfaction Metric, Time Window, Machine Learning Usage, and Limitations. This comparative framework highlights the progression of research in the field and contextualizes the evolution of approaches over time. The final row of the table represents this current study, positioning it within the broader landscape of existing literature.

**Table1**: Literature Review of Related Research (2017–2025)

| References | Focus Area | Methodology | Key Contributions | Customer Satisfaction Metric | Time Window | ML | Limitations |
|---|---|---|---|---|---|---|---|
| Necula et al. (2017) | Adaptive VRPTW routing | Ant Colony Optimization | Early heuristic method for adaptive routing | Indirectly | ✓ | - | Limited to metaheuristic optimization |
| Baniamerian et al. (2018) | Cross-docking scheduling | Two-phase Genetic Algorithm | Enhanced VRPCD scheduling under time constraints | Efficiency | ✓ | - | Static demand assumption |
| Küçükoğlu and Öztürk (2019) | Hybrid optimization | Genetic Algorithm & Simulated Annealing | More robust solution framework for VRPTW | Reliability | ✓ | - | Computational cost |
| Qian et al. (2020) | Logistics coordination | Systems analysis | Need for real-time info systems in logistics | Coordination | ✓ | - | Lack of data harmonization |
| Sultana et al. (2021) | Dynamic VRPTW | Deep Reinforcement Learning (RL) | DRL adaptation for real-time routing | Route adaptability | ✓ | ✓ | Limited generalization |
| Zhou et al. (2021) | Real-time delivery improvement | Deep Reinforcement Learning (RL) | Boosted on-time delivery rates by up to 22% | On-time delivery | ✓ | ✓ | Data-intensive requirements |
| Tirkolaee et al. (2021) | Sustainability in VRPTW | ML + Heuristics | Lowered environmental impact and improved routing | Emissions & cost | ✓ | ✓ | Integration difficulty |
| Filali and Filali (2021) | ML in supply chain optimization | Classification & Regression Models | Data-driven insights for forecasting and risk reduction | Service reliability | ✓ | ✓ | Feature selection limitations |
| Gunawan et al. (2022) | Reverse logistics in cross-docking | Two-phase metaheuristic | Solved multi-period VRPCD with returns | Return flow performance | ✓ | - | Static operational assumptions |

**Table1**: Literature Review of Related Research (2017–2025)

| References | Focus Area | Methodology | Key Contributions | Customer Satisfaction Metric | Time Window | ML | Limitations |
|---|---|---|---|---|---|---|---|
| Wölck and Meisel (2022) | Soft time window modeling | Branch-and-Price | Soft constraint optimization for delivery flexibility | Service flexibility | ✓ | - | Computational cost |
| Ghasemi et al. (2022) | Predictive vehicle scheduling | Graph Neural Networks | Forecast-based VRPTW optimization | Forecast precision | ✓ | ✓ | Training and data complexity |
| Kamble et al. (2023) | Warehouse analytics | Machine Learning | Improved supply chain decisions using AI | Forecasting Accuracy | ✓ | ✓ | Data reliability |
| Lima et al. (2023) | Inventory & Routing optimization | ML (Various) | Cost reduction and improved customer response | Cost & Service | ✓ | ✓ | Integration with legacy systems |
| Huang et al. (2023) | Replenishment under time constraints | PSO & Retrospective Approximation | Improved accuracy in time-windowed inventory planning | On-time delivery | ✓ | - | Demand variability impacts model robustness |
| Iklassov et al. (2024) | Stochastic VRPTW | Deep Reinforcement Learning | Real-time uncertainty in vehicle routing | Timeliness | ✓ | ✓ | High training cost and data complexity |
| Nezianya et al. (2024) | ML scalability in logistics | Empirical analysis of ML barriers | Highlighted barriers to ML scalability in logistics | System performance | ✓ | ✓ | Integration, data dependency |
| Zhang et al. (2025) | Demand forecasting | Predictive ML & Optimization | personalization at scale for delivery scheduling | Personalized delivery precision | ✓ | ✓ | Cost and privacy concerns |
| Holliday et al. (2025) | Quantum optimization | Hybrid Quantum Annealing | Real-time VRPTW optimization via quantum solutions | Constraint Optimization | ✓ | - | Requires quantum infrastructure |

**Table1**: Literature Review of Related Research (2017–2025)

| References | Focus Area | Methodology | Key Contributions | Customer Satisfaction Metric | Time Window | ML | Limitations |
|---|---|---|---|---|---|---|---|
| Osaba et al. (2024) | Pickup & delivery with time windows | Quantum-inspired heuristic | Extended quantum solutions to complex routing | Route quality & responsiveness | ✓ | ✓ | Experimental phase, limited adoption |
| **This paper (2025)** | **Online retailer business Expansion** | **ML and MIP** | **Real-time VRPCDTW optimization via ML & MIP** | **Customer predefined Time Window** | ✓ | ✓ | **Up to 5 Clusters for exact solutions** |

This paper fills the gaps by introducing a comprehensive framework: first, employing unsupervised clustering to simplify the network structure; and second, applying an optimization model that explicitly considers operational constraints such as synchronized flows between local and central cross-docks without split deliveries, and varying customer service expectations. By doing so, it not only enhances cost efficiency and service reliability but also provides a practical and computationally tractable solution for real-time logistics planning in modern e-commerce environments.

## 3. Problem definition

This research addresses the challenges of daily scheduling encountered by a large-scale online retailer, where customer orders, comprising multiple components sourced from different geographically dispersed suppliers, must be collected and consolidated for delivery using a homogeneous fleet of vehicles. The supply chain under consideration involves a centrally located main cross-dock, where incoming shipments are sorted, combined, and repackaged according to customer specified requirements, such as predefined delivery time frames, permissible delays, and varying service level expectations. Given the spatial scale and temporal sensitivity inherent in real-time e-commerce logistics, scheduling of vehicles and capacity allocation are critical to minimizing operational costs while ensuring a single delivery of each customer all order items in their predefined time frame to meet customer satisfaction. To address the computational intractability and coordination challenges of managing a large number of nodes and flows, this research introduces a hierarchical cross-docking structure in which suppliers and customers are clustered into regional groups. Each cluster is assigned a representative central node that functions as a local cross-dock, facilitating the intra-cluster consolidation of goods and enabling efficient inter-cluster coordination through the central cross-dock. This hierarchical structure reduces the dimensionality of the network, simplifies vehicle routing, ensures synchronized order flows from suppliers to customers, and minimizing direct point-to-point shipping across the entire system, which is illustrated in Figure1.

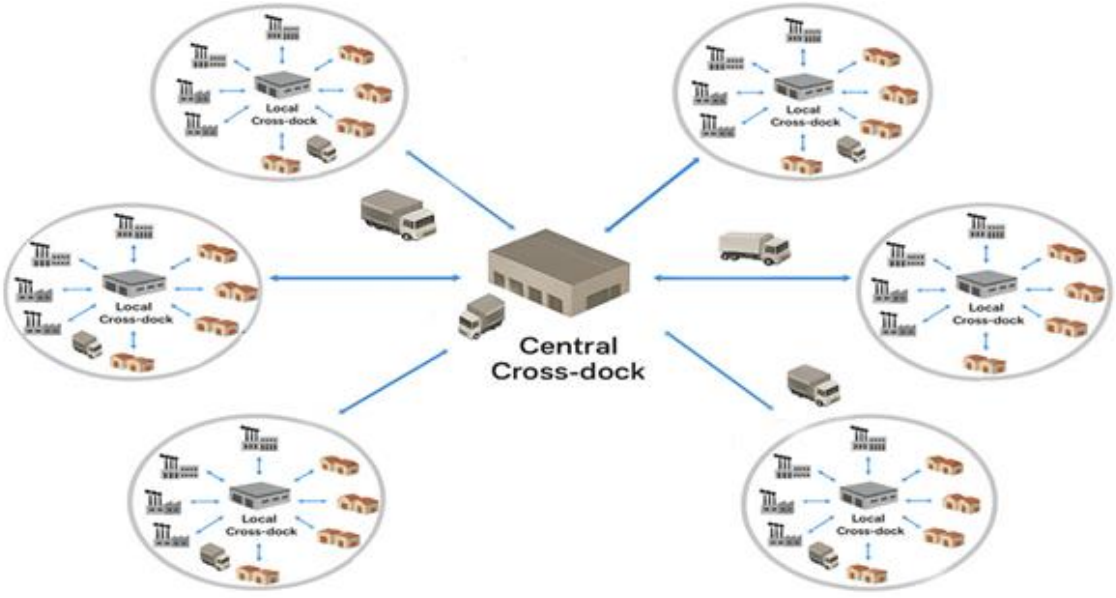


**Figure1:** Clusters in VRPCD logistics network

The proposed framework is structured into two integrated phases, within the first phase a comprehensive dataset of an online-retailer containing the geographical coordinates (latitude and longitude) of all suppliers and customers is used to calculate the pairwise distance matrix.

To reduce the computational burden of solving large-scale routing problems, an unsupervised machine learning technique is employed to group nearby suppliers and customers into discrete clusters. The cluster centers serve as local cross-docking facilities, which operate as bidirectional shipment cross-docks. Specifically, each local cross-dock receives inbound shipments from suppliers within its cluster and the central cross-dock, and also dispatches outbound shipments to customers within the same cluster and to the central cross-dock as needed. This simplified network supports a Mixed Integer Programming (MIP) model, the aim is to balance two central but conflicting objectives in e-commerce logistics: cost minimization and customer satisfaction. through timely, coordinated delivery expectations across various service categories, offering a scalable and exact approach for real-time logistics optimization in modern e-commerce systems. Ultimately, the model simultaneously optimizes, vehicle routing and assignment for both intra-cluster and inter-hub transportation, scheduling of shipments across local and central cross-docks, the allocation of logistics resources under service level constraints. To ensure model tractability and align with real-world constraints in large-scale e-commerce fulfillment, the following assumptions are made:

- Every order must be served in full by one vehicle during a single visit and Split delivery is not allowed.
- All vehicles in the fleet are homogeneous, having identical capacity and operational specifications.
- Customers define specific delivery time windows according to their selected service level, such as standard or express, which must be respected.
- Local and central cross-dock locations are fixed in advance, based on clustering outcomes such as those derived from the K-Means algorithm.
- Customer demand and supplier availability are considered deterministic and known at the time of planning.
- Local cross-docks are designed to support bi-directional flow, enabling them to both receive goods from suppliers and dispatch consolidated shipments to customers or the central hub.
- The planning is confined to a single operational day, meaning all pickups, cross-docking, and deliveries occur within the same daily timeframe.

By integrating ML techniques for spatial clustering and a MIP model for tactical planning and scheduling, this framework generates optimized solutions for key operational decisions in the hierarchical cross-docking system:

- Efficient utilization of real-time spatial data to maintain scalability and computational feasibility in large-scale systems while adhering to all operational constraints.
- Optimal identification of local cross-dock locations (cluster centers) to minimize intra-cluster and total transportation inefficiencies and enhance coordination.
- Vehicle routing and assignment for both local (within-cluster) and inter-hub (between local and central cross-docks) operations.
- Scheduling of inbound and outbound shipments relative to central and local cross-dock operations to ensure timely and synchronized deliveries.
- Balancing logistics cost and customer service quality, especially under differentiated service-level expectations.

### 3-1- Mathematical modeling

Based on the problem definition and the aforementioned assumptions, the proposed model is expressed in the following mathematical formulation, preceded by the specification of sets, indices, parameters, and decision variables:

| | |
|---|---|
| $I = \{1,2, \dots, i, \dots, i'\}$ | Set of all nodes |
| $O = \{o_1, o_2, o_3, o_4\} \subset I$ | Set of a cross-dock nodes |
| $S \subset I$ | Set of suppliers |
| $D \subset I$ | Set of customers |
| $v \subset \mathrm{V}$ | Set of all Vehicles |
| $i, i', j$ | Index of nodes |
| $v^s$ | Index of pickup vehicles |
| $v^d$ | Index of delivery vehicles |
| $m$ | Cross-dock inbound doors |
| $n$ | Cross-dock outbound doors |

Each pickup vehicle $V^s$ has a corresponding delivery vehicle $V^{i+n^v}$, where $n^v$denotes the pickup fleet size. For example, if $v^1, v^2, v^3$ are pickup vehicles, then their corresponding delivery vehicles are $v^4, v^5, v^6$ respectively.

**Parameters**

| | |
|---|---|
| $[a_i, b_i]$ | Time window of node $i$ |
| $[af_i, bf_i]$ | Customer preferred time window of node $i$ |
| $Q_i$ | Number of nodes $i$ order unit |
| $Cap_v$ | Capacity of vehicle $v$ |
| $n$ | Number of customers |
| $\mathrm{n}^v$ | Pickup fleet size |
| $AT_v$ | Fixed time of vehicles preparation at the cross-dock (unloading/reloading) |
| $BT_v$ | Variable time of vehicles preparation at the cross-dock (unloading/reloading) per order unit |
| $GT_v$ | Vehicle setup time at dock doors |
| $F$ | Penalty cost per time of deviation from the customers preferred time windows |
| $TT_{m,n}$ | Transportation time between dock doors $m$ and $n$ |
| $L_{i,j}$ | Distance between node $i$ and $j$ |
| $C_{i,j}$ | Transportation cost between node $i$ and $j$ |
| $T_{i,j}$ | Travel time between node $i$ and $j$ |
| $SL$ | Service level |
| $BM$ | An arbitrarily large constant |

**Binary variables**

| | |
|---|---|
| $X_{v,i,j}$ | 1, if vehicle $v$ travels from node $i$ to node $j$; 0, otherwise ($i, j \subset I$) |
| $as_{v,i}$ | 1, if vehicle $v$ is assigned to node $i$; 0, otherwise ($i \subset I$) |
| $u_{v,i}$ | 1, if vehicle $v$ unloads order $i$ at the cross-dock; 0, otherwise ($i \in S$, $v \in V$) |

$r_{v,i}$ 1, if vehicle $v$ reloads order $i$ at the cross-dock; 0, otherwise ($i \in S$, $v \in V$)

$g_v$ 1, if vehicle $v$ is unloaded at the cross-dock; 0, otherwise ($v \in V$)

$h_v$ 1, if vehicle $v$ is reloaded at the cross-dock; 0, otherwise ($v \in V$)

$x_{v,m}$ 1, if pickup vehicle $v$ is assigned to cross-dock door $m$; 0, otherwise ($v \in V^s$)

$y_{v,n}$ 1, if delivery vehicle $v'$ is assigned to cross-dock door $n$; 0, otherwise ($v' \in V^d$)

$p_{v,v'}$ 1, if pickup vehicles $v$ before $v'$are assigned to a door of cross-dock; 0, otherwise ($v, v' \in V^s$)

$q_{v,v'}$ 1, if delivery vehicles $v$ before $v'$are assigned to a door of cross-dock; 0, otherwise ($v, v' \in V^d$)

$w_{v,v'm,n}$ 1, if pickup vehicle $v$ is assigned to cross-dock door $m$ and delivery vehicle $v'$is assigned to dock door $n$; 0, otherwise ($v \in V^s, v' \in V^d, m \in M, n \in N$)

$ur_{v,v'}$ 1, if an order transfer from pickup vehicle $v$ to delivery vehicle $v'$ at the cross-dock; 0, otherwise ($v \in V^s, v' \in V^d$)

$sr_{i,v,v'}$ 1, if order $i$ transfer from pickup vehicle $v$ to delivery vehicle $v'$ at the cross-dock; 0, otherwise ($v \in V^s, v' \in V^d$)

**Continuous variables**

$DT_{v,i}$ Departure time of vehicle $v$ from node $i$ ($i \in I$, $v \in V$)

$UT_v$ Finish time of the unloading operation of vehicle $v$ at the cross-dock ($v \in V$)

$LT_v$ Start time of the reloading operation of vehicle $v$ at the cross-dock ($v \in V$)

$ST_i$ Finish time of the unloading order $i$ at the cross-dock ($v \in V$)

$E_i$ Positive deviation from customer preferred time window for node $i$

$P_i$ Negative deviation from customer preferred time window for node $i$

In accordance with the defined sets, indices, parameters, and variables the MIP model of this study is formulated as follows:

$$MinZ_1 = \sum_{i \in I} \sum_{j \in I} \sum_{v \in V} L_{i,j}.C_{i,j}.X_{v,i,j} + \sum_{i \in K} F.(E_i + P_i) \quad (1)$$

$$\sum_{m \in M} x_{v,m} = 1 \qquad \forall v \in V^s \quad (2)$$

$$\sum_{n \in N} y_{v,n} = 1 \qquad \forall v \in V^d \quad (3)$$

$$w_{v,v'm,n} \geq x_{v,m} + y_{v',n} + ur_{v,v'} - 2 \qquad \forall v \in V^s, v' \in V^d, m \in M, n \in N \quad (4)$$

$$x_{v,m} + x_{v',m} \leq p_{v,v'} + p_{v',v} + 1 \qquad \forall v, v' \in V^s, v \neq v', \forall m \in M \quad (5)$$

$$y_{v,n} + y_{v',n} \leq q_{v,v'} + q_{v',v} + 1 \qquad \forall v, v' \in V^d, v \neq v', \forall n \in N \quad (6)$$

$$UT_{v'} \geq UT_v + GT_{v'} - BM\left(1 - p_{v,v'}\right) \qquad \forall v, v' \in V^s, v \neq v' \qquad (7)$$

$$LT_{v'} \geq LT_v + GT_{v'} - BM\left(1 - q_{v,v'}\right) \qquad \forall v, v' \in V^d, v \neq v' \qquad (8)$$

$$\sum_{j \in I} \sum_{v \in V^s} X_{v,i,j} = 1 \qquad \forall i \in S \qquad (9)$$

$$\sum_{j \in I} X_{v,i,j} = as_{v,i} \qquad \forall i \in I, \forall v \in V \qquad (10)$$

$$\sum_{j \in I} \sum_{v \in V^d} X_{v,i,j} = 1 \qquad \forall i \in D \qquad (11)$$

$$\sum_{j \in I} X_{v,i,j} = as_{v,i} \qquad \forall i \in D, \forall v \in V \qquad (12)$$

$$\sum_{v \in V} as_{v,i} = 1 \qquad \forall i \in S \cup D \qquad (13)$$

$$\sum_{i \in S} \sum_{j \in I} Q_i X_{v,i,j} \leq Cap \qquad \forall v \in V^s \qquad (14)$$

$$\sum_{i \in D} \sum_{j \in I} Q_i X_{v,i,j} \leq Cap \qquad \forall v \in V^d \qquad (15)$$

$$\sum_{j \in S} X_{v,o_1,j} = 1 \qquad \forall v \in V^s \qquad (16)$$

$$\sum_{j \in D} X_{v,o_3,j} = 1 \qquad \forall v \in V^d \qquad (17)$$

$$\sum_{i \in I} X_{v,i,j} - \sum_{i \in I} X_{v,j,i'} = 0 \qquad \forall j \in S \cup D, v \in V \qquad (18)$$

$$\sum_{i \in S} X_{v,i,o_2} = 1 \qquad \forall v \in V^s \qquad (19)$$

$$\sum_{i \in D} X_{v,i,o_4} = 1 \qquad \forall v \in V^d \qquad (20)$$

$$DT_{v,j} \geq DT_{v,i} + T_{i,j}\left(L_{i,j}\right) + AT_v + BT_v\left(\sum_{i \in S} Q_i\right) - BM\left(1 - X_{v,i,j}\right) \qquad \forall i, j \in I, \forall v \in V \qquad (21)$$

$$a_i . as_{v,i} \leq DT_{v,i} \leq b_i . as_{v,i} \qquad \forall i \in I, \forall v \in V \qquad (22)$$

$$\sum_{v \in V^d} DT_{v,i} - bf_i \leq \mathrm{E}_i \qquad \forall i \in D \qquad (23)$$

$$af_i - \sum_{v \in V^d} DT_{v,i} \leq \mathrm{p}_i \qquad \forall i \in D \qquad (24)$$

$$u_{v,i} - r_{v+n^v,i} = \sum_{j \in S \cup o_2} X_{v,i,j} - \sum_{j \in D \cup o_4} X_{v+n^v,i+n,j} \qquad \forall i \in S, \forall v \in V^s \qquad (25)$$

$$u_{v,i} + r_{v+n^v,i} \leq 1 \qquad \forall i \in S, \forall v \in V^s \qquad (26)$$

$$\frac{1}{BM} \sum_{i \in S} u_{v,i} \leq g_v \leq \sum_{i \in S} u_{v,i} \qquad \forall v \in V \qquad (27)$$

$$UT_v = DT_{v,o_2} + GT_v + AT_v . g_v + BT_v . \sum_{i \in S} Q_i u_{v,i} \qquad \forall v \in V^s \qquad (28)$$

$$LT_v \geq UT_{v-n^v} \qquad \forall v \in V^d \qquad (29)$$

$$LT_{v'} \geq UT_v + TT_{m,n} - BM(1 - w_{v,v',m,n}) \qquad \forall v \in V^s, v' \in V^d, m \in M, n \in N \qquad (30)$$

$$LT_{v+n^v} \geq ST_i - BM(1 - u_{v,i}) \qquad \forall i \in S, \forall v \in V \qquad (31)$$

$$ST_i \geq UT_v - BM(1 - u_{v,i}) \qquad \forall i \in S, \forall v \in V \qquad (32)$$

$$\frac{1}{BM} \sum_{i \in S} r_{v,i} \leq h_v \leq \sum_{i \in S} r_{v,i} \qquad \forall v \in V \qquad (33)$$

$$DT_{v,o_3} = LT_v + GT_v + AT_v . h_v + BT_v . \sum_{i \in S} Q_i r_{v,i} \qquad \forall v \in V^d \qquad (34)$$

$$sr_{v,v',i} \leq u_{v,i} \qquad \forall i \in S, v \in V^s, v' \in V^d \qquad (35)$$

$$sr_{v,v',i} \leq r_{v',i} \qquad \forall i \in S, v \in V^s, v' \in V^d \qquad (36)$$

$$sr_{v,v',i} \geq u_{v,i} + r_{v',i} - 1 \qquad \forall i \in S, v \in V^s, v' \in V^d \qquad (37)$$

$$ur_{v,v'} \geq sr_{v,v',i} \qquad \forall i \in S, v \in V^s, v' \in V^d \qquad (38)$$

$$\frac{\sum_{i \in I} \sum_{j \in I} \sum_{v \in V} Q_i X_{v,i,j}}{\sum_{i \in I} Q_i} \geq SL \qquad (39)$$

$$X_{v,ij}, as_{v,i}, u_{v,i}, r_{v,i}, g_v, h_v \in \{0,1\} \qquad \forall i,j \in I, v \in V \qquad (40)$$

$$x_{v,m}, y_{v,n}, p_{v,v'}, q_{v,v'}, w_{v,v',m,n}, ur_{v,v'}, sr_{v,v',i} \in \{0,1\} \qquad \forall v,v' \in V, m \in M, n \in N, i \in I \qquad (41)$$

$$DT_{v,i}, UT_{v,i}, LT_v, ST_i, D_v^p, D_v^k, \mathrm{E}_i, \mathrm{P}_i \geq 0 \qquad \forall i \in I, v \in V \qquad (42)$$

Equation (1) The objective function is designed to simultaneously minimize total transportation costs and penalties incurred from violating customers' preferred time windows. Constraints (2) and (3) ensure that each pickup or delivery vehicle is respectively assigned to exactly one inbound or outbound door. Constraint (4) governs the relationship between the binary variables; specifically, ensures that any order unloaded, does not remain in the cross-dock. Constraint (5) requires that if two pickup trucks are assigned to the same inbound dock door, one must complete unloading before the other begins. Constraint (6) stipulates that if two delivery trucks share an outbound dock door assignment, one must finish loading before the other starts. Constraint (7) ensures that if two pickup vehicles are assigned to the same inbound dock door, the unloading of the second vehicle can only start after the first vehicle has finished and the required vehicle replacement time at the dock has passed. Constraint (8) Similarly, this constraint ensures that if two delivery vehicles are assigned to the same outbound dock door, the second vehicle cannot begin loading until the first vehicle has finished and the dock door's required replacement time has elapsed. Constraint (9) guarantees that every supplier must be visited by a pickup vehicle. Constraint (10) ensures if a vehicle assign to a supplier node, it must continue its route to another node. Constraint (11) ensures that every order must be sent to its customer node by a delivery vehicle. Constraint (12) ensures if a vehicle assign to a customer node, it must continue its route to another node. Constraint (13) guarantees that each supplier or customer is assigned to exactly one vehicle, preventing multiple vehicles from visiting the same node. Constraint (14) ensures that a delivery vehicle's load does not exceed its capacity. Constraint (15) ensures that a pickup vehicle's load does not exceed its capacity. Constraint (16) fixes the starting point of every pickup vehicle is at node $o_1$. Constraint (17) fixes the starting point of every delivery vehicle is at node $o_3$. Constraint (18) enforces route continuity; there is no return to any visited supplier or customer node. Constraint (19) ensures that every pickup vehicle finishes its route at node $o_2$ . Constraint (20) ensures that every delivery vehicle finishes its route at node $o_4$ . Constraint (21) calculates each vehicle's departure time from any visited node. Constraint (22) guarantees that all nodes are visited within their specified time frame. Constraints (23) and (24) compute the deviation from customers preferred time windows, allowing for early or late delivery penalties. Constraint (25) Indicates that if an order is collected by vehicle $V^s$and delivered by another vehicle $V^d$, then unloading must occur at the cross-dock. Constraint (26) ensures that if orders of supplier $i$ are unloaded from vehicle $v$ at the cross-dock, they do not load on the same vehicle to deliver. Constraint (27) determines whether a specific pickup vehicle unloads at the cross-dock. Constraint (28) calculates the unloading completion time for each pickup vehicle at the cross-dock, by summing the vehicle's arrival time, vehicle setup time at a dock door, a fixed unloading duration, and the additional time required for all orders being unloaded. Constraint (29) ensures that each delivery vehicle's loading process at the dock begins only after the corresponding pickup vehicle has completed unloading. Constraint (30) states that if orders are transferred from pickup vehicle $v$ to delivery vehicle $v'$, start of loading $v'$equals to sum of transfer time between dock doors and the unloading completion time of $v$. Constraint (31)

guarantees that delivery vehicles begin loading only after all necessary unloading operations have finished. Constraint (32) calculates the completion time of supplier $i$'s unloading operation. Constraints (33) and (34) mirror constraints 27 and 28 for the delivery operations (loading). Constraints (35) to (37) indicate that if unloading and loading between vehicles $v$ and $v'$are performed at the cross-dock, then binary variable $sr_{v,v',i}$ equals 1. Constraint (38) ensures that if a customer's order is transferred from pickup vehicle $v$ to delivery vehicle $v'$, then binary variable $ur_{v,v'}$ must be 1. Constraint (39) enforces the system's service level by ensuring that a specified percentage of deliveries is met. Constraints (40) to (42) define the decision variable types (binary, integer, continuous) and their permissible ranges.

## 4. Solution approach

The problem addressed in this study involves multiple interdependent decisions, including vehicle routing, shipment scheduling, and adherence to predefined time windows, all within a hierarchical cross-docking system. In large-scale, real-time e-commerce environments, logistical complexity is driven by the geographic dispersion of suppliers and customers, fluctuating daily demand, strict delivery constraints, and varying service-level agreements. This complexity is further compounded by operational features such as non-split deliveries and the synchronization required between local and central cross-docks. The combinatorial nature of routing and scheduling under these constraints significantly enlarges the solution space. Although unsupervised ML techniques, such as clustering, are applied to reduce structural complexity, the underlying MIP model still involves a high-dimensional decision space. These factors, temporal coupling, spatial dispersion, and real-time responsiveness, render the problem computationally intractable, classifying it as NP-hard, where commercial optimization solvers often fail to reach optimality within a feasible runtime as the instance size scales.

To simulate realistic logistics operations, this study employs real historical data from a publicly available online retail dataset hosted on Kaggle (Kaggle Online Retail Dataset). Using ML-based preprocessing, the latitude and longitude of all supply and demand nodes are transformed into a detailed distance matrix representing the pairwise spatial relationships within the network. This transformation provides the foundation for clustering and routing decisions, helping to structure the logistics network in a way that is both computationally tractable and spatially meaningful. The data's structure, along with its volume and granularity, make it highly relevant for modeling VRPCDTW, especially under real-time operational and service-level constraints typical of modern e-commerce.

To address the complexity of large-scale e-commerce logistics, the proposed solution framework first simplifies the problem structure using K-Means clustering. By grouping spatially proximate supply and demand nodes into localized clusters, the dimensionality of the network is reduced, enabling more effective coordination within regional cross-dock areas. This structured representation of the problem allows for the application of a MIP model that optimizes routing and scheduling under stringent capacity and time window constraints. The integration of clustering and exact optimization enables scalable, high-precision decision-making suitable for real-time operations, while also improving customer satisfaction and overall system efficiency. The following section outlines the clustering methodology used to form these localized logistics zones.

### 4-1- Network simplification via machine learning

To address the computational challenges associated with managing a dataset of 360 geographically dispersed suppliers and customers, this study employs unsupervised ML, specifically the K-Means clustering algorithm, as a dimensionality reduction and network

simplification technique. Clustering spatially proximate nodes reduces the complexity of the logistics network while preserving critical spatial relationships, thereby enabling more tractable optimization of cross-docking, routing, and scheduling decisions.

The K-Means algorithm divides the dataset into K clusters by iteratively assigning each data point to the cluster with the closest centroid, calculated based on Euclidean distances in the two-dimensional geographical coordinate space (latitude and longitude). After each assignment, centroids are recalculated as the arithmetic mean of the cluster members' coordinates, and the process is repeated until convergence, defined by negligible centroid movement or a maximum iteration limit. This approach effectively groups nodes into compact, non-overlapping clusters, minimizing within-cluster variance while maximizing inter-cluster separation.

A critical step in this process is determining the optimal number of clusters K, as it directly impacts intra-cluster transportation efficiency and cross-dock coordination. To identify the most appropriate K, the silhouette coefficient is employed as a robust cluster validation metric. The silhouette score measures the degree to which each point is well matched to its assigned cluster while being sufficiently separated from neighboring clusters. Scores approaching 1 indicate strong cohesion, values near 0 denote boundary points, and negative scores reveal potential misclassifications.

K-Means clustering was executed for a range of candidate values (K=2 to K=20), and the average silhouette score across all nodes was computed for each case. The results, summarized in Figure2, indicate that the highest silhouette score which statistically validates the clustering quality, achieving a maximum of 0.629 at K=11, indicating strong and reliable cluster separation, suggesting that this configuration produces the most cohesive and well-separated clusters. Selecting (K=11) balances dimensionality reduction with spatial fidelity, ensuring the simplified network remains representative of the original data while significantly decreasing computational burden.

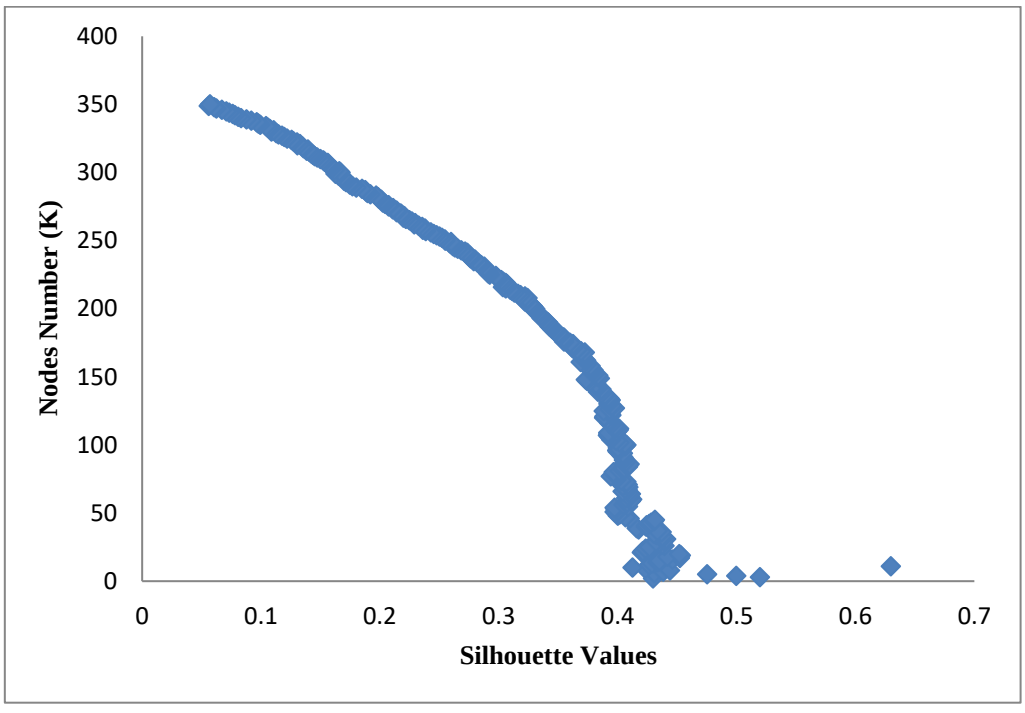

**Figure2:** Optimal number of K-Means clusters determined using the Silhouette method

In the finalized clustering structure, each cluster is anchored by a central node, designated as a local cross-dock for subsequent operations. This hierarchical organization of the cross-docking network supports efficient routing and scheduling, substantially reducing the computational requirements of solving large-scale VRPs. All computations were performed using Python on an Intel Core i7 2 GHz processor with 8 GB of RAM, demonstrating the framework's capability to support near real-time decision-making in dynamic e-commerce and logistics environments.

Figure3 presents the spatial distribution of the eleven cluster centroids derived from the 360 suppliers and customers, together with the central cross-dock hub (highlighted as an orange star). Annotated geographic coordinates indicate each centroid's position, which serves as the representative node for its cluster in the simplified model. The central hub, strategically positioned to minimize overall travel distances between clusters, anchors the cross-docking hierarchy and facilitates efficient routing and scheduling. This configuration maintains the spatial integrity of the original network while dramatically streamlining computational efforts for downstream optimization.

Building on the established clustering framework, the simplified logistics network now serves as the basis for subsequent optimization. By consolidating geographically proximate locations into representative centroids anchored by a central hub, the model achieves a balance between spatial fidelity and computational efficiency. This structure enables the development of advanced cross-docking, routing, and scheduling strategies, which are examined in the following analysis.

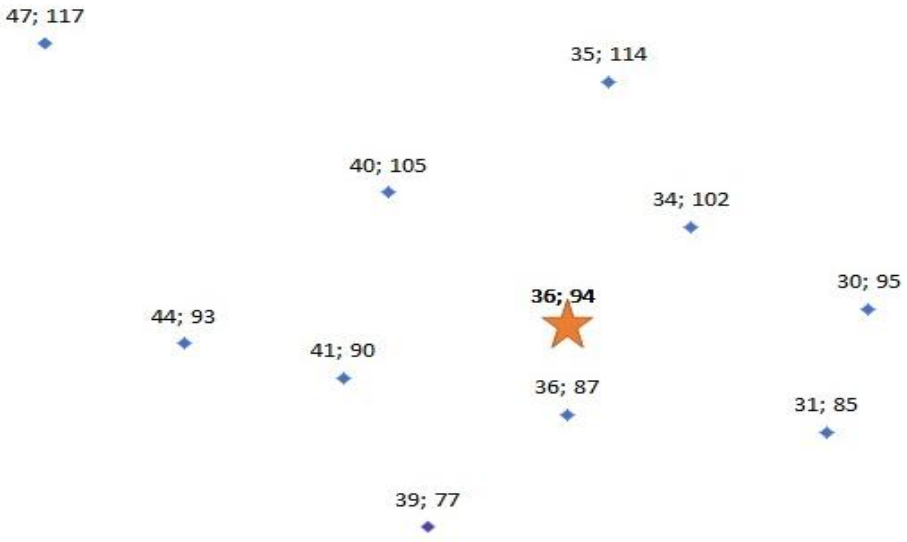


**Figure3:** Spatial distribution of cluster centroids (local cross-docks) and central cross-dock

Using the reduced and structured dataset generated through clustering, the second stage of the approach applies a MIP model to optimize vehicle routing and delivery scheduling under capacity and time window constraints. This model is implemented in GAMS version 24 and solved using the CPLEX solver, enabling efficient computation across clusters within short solution times. The MIP formulation ensures both accuracy and scalability, addressing the real-time operational demands of large-scale logistics systems. Following this, a sensitivity analysis

explores how adjusting key input parameters influences total cost, delivery performance, and ultimately, customer satisfaction.

## 5. Sensitive analysis and results

This section conducts a series of sensitivity analyses to validate the robustness of the proposed logistics optimization framework while identifying strategic parameter settings that support effective decision-making in dynamic online retail environments. The analysis examines how variations in vehicle capacity and fleet size influence cost efficiency and service quality under fluctuating demand patterns, thereby ensuring system adaptability and operational resilience. Furthermore, it evaluates the effect of modifying the number of delivery time frames offered to customers on both cost structures and scheduling flexibility, enabling the network to accommodate demand variability without degradation in performance. In the subsequent sub-sections, the first part investigates different vehicle capacity scenarios, the second part determines the optimal fleet size based on the most effective capacity obtained from the first analysis, and the third part assesses the impact of demand fluctuations on the selected optimal configuration in terms of both capacity and quantity.

### 5-1- Effects of different vehicle capacity

This analysis examines the effect of incremental 10% increases in vehicle capacity across varying fleet quantities on the total logistics costs of an online retailer. The evaluation distinctly explores how different combinations of fleet size and vehicle capacity influence overall costs, while also considering the role of scheduling flexibility, represented by configurations with six and eight daily time frames.

For feasibility, the capacity of a single vehicle operating alone must at least match the maximum number of orders; under this condition, both configurations yield identical costs about 307$. In the case of two-vehicle fleet, a minimum capacity of 165 packages (equivalent to 50% more than the baseline vehicle capacity of 110 packages) is required to be feasible and serve all orders. As shown in Figure4, at the first feasible capacity level, the eight-time-frame configuration results in lower costs than the six-time-frame configuration. However, both configurations exhibit a similar trend, costs decrease substantially when vehicle capacity increases to 60% above the baseline, after which further capacity growth yields negligible cost reductions. Beyond a 70% capacity increase, the eight-time-frame configuration consistently remains more expensive than the six-time-frame counterpart, despite following the same cost-stabilization pattern.

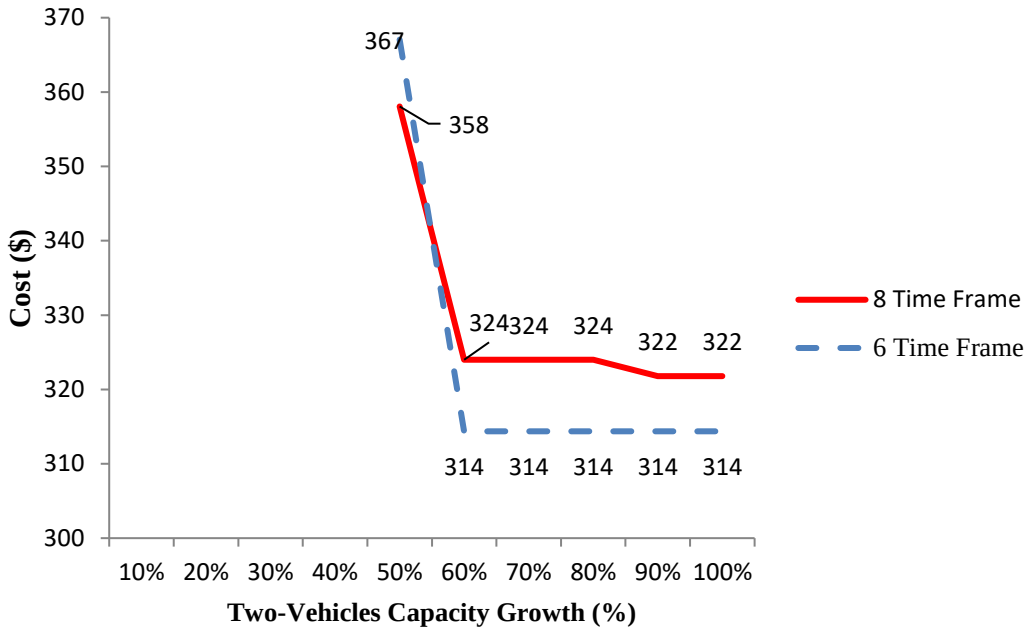


**Figure4:** Impact of two-vehicle capacity changes on total cost across different time-frame configurations

The three-vehicle case analysis of Figure5 begins at a 10% increase over the base vehicle capacity, where both the six and eight-time-frame configurations are feasible and total costs are closely aligned, with the eight-time-frame setting exhibiting a slight cost advantage. As capacity increases, both configurations follow a similar downward cost trajectory, with the steepest reductions occurring within the 30%–60% growth range. Beyond 60%, the rate of cost reduction slows markedly, indicating diminishing returns from additional capacity expansion. However, while the overall trend remains similar for both configurations, the cost gap between them widens progressively, with the six-time-frame option gaining a clearer and more consistent cost advantage as capacity grows. This shift suggests that although the eight-time-frame configuration may offer marginal efficiency at very low-capacity levels, higher capacities increasingly favor the six-time-frame arrangement in terms of cost performance.

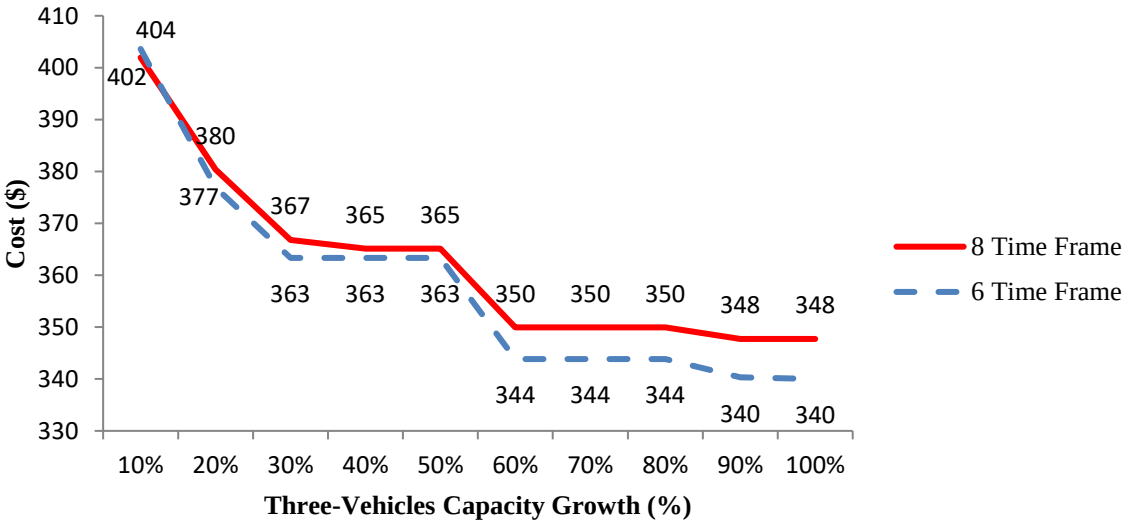

**Figure5:** Impact of three-vehicle capacity changes on total cost across different time-frame configurations

In the four-vehicle scenario illustrated in Figure6, the cost-reduction pattern remains, from 20% to 80% increase in capacity the total costs of both time-frame configurations are stable and same, suggesting improved load distribution efficiency with a larger fleet. In contrast, for the five-vehicle case, costs remain constant across all capacity growth levels, recorded at 480$ for six-time frames and 470$ for eight-time frames. This indicates full-service feasibility and no cost sensitivity to capacity in this fleet size, reflecting an over-provisioning effect where additional capacity does not yield operational savings. Collectively, these results underscore a nonlinear relationship between fleet size, vehicle capacity, and cost efficiency, with optimal cost responsiveness occurring in intermediate fleet sizes.

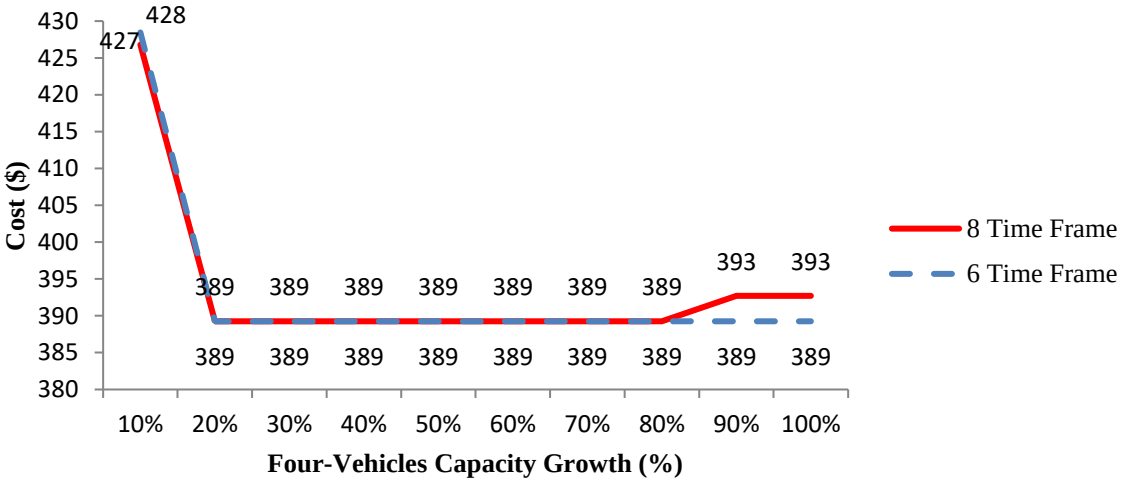


**Figure6:** Impact of four-vehicle capacity changes on total cost across different time-frame configurations

At the lowest fleet size, represented by a single vehicle, the system achieves the minimum operational cost; however, this configuration severely restricts flexibility. In the event of rising demand, the system may become infeasible, leading to missed orders, customer dissatisfaction, and potential reputational risks for the online retailer. Conversely, overestimating capacity under such a setting can impose unnecessary costs. From a managerial perspective, stability emerges when the fleet size reaches five or more vehicles, as costs remain constant across all capacity levels, even when scheduling is extended to eight-time frames per day. Considering the intermediate range, the analysis indicates that vehicle capacities between 60% and 90% added to the baseline vehicle capacity consistently yield the lowest and most stable costs across both six- and eight-time-frame settings. These findings highlight that the critical trade-off lies not only in capacity scaling but in aligning fleet size with scheduling flexibility. Accordingly, the following subsection examines in detail how different fleet sizes, combined with varying time-frame configurations, can balance cost effectiveness with adaptability to fluctuating demand in online retail logistics.

### 5-2- Effects of fleet size

The comparative evaluation of 60% and 90% added capacity in Figure7 demonstrates a consistent relationship between fleet size and total logistics cost. In both cases, costs increase in an almost same linear manner as the fleet size expands, while one vehicle in all scenarios is infeasible indicating insufficient fleet size to meet demand or a boundary case of the model, the five-vehicle configuration yielding the highest expenditures, though slightly less under the

eight-time-frame scheduling scheme at $470. This pattern highlights that excessive fleet expansion reduces system efficiency by 16% under six time frames and by 19% under eight time frames, and incurring substantial additional costs.

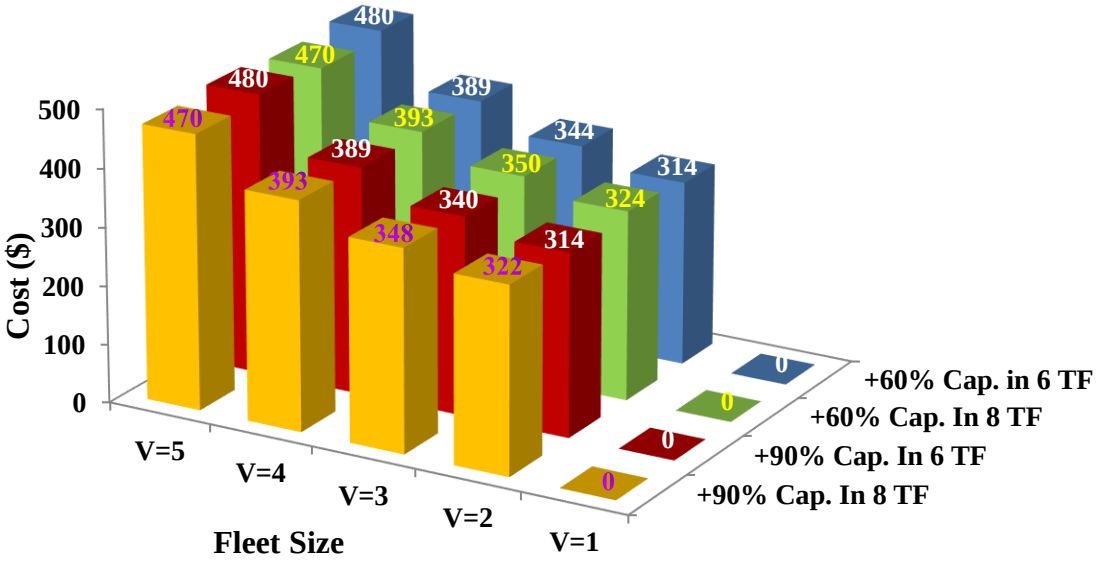


**Figure7:** Fleet size comparisons under 60% and 90% added capacity across 6 and 8 time frames

The evaluation of mid-range fleet sizes highlights important trade-offs between cost efficiency, flexibility, and system resilience, with the four-vehicle configuration consistently emerging as the most balanced solution. In this setup, costs increase only moderately (around 12%), while the cost gap between six- and eight-time frame schedules is the smallest across mid-range configurations. This indicates that managers can offer customers shorter and more frequent delivery windows without incurring significant additional expenses. Furthermore, the identical cost outcomes under 60% and 90% capacity expansion demonstrate the configuration's insensitivity to demand fluctuations, underscoring its role as a strategically resilient design point. This stability suggests that managers need not overinvest in fleet augmentation at this level; instead, resources may be more effectively allocated to enhancing service differentiation, particularly through flexible delivery options. Consequently, the four-vehicle configuration emerges as a structurally balanced solution that maintains efficiency, strengthens customer satisfaction, and secures long-term competitiveness in volatile online retail markets.

The two and three vehicle configurations present distinct trade-offs between cost efficiency and operational resilience. While the two-vehicle fleet achieves the lowest overall costs, particularly under 90% added capacity; The three-vehicle configuration, slightly more expensive, with cost increases of approximately 8% compared to the two-vehicle fleet, offers improved resilience, and when supported by higher capacity scaling (90% above baseline), it maintains more service reliability under most demand conditions. Interestingly, under 90% capacity expansion, the costs of two- and three-vehicle fleets are slightly lower, but the sensitivity to increasing the number of daily time frames (from six to eight) is higher for these smaller fleets, particularly under 60% capacity growth, where flexibility is noticeably reduced. Overall, these analyses consider only systemic operational costs, excluding vehicle purchase prices, highlighting that managerial decisions in online retail logistics must balance day-to-day

tactical and operational considerations with strategic investment choices, taking into account current vehicle market prices and long-term service reliability objectives.

Taken together, the analysis shows that while smaller fleets can minimize costs, they do so at the expense of flexibility and resilience, limiting the system's ability to adapt to demand volatility. Temporal flexibility, achieved by increasing scheduling windows from six to eight, consistently reduces costs across all fleet sizes, highlighting its critical role in enhancing efficiency and reliability. Among the configurations studied, the four-vehicle fleet emerges as the most robust option, balancing cost efficiency with flexibility, whereas larger fleets, such as five vehicles, prove generally inefficient. Building on these insights, the next subsection explores how demand fluctuations affect performance across different fleet sizes and capacity levels, providing a deeper understanding of how fleet sizing and capacity scaling interact with uncertainty in online retail logistics.

### 5-3- Effects of demand variation

Building upon these insights, the analysis now turns to the role of demand variability in shaping system performance under different fleet and capacity configurations. Specifically, it contrasts a four-vehicle fleet identified as the most balanced configuration in terms of cost and flexibility, with other fleet sizes under 90% added capacity, which offers broader scalability. This comparison provides a deeper understanding of how capacity scaling and fleet sizing interact with fluctuating order volumes, highlighting critical trade-offs between efficiency, adaptability, and resilience in online retail logistics.

Table2 illustrates the sensitivity of the system to demand fluctuations under mid-sized fleet and vehicle configuration with 90% added capacity of the baseline. The results show that reductions in demand do not produce any observable changes in total cost, as the available capacity remains underutilized yet sufficient to accommodate lower volumes. However, the system demonstrates limited resilience to upward fluctuations in demand. Specifically, with two vehicles it can only tolerate increases up to 20% beyond the baseline level, after which the problem becomes infeasible. This infeasibility reflects the system's inability to allocate sufficient capacity to cover all customer orders, rendering it unresponsive to higher-than-expected demand. This efficiency comes at the expense of adaptability, as the system can only maintain stable operations under average or below-average demand. Such constraints not only risk unserved orders but also jeopardize customer satisfaction and potentially undermine the reputation and competitiveness of the online retailer, introducing hidden long-term costs that extend beyond immediate logistics expenses.

**Table2**: Comparison of Fleet Size Costs under Capacity Expansion and Time-Frame Configurations across Demand Fluctuations

| | | **Demand Fluctuation (%)** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Fleet Size /Capacity** | **Time Frames** | **-20** | **-10** | **0** | **+10** | **+20** | **+30** | **+40** | **+50** | **+60** | **+70** | **+80** | **+90** | **+100** |
| **4V /+60% Cap.** | 6TF | 389 | 389 | 389 | 389 | 389 | 428 | 428 | 428 | 428 | Inf. | | | |
| | 8TF | 393 | 393 | 393 | 393 | 393 | 393 | 427 | 427 | 427 | Inf. | | | |
| **4V** | 6TF | 389 | 389 | 389 | 389 | 389 | 389 | 389 | 389 | 423 | 428 | 428 | 428 | Inf. |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **/+90% Cap.** | 8TF | 393 | 393 | 393 | 393 | 393 | 393 | 393 | 393 | 421 | 427 | 427 | 427 | Inf. |
| **3V /+90% Cap.** | 6TF | 340 | 340 | 344 | 344 | 344 | 363 | 363 | 363 | 399 | 404 | 404 | Inf. | |
| | 8TF | 348 | 348 | 350 | 350 | 365 | 365 | 365 | 367 | 396 | 402 | 402 | Inf. | |
| **2V /+90% Cap.** | 6TF | 314 | 314 | 314 | 321 | 321 | 371 | Inf. | | | | | | |
| | 8TF | 322 | 322 | 322 | 324 | 324 | 371 | Inf. | | | | | | |

The effects of demand fluctuations on system performance under a three-vehicle configuration with 90% added capacity reveal that the system remains cost-efficient and stable across a broad range of demand conditions. When demand decreases, logistics costs decline slightly, reflecting efficient operations under partial utilization. As demand rises, costs increase gradually, and the system remains feasible and responsive up to an 80% demand surge. Within this range, scheduling flexibility improves, as the cost differences between the six and eight time-frame delivery configurations become negligible. This convergence highlights enhanced resilience, where flexibility, responsiveness, and service reliability align to strengthen competitiveness in online retail operations. In contrast, the performance of a four-vehicle fleet with 90% added capacity demonstrates even greater resilience. While costs remain stable under moderate fluctuations, the system exhibits remarkable adaptability, absorbing demand increases of up to 90% without significant cost escalation. Importantly, the cost gap between the six and eight time-frame configurations is even smaller in this setting, underscoring that the marginal cost of increased scheduling flexibility is minimized. This capacity for extended responsiveness enables managers to maintain service quality and customer satisfaction even under highly volatile conditions, while still preserving cost control.

Taken together, these findings indicate that the optimal fleet strategies are the four-vehicle configuration with 90% added capacity and the three-vehicle configuration with 90% added capacity. While the three-vehicle option is approximately 12% less costly, it can only sustain responsiveness up to 80% demand growth. By comparison, the four-vehicle system supports demand surges up to 90%, while also reducing sensitivity to time-frame expansions. From a managerial perspective, this balance suggests that the four-vehicle configuration not only safeguards operational efficiency but also provides strategic room for initiatives such as promotional campaigns or free-shipping policies, reinforcing long-term competitiveness in dynamic e-commerce markets.

### 5-4- Comparative Interpretation of Results with Recent Studies

This subsection interprets the numerical findings of the present study in comparison with recent cross-docking and time-window–based optimization researches.

**Table 3:** comparative summary of key numerical results with recent studies

| **Study** | **Baniamerian (2020); Liu (2021); Khorasany (2023)** | **This paper (2025)** |
|---|---|---|
| **Sensitivity to Capacity Increase** | Moderate cost sensitivity (10–15% reductions) | Strong reduction in cost (22–28%) when capacity increases by 60–90%; clear saturation points after 70% |

| | | |
|---|---|---|
| **Sensitivity to Fleet/Resource Expansion** | Resource expansion often reduces lateness but increases operating cost; optimal point rarely detected | Cost stabilizes for ≥4 vehicles; 4-vehicle fleet shows optimal balance; over-expansion adds 16–19% cost |
| **Demand Resilience** | Feasible up to ~50% additional demand in most benchmarks | 3-vehicle feasible up to +80% demand; 4-vehicle feasible up to +90% |
| **Impact of Time-Window Flexibility** | Higher flexibility increases computational complexity and sometimes costs | Increasing delivery windows from 6 to 8 always reduces costs by 4–6% |

The comparative analysis reveals that the results of the proposed model outperform those reported in recent studies, including *Two-phase Genetic Algorithm for VRP with Cross-Docking and Time Windows* and the *Bi-objective Bi-level Cross-Dock Truck Scheduling* framework. While earlier works typically demonstrate moderate cost reductions (8–15%) when adjusting capacity or scheduling structures, the present study shows substantially stronger sensitivity-costs decline by 22–28% when vehicle capacity increases by 60–90%, and they converge to a stable region beyond the 70% expansion point. Additionally, unlike prior research, which often identifies continuous benefits from fleet growth without clear saturation, this study uncovers a distinct optimal fleet size: a four-vehicle configuration that balances efficiency, flexibility, and resilience while avoiding the 16–19% cost penalties associated with over-expansion. Moreover, the model supports unprecedented levels of demand variability, maintaining feasibility up to 80–90% surges, whereas comparable studies generally sustain only 40–50%. Finally, temporal flexibility, operating with eight delivery time frames, consistently lowers costs without compromising feasibility, demonstrating a stronger and more stable effect than previously reported. Collectively, these findings emphasize that the proposed model not only aligns with the best practices established in recent literature but also advances them by offering clearer optimality thresholds, higher resilience to uncertainty, and a more pronounced interplay between capacity, scheduling, and demand dynamics.

## 6. Conclusion and future researches

This study introduces an integrated optimization framework for real-time e-commerce logistics, emphasizing vehicle routing and scheduling within a cross-docking system subject to multiple constraints. The framework tackles key operational challenges, including synchronized supplier coordination, customer-specific time windows, and non-split deliveries within a large-scale, geographically dispersed network. The complexity of the logistics network was reduced by applying unsupervised machine learning, specifically K-Means clustering, to group suppliers and customers into regional clusters. Each cluster was assigned a central local cross-dock to manage both inbound and outbound flows. This clustering not only enabled efficient vehicle allocation but also simplified the distance matrix from raw geographic coordinates, making the problem computationally tractable. Dimensionality reduction techniques were employed by identifying the maximum silhouette score, evaluated to identify the optimal number of clusters, preserving routing efficiency while significantly reducing computational burden.

Subsequently, a mixed-integer programming (MIP) model was developed to determine the optimal vehicle routing and scheduling. This model incorporated hard time windows, homogeneous vehicle capacity constraints, and bi-directional flows between the main and local cross-docks. Orders from multiple suppliers were required to be consolidated and delivered to

each customer as a single shipment, without split deliveries, in accordance with the predefined service category and time window. The model successfully captured the flow of goods from dispersed suppliers to local cross-docks and onward to the central hub, and ultimately to customers, enabling optimal coordination of vehicle routes and terminal operations.

Sensitivity analyses conducted on key parameters, including vehicle capacity, fleet size, demand levels, and the number of time frames, revealed important managerial insights. The findings highlighted cost thresholds beyond which further investments in capacity or scheduling flexibility no longer improved efficiency or effectiveness. Notably, aligning vehicle capacity and fleet size with market conditions and leveraging time-frame flexibility led to substantial cost savings. By sustaining stable performance across a wider spectrum of demand fluctuations, the system enhances its adaptability to the inherent volatility of online retail markets, thereby ensuring both service quality and long-term competitiveness.

By integrating data-driven machine learning techniques with mathematical optimization, the proposed VRPCDTW framework improves cost efficiency and on-time delivery rates, while delivering a practical and scalable solution for real-world logistics scheduling in dynamic, high-volume e-commerce environments. In addition to operational gains, it supports strategic decision-making for logistics managers aiming to elevate service levels in competitive retail markets. The framework also addresses critical gaps in the literature by incorporating service-level differentiation, multi-layer cross-docking coordination, and real-time adaptability within complex e-commerce logistics systems.

Future research can extend this framework by incorporating stochastic demand patterns, real-time order fluctuations, and uncertain traffic conditions to better reflect the dynamic nature of e-commerce logistics. Additionally, further exploration of hybrid quantum computing or reinforcement learning-based solution methods may offer enhanced scalability and efficiency in computation for solving real-time, large-scale optimization problems.

**References**


Baniamerian, A., Bashiri, M., & Zabihi, F. (2018). Two phase genetic algorithm for vehicle routing and scheduling problem with cross-docking and time windows considering customer satisfaction. *Journal of Industrial Engineering International*, *14*, 15-30.

Bodnár, B., & Juhász, J. (2023). Examination of the Development Possibilities of the Cross-docking Strategy. *Advanced Logistic Systems-Theory and Practice*, *17*(1), 33-38.

Boysen, N., De Koster, R., & Weidinger, F. (2019). Warehousing in the e-commerce era: A survey. *European Journal of Operational Research*, *277*(2), 396-411.

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management science*, *6*(1), 80-91.

Filali, A. E., & Filali, S. E. (2021). Exploring applications of Machine Learning for supply chain management. 2021 Third International Conference on Transportation and Smart Technologies (TST),

Ghasemi, N., Safavi, A., Saremi, H. R., & Asgary, A. (2022). Assessing the impact of Internet of Things (IoT) on urban multi-modal mobility for optimal routing: A meta-review. *International Journal of Transportation Engineering*, *10*(1), 919-945.

Ghomi, V., Nooraei, S. V. R., Shekarian, N., Shokoohyar, S., & Parast, M. (2023). Improving supply chain resilience through investment in flexibility and innovation. *International Journal of Systems Science: Operations & Logistics*, *10*(1), 2221068.

Ghorbani, E., Alinaghian, M., Gharehpetian, G. B., Mohammadi, S., & Perboli, G. (2020). A survey on environmentally friendly vehicle routing problem and a proposal of its classification. *Sustainability*, *12*(21), 9079.

Gunawan, A., Widjaja, A. T., Vansteenwegen, P., & Yu, V. F. (2022). Two-phase Matheuristic for the vehicle routing problem with reverse cross-docking. *Annals of Mathematics and Artificial Intelligence*, *90*(7), 915-949.

Hasani-Goodarzi, A., & Tavakkoli-Moghaddam, R. (2012). Capacitated vehicle routing problem for multi-product cross-docking with split deliveries and pickups. *Procedia-Social and Behavioral Sciences*, *62*, 1360-1365.

Holliday, J. B., Blount, D., Osaba, E., & Luu, K. (2025). Advanced Quantum Annealing Approach to Vehicle Routing Problems with Time Windows. *arXiv preprint arXiv:2503.24285*.

Huang, Y.-D., Wu, S., & Yuan, X.-F. (2023). An Optimal Inventory Replenishment Strategy with Cross-docking System and Time Window Problem. International Conference on Genetic and Evolutionary Computing,

Iklassov, Z., Sobirov, I., Solozabal, R., & Takáč, M. (2024). Reinforcement Learning for Solving Stochastic Vehicle Routing Problem. Asian Conference on Machine Learning,

Kamble, S. S., Gunasekaran, A., Subramanian, N., Ghadge, A., Belhadi, A., & Venkatesh, M. (2023). Blockchain technology's impact on supply chain integration and sustainable supply chain performance: Evidence from the automotive industry. *Annals of Operations Research*, *327*(1), 575-600.

Küçükoğlu, İ., & Öztürk, N. (2019). A hybrid meta-heuristic algorithm for vehicle routing and packing problem with cross-docking. *Journal of Intelligent Manufacturing*, *30*, 2927-2943.

Lee, Y. H., Jung, J. W., & Lee, K. M. (2006). Vehicle routing scheduling for cross-docking in the supply chain. *Computers & industrial engineering*, *51*(2), 247-256.

Lima, C., Batista, M., Relvas, S., & Barbosa-Povoa, A. (2023). Optimizing the design and planning of a sugar-bioethanol supply chain under uncertain market conditions. *Industrial & Engineering Chemistry Research*, *62*(15), 6224-6240.

Necula, R., Breaban, M., & Raschip, M. (2017). Tackling dynamic vehicle routing problem with time windows by means of ant colony system. 2017 IEEE Congress on Evolutionary Computation (CEC),

Nozari, H. (2025). NeuroTwinceutics™ as a Neuromorphic Digital Twin Model for Predictive and Personalized Pharmacotherapy. Transformative Science, 1(1), 1-8.

Osaba, E., Villar-Rodriguez, E., & Asla, A. (2024). Solving a real-world package delivery routing problem using quantum annealers. *Scientific Reports*, *14*(1), 24791.

Parasuraman, A., Zeithaml, V. A., & Berry, L. L. (1988). Servqual: A multiple-item scale for measuring consumer perc. *Journal of retailing*, *64*(1), 12.

Pei, P. P.-E., Simchi-Levi, D., & Tunca, T. I. (2011). Sourcing flexibility, spot trading, and procurement contract structure. *Operations Research*, *59*(3), 578-601.

Poullet, J. (2020). *Leveraging machine learning to solve The vehicle Routing Problem with Time Windows* Massachusetts Institute of Technology].

Qian, C., Zhang, Y., Jiang, C., Pan, S., & Rong, Y. (2020). A real-time data-driven collaborative mechanism in fixed-position assembly systems for smart manufacturing. *Robotics and Computer-Integrated Manufacturing*, *61*, 101841.

Rushton, A., Rivett, D., Carlesso, L., Flynn, T., Hing, W., & Kerry, R. (2014). International framework for examination of the cervical region for potential of Cervical Arterial Dysfunction prior to Orthopaedic Manual Therapy intervention. *Manual therapy*, *19*(3), 222-228.

Santos, M. J., Amorim, P., Marques, A., Carvalho, A., & Póvoa, A. (2020). The vehicle routing problem with backhauls towards a sustainability perspective: A review. *Top*, *28*(2), 358-401.

Sippel, L., & Forbes, M. (2024). Enhancements of Fragment Based Algorithms for Vehicle Routing Problems. *arXiv preprint arXiv:2411.13151*.

Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, *35*(2), 254-265.

Sultana, N. N., Baniwal, V., Basumatary, A., Mittal, P., Ghosh, S., & Khadilkar, H. (2021). Fast approximate solutions using reinforcement learning for dynamic capacitated vehicle routing with time windows. *arXiv preprint arXiv:2102.12088*.

Taillard, É., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J.-Y. (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation science*, *31*(2), 170-186.

Tirkolaee, E. B., Sadeghi, S., Mooseloo, F. M., Vandchali, H. R., & Aeini, S. (2021). Application of machine learning in supply chain management: a comprehensive overview of the main areas. *Mathematical problems in engineering*, *2021*(1), 1476043.

Wölck, M., & Meisel, S. (2022). Branch-and-price approaches for real-time vehicle routing with picking, loading, and soft time windows. *INFORMS Journal on Computing*, *34*(4), 2192-2211.

Zhang, X., Li, W., Li, R., Fu, Z., Tang, T., Zhang, Z., Chen, W.-Y., Noorshams, N., Jasapara, N., & Ding, X. (2025). Personalized Interpolation: An Efficient Method to Tame Flexible Optimization Window Estimation. *arXiv preprint arXiv:2501.14103*.

Zhou, Y., Huang, J., Shi, J., Wang, R., & Huang, K. (2021). The electric vehicle routing problem with partial recharge and vehicle recycling. *Complex & Intelligent Systems*, *7*, 1445-1458.