Advances in Industrial Engineering, December 2026, 60(2): 447-469
DOI: 10.22059/aie.2026.412532.1977

RESEARCH PAPER

# A Learning-Based Co-Evolutionary Approach with Semi-Bandit Feedback for the Vehicle Routing Problem

**Masoumeh Messi Bidgoli[1]* and Sara Mohammadi Jozani[2]**

[1]*Assistant Professor, Industrial Engineering Group, Golpayegan College of Engineering, Isfahan University of Technology, Golpayegan, 87717-67498, Iran.*
[2]*Ph.D. Candidate, Department of Industrial Engineering, Faculty of Engineering, College of Farabi, University of Tehran, Qom, Iran.*

Received: 09 April 2026, Revised: 19 May 2026, Accepted: 10 June 2026
© The Author(s) retain the copyright 2026

## Abstract

He Network-Interdicted Vehicle Routing Problem (NIVRP) extends classical VRP settings by incorporating network interdiction, where selected arcs or nodes may be disrupted or rendered inaccessible. Consequently, routing decisions must account for potential interdictions. This setting induces a strategic conflict between Logistics Service Providers (LSPs) and an adversarial interdictor seeking to disrupt routes by targeting specific arcs. Typically, the interdictor lacks complete prior information regarding depot locations and customer demands, and acquiring such information is costly. Through repeated interactions, however, the interdictor can observe selected routes and seized cargo on interdicted arcs, gradually refining demand estimates and reducing uncertainty. Over time, this process enables increasingly accurate predictions of illicit cargo flows. The interaction is modeled as a repeated game, in which the interdictor engages in an online learning process to adapt to dynamically changing checkpoint configurations. This study analyzes this conflict within a repeated game framework using an online learning approach. A bi-level metaheuristic algorithm, leveraging semi-bandit feedback, is proposed. Computational experiments on randomly generated instances demonstrate that the proposed learning-based method significantly outperforms benchmark approaches and exhibits satisfactory convergence behavior.

**Keywords:**
Repeated Network Interdiction; Vehicle Routing Problem; Online Learning; Semi-Bandit Feedback; Bi-Level Genetic Algorithm.

## 1. Introduction

Vehicle routing problems (VRPs) under adversarial conditions have recently gained significant attention due to mounting security concerns and operational uncertainties within transportation networks. In real-world logistics, distributors may intentionally transport illicit, hazardous, or high-risk cargo that threatens public safety, or they may operate overloaded vehicles that degrade infrastructure. Government agencies, acting as interdictors, often respond by deploying monitoring technologies or enforcing route blockages to restrict the movement of such cargo. Conversely, legal and high-value shipments may be targeted by criminal actors seeking to disrupt or seize goods.

This study focuses on adversarial interactions where the interdictor assumes a socially beneficial role, while the logistics service provider (LSP) is modeled as the adversarial agent

*Corresponding author: (Masoumeh Messi Bidgoli)
Email: m.messi@iut.ac.ir

attempting to evade detection. Both decision-makers are assumed to be rational and pursue optimal strategies. NIVRPs capture these interactions by modeling how interdiction threats shape routing behavior and how limited resources can be deployed strategically. This model provides insights into selecting less vulnerable routes, identifying critical arcs, and allocating interdiction assets, such as sensors or inspection checkpoints, to maximize security impact.

A primary practical challenge is information asymmetry: interdictors rarely possess full knowledge of an LSP's depot locations, demand profiles, or routing behavior. Over repeated interactions, however, interdictors can gather intelligence by monitoring observed routes and identifying seized cargo, thereby gradually improving their estimates of hidden parameters. This motivates the learning-based adversarial setting explored in this work, where the interdictor adapts their strategy over multiple sequential rounds.

The present study extends the destructive NIVRP by incorporating a repeated-game learning mechanism in which the interdictor updates their estimates of customer demands based on observed interdiction outcomes. This research addresses several essential questions for interdiction authorities:

- How can the interdictor infer customer demands over time?
- Which arcs should be disrupted to most effectively hinder illicit cargo movement?
- How does the LSP adjust routing decisions in response to interdiction?
- What vehicle assignments minimize operational costs in an interdicted network?

To address these questions, we propose a repeated, learning-based, bi-level model and develop a heuristic solution algorithm capable of handling large-scale instances.

The remainder of this paper is structured as follows: Section 2 reviews the existing literature, while Section 3 formulates the proposed bi-level programming model. Section 4 elaborates on the development of the learning-based heuristic algorithm. Computational results and comparative analyses are reported in Section 5. Finally, Section 6 offers concluding remarks and suggests avenues for future research.

## 2. Literature Review

Network interdiction problems describe adversarial settings in which a defender, typically a logistics service provider (LSP), aims to operate a transportation system efficiently, while an interdictor attempts to degrade this performance by disrupting specific network components. Within this context, NIVRPs focus on how arc disruptions or penalties affect routing decisions and overall network performance.

### 2.1. Complete-Information Interdiction Models

Most early studies on interdicted VRPs assume that both players possess complete information about the network and operational parameters. Under this assumption, several models have examined how arc penalties influence routing decisions in hazardous-material transportation settings. For example, the interaction between a regulatory agency and an LSP has been formulated as a bi-level optimization problem and solved using metaheuristic approaches (Kheirkhah et al., 2016a).

Related work has investigated destructive interdiction strategies designed to prevent the movement of illicit or risky cargo. In such settings, interdiction decisions aim to block or disrupt selected routes, and exact approaches based on Benders' decomposition have been proposed to solve the resulting optimization problem (Kheirkhah et al., 2016b).

The literature has also evolved toward more complex hierarchical decision structures. In particular, interdiction has been studied in tri-level settings where depot protection and attack decisions precede routing optimization. These models capture strategic interactions in which the defender first allocates protection resources, and the attacker subsequently selects depots to

interdict in order to maximize unmet demand (Sadati et al., 2020a). Extensions of this framework further incorporate fortification decisions at the upper level before interdiction actions occur (Sadati et al., 2020b).

Beyond depot-focused interdiction, researchers have also considered arc disruptions in integrated location–routing problems. In these models, interdiction decisions influence both routing patterns and facility-location choices, demonstrating the broader impact of network disruptions on logistics system design. Similar perspectives have been explored in studies addressing arc interdiction within location-routing frameworks (Nadizadeh and Sabzevari Zadeh, 2021).

### 2.2. Incomplete-Information Interdiction Models and Learning

Compared with the extensive literature under complete information, studies addressing incomplete-information settings remain limited. One contribution in this direction considers adversarial VRPs where the interdictor does not have full knowledge of the evader's parameters; however, the interaction is modeled as a static game and does not incorporate learning across repeated encounters (Bidgoli and Kheirkhah, 2018).

Learning and adaptation have been investigated more extensively in broader security and online decision-making environments. Some studies have examined how limited adaptability affects attacker performance in adversarial systems (Haskell et al., 2014), while related research has analyzed the role of adaptive strategies in adversarial optimization settings.

In the online learning literature, decision-makers often operate under uncertainty about adversarial actions and rely on feedback mechanisms to update their strategies over time. For instance, several studies have developed algorithms for online optimization assuming oblivious adversaries and specific feedback structures, such as full-information or semi-bandit feedback (Kakade et al., 2009). More recent work has proposed learning algorithms for repeated games without prior knowledge, although these approaches typically assume simultaneous decision-making rather than sequential interactions (Guo et al., 2017).

Motivated by these developments, the present study introduces a learning-based framework for the NIVRP in which the interdictor gradually updates demand estimates based on observed outcomes over multiple rounds. By combining interdiction modeling with adaptive learning mechanisms within a Stackelberg structure, the proposed approach addresses an important gap in the existing literature.

## 3. A Bi-Level Programming Model for NIVRP without Complete Prior Knowledge

In this study, the vehicle routing problem is modeled on a network where a single depot and a set of demand points (customers) are connected via multiple arcs. The interdictor's objective is to detect and intercept vehicles at the earliest possible stage of their routes, while the LSP seeks to satisfy all predetermined customer demands and minimize the total unmet demand. The problem involves a "legislator" (the interdictor) who strategically deploys checkpoints along arcs to intercept vehicles carrying illicit or unauthorized cargo, thereby preventing them from completing their routes.

Although the legislator typically oversees a large-scale, multi-player network involving numerous LSPs, we simplify this interaction into a one-to-one game without loss of generality. This simplification is justified by the fact that LSPs make routing decisions independently after observing the checkpoint configuration; thus, the presence of other LSPs does not structurally alter the decision-making logic of a single agent.

The modeling assumptions adopted in this study are grounded in established streams of the vehicle routing and interdiction literature. The use of bi-level or leader–follower structures to represent adversarial interactions between an LSP and an interdictor is well-documented in

security-oriented routing research (Bidgoli and Kheirkhah, 2018). Specifically, Stackelberg game formulations are widely employed to capture the sequential nature of defensive and offensive decisions in transportation networks (Tambe, 2011).

In our framework, the interdictor (leader) moves first to deploy checkpoints, followed by the LSP's (follower) routing response. This sequence aligns with standard interdiction-based VRP models, where the interdictor leverages information regarding potential vehicle movements. Furthermore, modeling incomplete or uncertain customer demands is a cornerstone of stochastic and robust VRP variants, where demand information is progressively revealed through operational feedback (Gendreau et al., 2016). Finally, the application of online demand estimation based on partial observations is consistent with learning-based routing and semi-bandit frameworks, where only limited feedback, such as residual vehicle load or arc-level observations, is available after each round (Bertsimas and Kallus, 2020).

The interaction is modeled as a Repeated Network Interdiction Game (NIG), a Stackelberg game played over $T$ rounds. Initially, both players lack full prior knowledge of their opponent's parameters. In each round, the leader (interdictor) makes decisions based on estimates of customer demands and vehicle types. These estimates are used to predict the LSP's routes and determine optimal roadblock locations. Subsequently, the interdictor receives feedback from the completed round and updates their demand estimates, a mechanism referred to as semi-bandit feedback.

### 3.1. Model Characteristics and Assumptions

The problem exhibits the following characteristics and assumptions:

(1) We consider a capacitated vehicle routing setting with a single central depot.
(2) A set of $V$ heterogeneous vehicles is available, each with a specific load capacity $Q^v$ (in tons).
(3) The primary objective of the LSP is to minimize total unmet demand.
(4) Customer demands are deterministic from the LSP's perspective, and demand splitting is not permitted; each node is visited exactly once, provided the vehicle has sufficient remaining capacity.
(5) The model follows a discrete network interdiction structure with limited interdiction resources.
(6) The interdictor does not possess complete prior knowledge of customer demands and must infer them through repeated interactions.

Let $\mathcal{C}$ denote the set of customer nodes and $o$ denote the depot. The transportation network is represented by a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N} = \mathcal{C} \cup \{o\}$. Vehicles depart from the depot, visit a subset of customers, and return. The indices, sets, decision variables, and parameters are summarized in Tables 1 and 2.

**Table 1.** Notations of the variables

| Notation | Definition |
|---|---|
| $z_{ijt}^{v}$ | Binary variable equal to 1 if arc $ij$ traversed by vehicle $v$ is interdicted by the interdictor in round $t$, and 0 otherwise |
| $x_{ijt}^{v}$ | Binary routing variable equal to 1 if vehicle $v$ traverses arc $ij$ in round $t$, and 0 otherwise |
| $l_{jt}^{v}$ | Remaining load carried by vehicle $v$ after visiting node $j$ in round $t$ (in tons) |

**Table 2.** Notations of the parameters

| Notation | Definition |
|---|---|
| $i, j$ | Indices of nodes in the transportation network |
| $t$ | Index of decision rounds (periods), $t \in \{1, \dots, T\}$ |
| $v$ | The index for vehicles, $v \in \{1, \dots, V\}$ |
| $C$ | The set of customer nodes |
| $\{o\}$ | Depot node |

| | |
|---|---|
| $Q^v$ | Load capacity of vehicle $v$ (in tons) |
| $D_{it}$ | Actual demand of customer $i$ in round $t$ |
| $cu$ | Unit penalty cost associated with one unit of unmet demand |
| $bud_t$ | Total interdiction budget available to the interdictor in round $t$ |
| $b_{ij}$ | Budget required to interdict arc $ij$ |
| $c_{ij}$ | Transportation cost of traversing arc $ij$ |
| $\hat{x}_{ijt}^v$ | Predicted routing decision of vehicle $v$ on arc $ij$at round $t$, estimated by the interdictor |
| $\hat{l}_{it}^v$ | Predicted remaining load of vehicle $v$ after visiting node $i$ at round $t$, estimated by the interdictor |

Accordingly, a bi-level programming model is developed to capture each round of this Stackelberg game within the NIVRP framework. The interaction between the leader (interdictor) and the follower (LSP) in a given round is formulated as a bi-level optimization problem consistent with the Stackelberg structure. For each round $t$, both players determine their optimal strategies by solving the proposed bi-level model. Following the round's conclusion, the interdictor updates its demand predictions for round $t+1$ based on the information acquired in prior periods. Specifically, the sequence of events within a single round proceeds as follows:

1. **Leader's decision:** The interdictor (leader) deploys checkpoints on selected arcs based on its current estimates of customer demands and anticipated routing patterns.
2. **Follower's response:** After observing the interdiction configuration, the LSP (follower) determines its routing plan for the round.
3. **Interdiction realization:** The interdictor's prepositioned checkpoints are activated on the arcs traversed by the LSP's vehicles.
4. **Feedback collection:** If a vehicle is intercepted on an arc, its remaining load at the interception point cannot be used to satisfy subsequent customer demands. This residual load is then observed by the interdictor and utilized as partial (semi-bandit) feedback to refine demand estimates for future rounds.

$$\boldsymbol{max}\ \ \boldsymbol{Z_{1t}} = \sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \sum_{\boldsymbol{i\in C}} \sum_{\boldsymbol{j\in C}} \boldsymbol{\hat{l}_{it}^v z_{ijt}^v} \tag{1}$$

$$\boldsymbol{s.t.} \qquad \boldsymbol{z_{ijt}^v \le \hat{x}_{ijt}^v}, \quad \boldsymbol{\forall i,j \in C},\ \ \boldsymbol{\forall v \in V} \tag{2}$$

$$\sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \sum_{\boldsymbol{i\in C}} \sum_{\boldsymbol{j\in C}} \boldsymbol{b_{ij} z_{ijt}^v \le bud_t}, \tag{3}$$

$$\boldsymbol{z_{ijt}^v \in \{0,1\}}, \qquad \boldsymbol{\forall i,j \in C},\ \ \boldsymbol{\forall v \in V} \tag{4}$$

$$\boldsymbol{min\, Z_{2t}} = \sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \left( \sum_{\boldsymbol{j\in C}} \sum_{\boldsymbol{i\in C}} \boldsymbol{l_{it}^v z_{ijt}^v} \right) \boldsymbol{.cu} + \sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \left( \sum_{\boldsymbol{j\in C}} \sum_{\boldsymbol{i\in C}} \boldsymbol{c_{ij} x_{jit}^v} \right) \tag{5}$$

$$\boldsymbol{s.t.} \quad \sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \sum_{\boldsymbol{j\in C\cup\{o\}}} \boldsymbol{x_{ijt}^v = 1}, \qquad \boldsymbol{\forall i \in C} \tag{6}$$

$$\sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \sum_{\boldsymbol{j\in C\cup\{o\}}} \boldsymbol{x_{jit}^v} - \sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \sum_{\boldsymbol{j\in C\cup\{o\}}} \boldsymbol{x_{ijt}^v = 0}, \qquad \boldsymbol{\forall i \in C} \tag{7}$$

$$\sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \sum_{\boldsymbol{j\in C\cup\{o\}}} \boldsymbol{x_{0jt}^v \le V} \tag{8}$$

$$\sum_{\boldsymbol{v=1}}^{\boldsymbol{V}} \sum_{\boldsymbol{j\in C\cup\{o\}}} \boldsymbol{x_{i0t}^v \le V} \tag{9}$$

$$\boldsymbol{l_{jt}^v \ge l_{it}^v - D_{jt} x_{ijt}^v - Q^v\left(1 - x_{ijt}^v\right)}, \qquad \boldsymbol{\forall i,j \in C \cup \{o\}},\ \ \boldsymbol{\forall v \in V} \tag{10}$$

$$\boldsymbol{l_{0t}^v \le Q^v}, \qquad \boldsymbol{\forall v \in V} \tag{11}$$

$$\boldsymbol{0 \le l_{jt}^v \le Q^v(1 - z_{ijt}^v)}, \qquad \boldsymbol{\forall i,j \in C},\ \ \boldsymbol{\forall v \in V} \tag{12}$$

$$\boldsymbol{x_{ijt}^v \in \{0,1\},\ l_{jt}^v \ge 0} \qquad \boldsymbol{\forall i,j \in C \cup \{o\}},\ \ \boldsymbol{\forall v \in V}. \tag{13}$$

Here, the leader (interdictor) seeks to maximize the intercepted flow, as defined in objective function (1), whereas the follower (LSP) minimizes the total economic loss, consisting of transportation costs and unmet customer demand, as represented in objective function (5). The interdictor attempts to intercept offending vehicles as early as possible along their routes.

Constraints (2) ensure that an arc can be interdicted only if it is traversed by a vehicle in the LSP's routing plan. Constraints (3) enforce the interdiction budget limitation, ensuring that the total cost of interdicted arcs does not exceed the available interdiction budget. Constraints (4) define the interdiction decision variables as binary. On the follower's side, constraints (6) ensure that each customer is served at most once by a single vehicle, thereby prohibiting demand splitting. Constraint (7) establishes flow conservation at each customer node, ensuring that whenever a vehicle enters a node, it must also leave it. Constraints (8) and (9) limit the total number of routes to the maximum number of available vehicles. Constraints (10) enforce load balance, ensuring that a vehicle has sufficient remaining capacity to satisfy the demand of each visited customer. Constraints (11) restrict the total load carried by each vehicle to its capacity limit. Constraint (12) represents the post-interdiction state of a vehicle. Specifically, if a vehicle is intercepted on arc $ij$, the remaining load that would otherwise be used to serve subsequent customers becomes unavailable for delivery beyond that point. Consequently, the usable load of the vehicle is forced to zero after interdiction. Finally, constraints (13) specify that routing decision variables are binary, while the remaining vehicle load variables are non-negative continuous variables.

## 4. The Developed Solution Methodology for the NIVRP

The vehicle routing problem (VRP) and many of its variants are known to be NP-hard problems (Farrokhi-Asl et al., 2017). Consequently, exact approaches often become computationally intractable for large-scale instances. Metaheuristic algorithms, however, have proven effective in addressing the computational complexity of various VRP variants and are widely used to solve bi-level optimization problems within reasonable computational time. Guo et al. (2017) proposed a learning algorithm, referred to as the SBGA, for repeated network interdiction games with semi-bandit feedback. In this approach, the leader progressively improves its knowledge of unknown parameters by learning from the information collected in previous rounds.

Building on this idea, the present study develops a metaheuristic framework consisting of two main components executed at each round: (i) a learning mechanism that updates estimates of the unknown parameters, and (ii) a bi-level co-evolutionary algorithm designed to solve the single-round NIVRP. The overall procedure of the proposed framework is summarized in the pseudocode presented in Algorithm 1.

| **Algorithm 1.** The overall procedure of the proposed framework | |
|---|---|
| 1 | **Procedure** Game (Network, vehicle-specific parameters, customers' parameters) |
| 2 | $D_{i1} \leftarrow random\ values$ |
| 3 | Run Bi-level Co-evolutionary algorithm. |
| 4 | Update equilibrium for two players: $(x^v_{ij1}, z^v_{ij1}) \leftarrow (Routes, Interdicted\ arcs)$ |
| 5 | **Repeat** |
| 6 | $Current\ period \leftarrow t$ |
| 7 | Process the information collected from periods $1, \dots, t-1$ |
| 8 | Run the learning process. |
| 9 | $D_{it} \leftarrow learned\ values$ |
| 10 | Run Bi-level Co-evolutionary algorithm. |
| 11 | Update equilibrium for two players: $(x^v_{ijt}, z^v_{ijt}) \leftarrow (Routes, Interdicted\ arcs)$ |
| 12 | $t \leftarrow t+1$ |
| 13 | **Until** $t > T$ |
| 14 | **Return** $(Routes, Interdicted\ arcs)$ for all periods |
| 15 | **End procedure** |

In this study, the interdictor does not possess complete information regarding customer demands. Accordingly, we aim to leverage the observable feedback collected in previous rounds to infer these unknown demand values. To address this challenge, Subsection 4.1 introduces an estimator that updates the interdictor's belief about customer demands by exploiting the available historical feedback. Based on these estimated values, we formulate an imaginary bi-level game in which both players interact strategically.

To solve this imaginary game, we develop a bi-level co-evolutionary algorithm that enables the leader and follower to make online decisions in each round. Subsections 4.2 and 4.3 present the details of this approach, where a bi-level genetic algorithm is designed to efficiently handle the nested structure of the problem.

### 4.1. Learning Framework for Predicting Routes and Demands at Each Round

In the repeated NIVRP game, the interdictor does not have complete information about the customers' demands. However, after each interdiction event, the interdictor observes the amount of seized cargo on the intercepted arc. This information provides semi-bandit feedback, which can be used to gradually improve demand estimates over time. Based on these observations, a prediction-response interaction emerges between the interdictor (leader) and the LSP (follower) in each round of the game. At the beginning of round $t$, the interdictor predicts customer demands and the likely routes of the LSP using the information collected in previous rounds. These predictions are then used to determine the arcs that are most likely to contain large shipment flows.

In the first round, the interdictor relies on its initial belief about customer demand. Unlike Bidgoli and Kheirkhah (2018), where the interdiction decisions are based only on this initial information, the proposed framework updates the demand estimates dynamically across rounds. Specifically, after an interdiction occurs, the remaining load of the intercepted vehicle provides partial information about which customers have already been served and which customers remain unvisited. This information is then used to update the predicted demand distribution for the next round.

Algorithm 2 summarizes the proposed online estimation procedure. At each round, the interdictor first updates the observed interdicted arcs and the information collected from surveillance cameras. Using these observations, the interdictor predicts the remaining route of the intercepted vehicles and updates the demand estimates through a smoothing rule that combines the previous demand belief with the newly observed residual load information.

Once the predictions are obtained, the interdictor solves the interdiction problem to identify the arcs with the highest expected intercepted load. After observing the interdiction configuration, the LSP determines its routing plan for that round. Both players are assumed to be rational and adversarial, aiming to maximize their respective payoffs.

| **Algorithm 2.** Developed Online Demand-Estimation Mechanism for the Repeated NIVRP Game | | |
|---|---|---|
| 1 | **Procedure** ($t$, ($D_{j1}, \dots, D_{jt-1}$), Network, interdicted vehicles and flows, surveillance cameras information, vehicle-specific parameters, customers' parameters) | |
| 2 | $\widehat{D}_{jt} \leftarrow$ *The initial demands based on the demand distribution* | |
| 3 | *Current period* $\leftarrow t$ | |
| 4 | Update *Interdicted arcs*. | |
| 5 | Update the information on the interdicted vehicles observed by the surveillance cameras. | |
| 6 | Predict the remained routes for the unobserved nodes by the interdicted vehicles. | |
| 7 | Update the predicted demands based on the unobserved nodes and the interdicted flow. | |
| 8 | $\widehat{D}_{jt} \leftarrow$ *learned values* | |
| 9 | **Return** $\widehat{D}_{jt}$ for period $t$ | |
| 10 | **End procedure** | |

To illustrate the prediction mechanism, consider an example with two vehicles and ten customers, as shown in Figure 1. Assume that vehicle $v_1$is observed by surveillance cameras

on arcs (11–1) and (1–5) before being interdicted on arc (5–6), while vehicle $v_2$is observed on arc (11–2) before being interdicted on arc (2–7) in period $t-1$. Based on these observations, the set of customers visited and unvisited by each vehicle can be inferred.

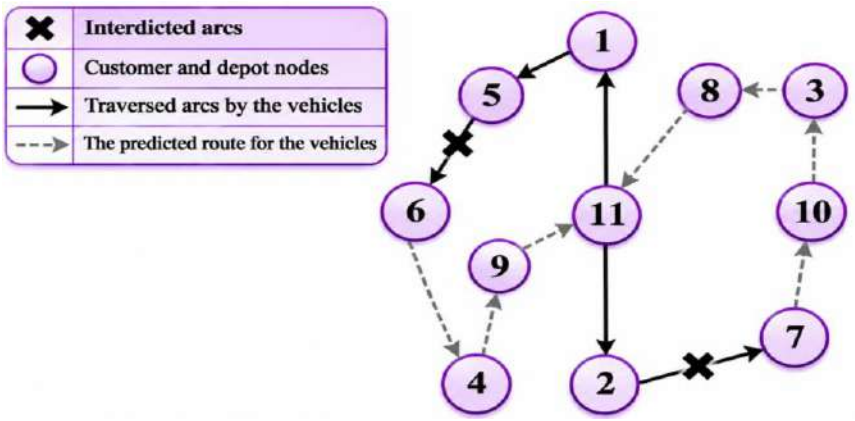



**Figure 1.** An instance with two vehicles for prediction the remainder routes

Let $Visited_{v,t-1}$ and $Unvisited_{v,t-1}$ denote the sets of customers already served and not yet served by vehicle $v$ before the interdiction occurs. If $l^v_{t-1}$ represents the remaining load of vehicle $v$ at the moment of interception, the predicted customer demands are updated according to Eqs. (14)-(17). In these equations, the remaining vehicle load indicates the amount of demand that has not yet been delivered, while the difference between vehicle capacity and remaining load corresponds to the amount already served.

Accordingly, the predicted demand for customers in the visited and unvisited sets is updated proportionally to their previous demand estimates and the observed intercepted load. This update rule provides an adaptive estimation of customer demand using the partial feedback obtained from interdicted vehicles.

$$\widehat{D}_{j1} = \widehat{D}_{j0}, \qquad \forall j \in C \tag{14}$$

$$D'_{jt} = \frac{\widehat{D}_{jt-1}}{\sum_{k\in Visited_{v,t-1}} \widehat{D}_{kt-1}} . (Q^v - l^v_{t-1}), \qquad \forall j \in Visited_{v,t-1} \tag{15}$$

$$D'_{jt} = \frac{\widehat{D}_{jt-1}}{\sum_{k\in Unvisited_{v,t-1}} \widehat{D}_{kt-1}} . l^v_{t-1}, \qquad \forall j \in Unvisited_{v,t-1} \tag{16}$$

$$\widehat{D}_{jt} = D'_{jt}, \qquad \forall j \in C \tag{17}$$

Based on relations (14)-(17), the predicted demands for all customers in period $t = 2$ can be computed as illustrated in Figure 2.

| Customers $j$ | | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $Visited_{v,t-1}$ | $Unvisited_{v,t-1}$ | | | | | | | | | | | |
| $t=1$ | $\widehat{D}_{j1}$ | | | | **350** | **100** | **450** | **450** | **350** | **450** | **350** | **250** | **500** | **250** |
| $t=2$ | $D'_{j2}$ | $v_1$ | {1, 5, 10} | {6, 4} | 368 | | | 500 | 368 | 500 | | | | 263 |
| | | $v_2$ | {2, 8} | {3, 7, 9} | | 86 | 370 | | | | 323 | 214 | 462 | |
| | $\widehat{D}_{j2}$ | | | | **368** | **86** | **370** | **500** | **368** | **500** | **323** | **214** | **462** | **263** |

$Visited_{v,t-1}$ and $Unvisited_{v,t-1}$ denote the visited and unvisited customer sets.

**Figure 2.** Prediction of demands for period $t$ based on the collected information from period $t-1$

### 4.2. Convergence Discussion

Although the proposed update rule is not intended to serve as a complete learning algorithm, it demonstrates stable estimation behavior under repeated observations. The demand estimation mechanism in Eqs. (14)-(17) updates the predicted customer demands using partial information inferred from the residual load of interdicted vehicles, which can be interpreted as semi-bandit feedback.

Let $\widehat{D}_j^t$ denote the estimated demand of customer $j$ at iteration $t$. Since the update rule redistributes the observed residual vehicle load proportionally among visited and unvisited customers while preserving the vehicle capacity, the resulting sequence of demand estimates remains bounded. Therefore, the vector of estimated demands lies within a compact feasible region. Under the assumption that customer demands are stationary and that observations of residual vehicle load are unbiased, repeated updates progressively incorporate new information into the estimates. In this respect, the estimation process resembles classical stochastic approximation schemes based on partial observations.

Although a formal convergence proof is beyond the scope of this study, the boundedness of the update rule and its normalization structure suggest that the estimates tend to stabilize as the number of observations increases. This claim is further supported by the computational experiments, which demonstrate rapid empirical stabilization of the predicted demands across all tested network sizes.

### 4.3. A Remark on the Realism of the Learning Assumption

The proposed learning mechanism implicitly assumes that interdicted flows and surveillance-based observations provide sufficiently informative signals to progressively estimate customer demands. We acknowledge that, in real-world operational environments, such information can be noisy, delayed, or only partially observable, and exact demand recovery may not always be achievable. However, the proposed framework does not require exact or near-exact demand estimation. Rather, it leverages the accumulation of informative feedback over repeated interactions to refine the interdictor's belief about underlying demand patterns and the follower's likely routing behavior.

This assumption is reasonable in practical settings where interdiction points are equipped with monitoring technologies, such as license-plate recognition systems, weigh-in-motion sensors, or inspection checkpoints, that generate structured but imperfect data on vehicle loads and traversal histories. In such scenarios, interdicted flows serve as informative signals, not precise measurements, enabling gradual learning over time instead of immediate inference.

Furthermore, the learning component can be extended to incorporate weaker or more realistic information structures. For example, demand observations may be affected by stochastic noise, systematic bias, or bounded estimation errors. In such cases, the update rule could be adapted to operate with probabilistic or interval-valued demand estimates instead of deterministic point values. Importantly, these extensions do not alter the underlying bi-level problem structure or the co-evolutionary algorithmic framework. They instead represent promising avenues for future research on robust learning mechanisms within network interdiction settings.

### 4.4. The Bi-Level Co-Evolutionary Algorithm for Each Round of the Repeated NIVRP

To address the repeated NIVRP, we extend the COBRA algorithm (Legillon et al., 2012) to incrementally evolve the sub-populations of both levels in a parallel framework, with information exchange between the leader and follower occurring at the end of each iteration. We develop two distinct Genetic Algorithms (GAs) to solve the upper-level interdiction problem and the lower-level vehicle routing problem (VRP) with heterogeneous fleets.

The selection of GAs for this framework is motivated by the structural complexity of the

NIVRP rather than general exploration-exploitation requirements. Both the interdiction problem and the VRP are NP-hard, characterized by non-convex, combinatorial, and discrete decision spaces. Furthermore, the problem is governed by complex logical and capacity constraints that render exact methods computationally prohibitive for realistically sized instances. GAs offers flexible encoding schemes that allow for the simultaneous representation of routing plans, vehicle assignments, and interdiction actions, while feasibility is maintained through tailored repair and decoding operators.

Moreover, the co-evolutionary nature of our framework is naturally aligned with population-based evolutionary search. This allows for adaptive strategy updates for both the interdictor and the LSP across repeated game rounds. In contrast, alternative metaheuristics such as particle swarm optimization or differential evolution are primarily designed for continuous decision spaces. Their application to this problem would require complex discretization mechanisms, which typically complicate constraint handling and feasibility preservation. Consequently, GAs represents a pragmatic and structurally compatible choice for this bi-level, learning-based interdiction framework.

From an algorithmic perspective, the population-based nature of GAs is particularly advantageous, as it facilitates the exchange of information between the leader and follower populations. This mechanism is essential for capturing the adaptive behavior of both decision-makers, as the improved populations are evaluated to determine the most effective strategies co-evolved at the end of each iteration.

### 4.4.1. Solution Encoding (Upper & Lower levels) at Each Round

Figure 3 displays a solution for the VRP with ten customers {1,2,3,4,5,6,7,8,9,10}, and depot {11}, and three types of vehicles.

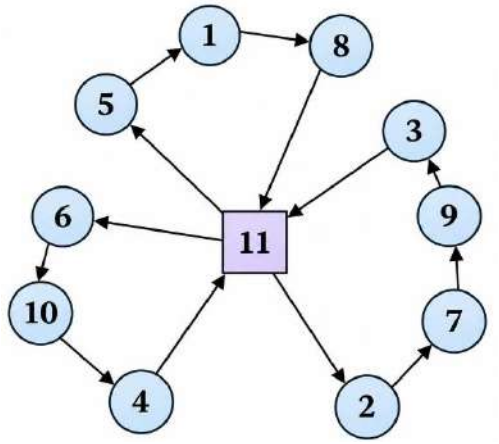



**Figure 3.** A VRP instance with three vehicles, and ten customers

To encode the solutions, we employ a permutation-based representation involving customers and the depot. In this scheme, the assignment of customers to each route and their visit sequence are explicitly defined. Figure 4 illustrates a solution structure for the VRP instance. Each row corresponds to a specific vehicle route, and the total number of rows is constrained by the maximum number of available vehicles. The sequence of customers visited by each vehicle is represented in its corresponding row. As shown in the figure, the depot is represented by the integer 11 within the permutation.

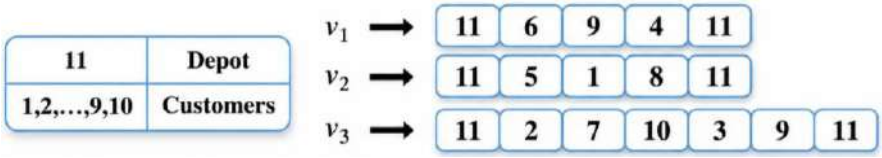
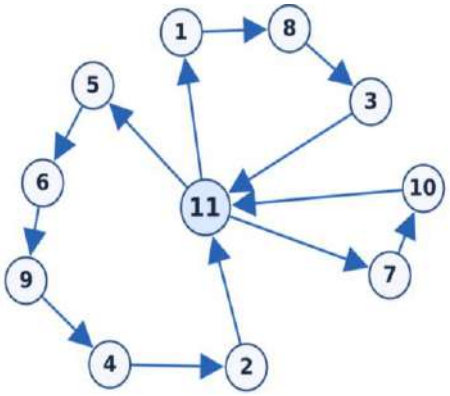



**Figure 4.** An instance of a LSP chromosome

This encoding scheme ensures that several structural constraints are implicitly satisfied, eliminating the need for explicit verification during the evolutionary process. Specifically: (i) Since each row in the representation corresponds to a single route for the heterogeneous fleet, the total number of routes inherently remains within the available vehicle limit. (ii) As each customer node appears in exactly one row, the requirement that each customer is served by only one vehicle is naturally enforced.

However, feasibility regarding vehicle capacity must still be validated to ensure that each vehicle possesses sufficient load to satisfy the demands of all assigned customers. In this model, we assume that each vehicle is fully loaded upon departing from the depot. Furthermore, to streamline the application of genetic operators to the interdiction population, we index the arcs of the complete graph as illustrated in Figure 5.

| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 1 | 12 | 23 | 34 | 45 | 56 | 67 | 78 | 89 | 100 | 111 |
| **2** | 2 | 13 | 24 | 35 | 46 | 57 | 68 | 79 | 90 | 101 | 112 |
| **3** | 3 | 14 | 25 | 36 | 47 | 58 | 69 | 80 | 91 | 102 | 113 |
| **4** | 4 | 15 | 26 | 37 | 48 | 59 | 70 | 81 | 92 | 103 | 114 |
| **5** | 5 | 16 | 27 | 38 | 49 | 60 | 71 | 82 | 93 | 104 | 115 |
| **6** | 6 | 17 | 28 | 39 | 50 | 61 | 72 | 83 | 94 | 105 | 116 |
| **7** | 7 | 18 | 29 | 40 | 51 | 62 | 73 | 84 | 95 | 106 | 117 |
| **8** | 8 | 19 | 30 | 41 | 52 | 63 | 74 | 85 | 96 | 107 | 118 |
| **9** | 9 | 20 | 31 | 42 | 53 | 64 | 75 | 86 | 97 | 108 | 119 |
| **10** | 10 | 21 | 32 | 43 | 54 | 65 | 76 | 87 | 98 | 109 | 120 |
| **11** | 11 | 22 | 33 | 44 | 55 | 66 | 77 | 88 | 99 | 110 | 121 |



Predicted route for period t

**Figure 5.** An instance of the interdictor chromosome for period t

Since certain arcs in the complete graph may not exist in the physical network, or may not be traversed by vehicles within the predicted routing plans, interdicting such arcs is considered infeasible. Accordingly, only the existing arcs, represented by the colored cells in Figure 6, are considered candidates for interdiction. In our algorithm, crossover and mutation operators are applied uniformly at random across these admissible interdiction arcs, ensuring that all generated solutions satisfy both operational and budget constraints.

Assuming a maximum of five interdictions is permitted per period, an example interdiction strategy is illustrated in Figure 6. In this instance, the interdictor deploys roadblocks on arcs (11,1), (11,7), (10,6), (8,10), and (11,3).

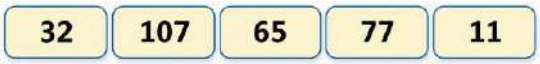



**Figure 6**. A solution seed for interdiction

Initial populations for both levels are generated randomly and evaluated based on their respective objective functions. Subsequently, selection, mutation, and crossover operators are applied to generate offspring, as detailed in the following subsections.

### 4.4.2. Genetic Operators and Repair Mechanisms

At both levels, tournament selection is employed to select parents due to its robustness and low computational overhead. Offspring are generated using problem-specific operators designed to preserve route feasibility, vehicle assignment consistency, and interdiction admissibility. Mutation operators introduce localized perturbations, such as arc swaps (upper-level), as well as route reordering, customer inversion, and vehicle reassignment (lower-level). All genetic operators are applied uniformly at random across the set of admissible decision components.

*Upper-level Genetic operators***:** In each generation, crossover and mutation operators are applied to the interdiction population, while an elitist strategy preserves the best-performing interdiction plan discovered to date. Newly generated solutions are validated against the maximum interdiction budget; any solution violating this constraint is discarded. The remaining individuals are evaluated and ranked according to the upper-level objective function, and a subset of top-performing solutions is selected to form the next generation. Illustrative examples of these crossover and mutation operators are provided in Figures 7 and 8.

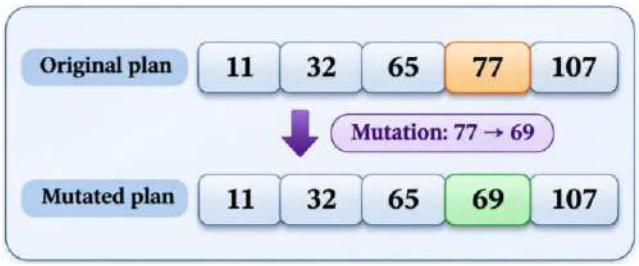



**Figure 7.** The mutation for interdiction plan

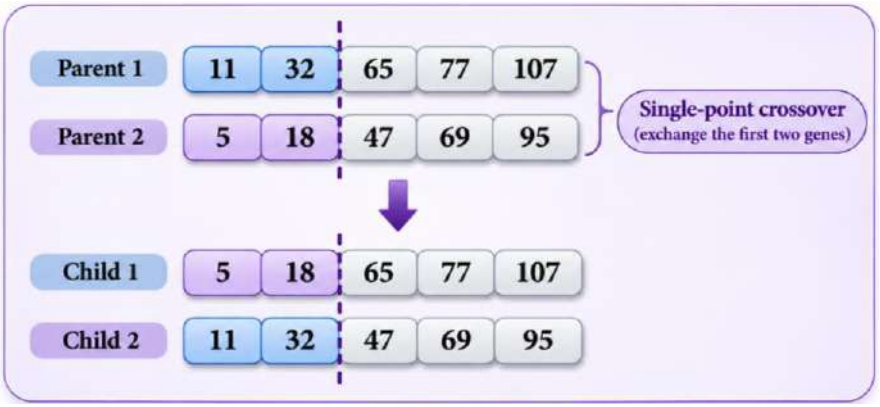



**Figure 8.** The crossover operator for interdiction

*Genetic operators for the VRP*: A dedicated GA is developed for the VRP with heterogeneous vehicles. Three mutation operators are employed: single-point mutation, inversion, and replacement, enabling both local and structural modifications of the routes. These operators facilitate efficient exploration of routing and vehicle-assignment decisions under interdicted network conditions. Representative examples of the lower-level genetic operators are shown in Figures 9–11.

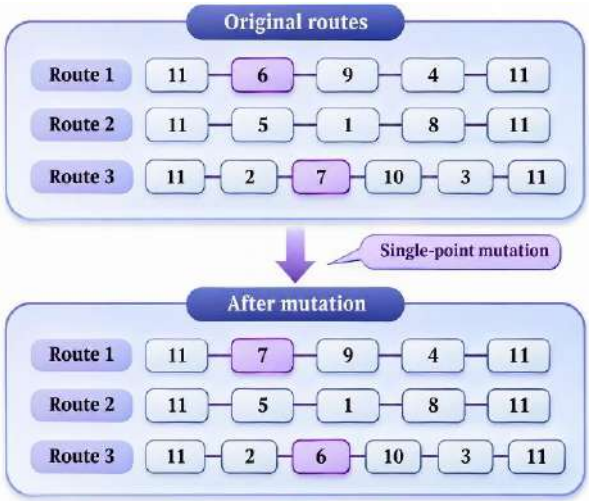



**Figure 9.** Single point mutation for VRP

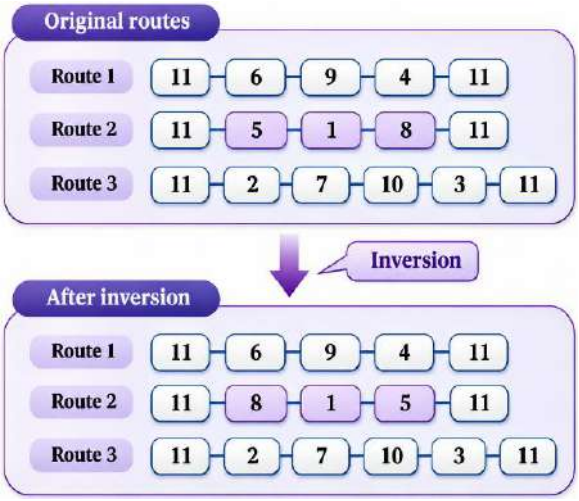



**Figure 10.** Inversion for VRP

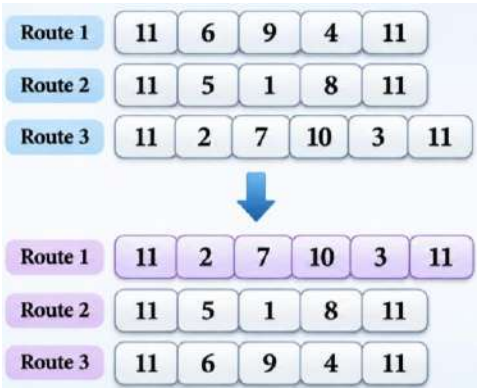



**Figure 11.** Replacement for VRP

*Repair and feasibility:* Given that feasibility is governed by complex routing, connectivity, and capacity constraints, a dedicated repair mechanism is triggered whenever infeasible offspring are generated. The repair process follows a sequential procedure: it first enforces vehicle capacity limits, then identifies and removes disconnected route segments resulting from interdictions, and finally reconstructs feasible routes using shortest-path heuristics within the residual network. This systematic approach ensures that all individuals entering the evaluation phase satisfy the operational requirements of the NIVRP.

### 4.4.3. Co-evolutionary Information Exchange

The proposed framework adopts an elitist and symmetric co-evolutionary strategy. The upper- and lower-level populations evolve independently for a fixed number of generations, periodically exchanging their respective best-found solutions. This synchronous information exchange enables each decision-maker to adapt its strategy in response to the opponent's most recent behavior, mimicking the strategic interaction of the game. Following each exchange, the best co-evolved solutions are archived for further analysis. A schematic representation of the co-evolutionary procedure is provided in Appendix A.

### 4.4.4. Stopping Criteria and Parameter Settings

The genetic algorithms at both levels terminate when either the maximum number of generations is reached or no improvement in the best fitness value is observed for a predefined number of consecutive generations. To ensure optimal performance, algorithmic parameters are tuned through preliminary experiments on a subset of representative instances. The selected values reflect a carefully calibrated trade-off between solution quality and computational efficiency. To maintain consistency and ensure a fair comparison across all test cases, these parameters are kept constant throughout the study. Table 3 summarizes the final parameter settings used in the computational experiments.

**Table 3.** Genetic algorithm parameters

| Parameter | Upper level | Lower level |
|---|---|---|
| Population size | 300 | 300 |
| Mutation rate | 0.3 | 0.3 |
| Max generations | 50 | 50 |
| Crossover rate | 0.4 | - |
| Inversion rate | - | 0.4 |
| Replacement rate | - | 0.4 |

## 5. Computational Experiments

In this section, we evaluate the computational performance of the proposed algorithm. To

validate the effectiveness of the approach and assess its behavior under different conditions, the algorithm is applied to a set of test instances.

Although testing on benchmark instances is generally desirable for comparative evaluation, an extensive review of the literature revealed that no publicly available benchmark instances fully match the specific characteristics of the problem studied in this paper. In the existing NIVRP literature, individual studies often introduce distinct modeling assumptions and operational constraints, which typically require modifications to standard benchmark instances.

Therefore, to ensure consistency with the requirements of our model, we generate a set of synthetic instances of varying sizes. These instances are designed to reflect the structural characteristics of the proposed problem setting. The details of the generated instances and their configurations are reported in Table 4.

**Table 4.** Parameters for the randomly generated

| Parameter | $d_{ij}$ | $Q^v$ | $cu$ | $\mathbf{bud_t}$ | $\mathbf{b_{ij}}$ | $\mathbf{D_{jt}}$ |
|---|---|---|---|---|---|---|
| Value | Uniform (10,20) | Uniform (30,35) | 50 | 160 | Uniform (15,25) | Uniform (10,15) |

From an implementation perspective, the proposed framework is developed in MATLAB R2023b. All computational experiments can be reproduced on a standard workstation equipped with an Intel Core i5 processor, 12 GB of RAM, and a 64-bit Windows 10 operating system. A selected subset of the experimental results is reported to identify the most effective learning configuration, followed by a sensitivity analysis to examine the behavior of the proposed algorithm under different parameter settings. All instance generators and source codes developed in this study are publicly available in a Zenodo repository† to ensure transparency and reproducibility. The repository includes: (i) the MATLAB implementation of the different learning-scenario algorithms for the network interdiction problem, and (ii) the randomly generated benchmark instances.

### 5.1. Reformulating the Co-evolutionary Algorithm of Bidgoli and Kheirkhah (2018)

To evaluate the performance of the proposed co-evolutionary algorithm, we compare its results with those of a state-of-the-art method from the literature. For this purpose, we adapt the co-evolutionary algorithm of Bidgoli and Kheirkhah (2018) for the NIVRP with asymmetric information to fit the specific requirements of the present study. The original algorithm is modified in the following directions:

Fleet structure and encoding scheme: Their algorithm was originally developed for a VRP with a homogeneous fleet. In contrast, we extend it to a heterogeneous fleet by modifying the solution encoding, mutation operator, and crossover operator. In addition, they used a binary matrix to represent the interdiction plan, whereas we represent the interdiction strategy as shown in Figure 7. This representation is more suitable for applying mutation and crossover operators.

Fitness definition: We consider a different fitness function. In the original study, the interdictor aims to destroy selected arcs in the network so as to minimize the maximum profit of the distributor. In our formulation, however, the interdictor seeks to maximize the total undelivered load to customers.

We compare the proposed online learning-based algorithm with this reformed version of the algorithm of Bidgoli and Kheirkhah (2018), in which learning is not considered. This comparison provides better insight into the impact of learning on the quality of decisions made by both players. In addition, we evaluate two different learning scenarios to assess the

†[https://doi.org/10.5281/zenodo.18502844]

effectiveness of the proposed learning mechanism. All algorithms are implemented and executed on a Core i5-5200U CPU at 2.2 GHz with 12 GB RAM. Selected results are reported to identify the most efficient learning scenario, and a sensitivity analysis is then conducted to examine the behavior of the algorithm.

### 5.2. Analyzing the Effect of Learning

This section examines the role of the learning mechanism in the studied game. Three scenarios are considered for estimating customer demands by the interdictor: (1) the proposed learning algorithm, (2) a moving average of the demands observed over the three previous periods, and (3) no learning, in which a fixed demand estimate is used for all customers. These scenarios are compared in terms of the LSP objective value, defined as the loss due to unmet demand and distribution cost, the interdictor objective value, defined as the number of unmet demands, and computational time (in seconds). The results for fifteen instances of different sizes are reported in Table 5.

**Table 5.** The results of different learning scenarios for $T = 10$

| J_V | The proposed learning algorithm | | | | The moving average-based learning algorithm | | | | Without learning (Bidgoli and Kheirkhah, 2018) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | LSP | Interdictor objective | | Risk | LSP | Interdictor objective | | Risk | LSP | Interdictor objective | | Risk |
| | | Predicted | Real | | | Predicted | Real | | | Predicted | Real | |
| 10_2 | 459 | 156 | 141 | 0.106 | 459 | 156 | 141 | 0.106 | 473 | 156 | 141 | 0.106 |
| 20_4 | 719 | 233 | 232 | 0.004 | 857 | 278 | 221 | 0.258 | 843 | 278 | 232 | 0.198 |
| 30_5 | 1041 | 451 | 400 | 0.128 | 1213 | 451 | 400 | 0.128 | 1159 | 434 | 390 | 0.113 |
| 40_5 | 1347 | 396 | 379 | 0.045 | 1374 | 546 | 493 | 0.108 | 1449 | 530 | 481 | 0.102 |
| 40_6 | 1412 | 450 | 435 | 0.034 | 1448 | 506 | 438 | 0.155 | 1398 | 484 | 445 | ۰/۰۸۸ |
| 50_6 | 1572 | 542 | 526 | 0.030 | 1841 | 693 | 649 | 0.068 | 1844 | 697 | 645 | 0.081 |
| 50_7 | 1769 | 440 | 396 | 0.111 | 1783 | 564 | 493 | 0.144 | 1734 | 605 | 560 | 0.080 |
| 70_6 | 2363 | 752 | 719 | 0.046 | 2903 | 859 | 808 | 0.063 | 2515 | 774 | 692 | 0.118 |
| 70_8 | 2495 | 459 | 429 | 0.070 | 2511 | 784 | 667 | 0.175 | 2484 | 770 | 710 | 0.085 |
| 100_7 | 3309 | 680 | 675 | 0.007 | 3403 | 996 | 486 | 1.049 | 3492 | 938 | 860 | 0.091 |
| 150_7 | 5394 | 954 | 946 | 0.008 | 4846 | 1661 | 432 | 2.845 | 5161 | 1102 | 1213 | 0.092 |
| 175_8 | 6204 | 1115 | 1065 | 0.047 | 5831 | 1888 | 644 | 1.932 | 6423 | 1612 | 1473 | 0.094 |
| 200_8 | 7343 | 1561 | 1458 | 0.071 | 6484 | 1340 | 500 | 1.680 | 7238 | 1677 | 1602 | 0.047 |
| 250_9 | 8697 | 1306 | 1292 | 0.011 | 8241 | 2049 | 1867 | 0.097 | 8486 | 1558 | 1078 | 0.445 |
| 300_9 | 9997 | 1230 | 1143 | 0.076 | 10349 | 1765 | 1696 | 0.041 | 10510 | 1941 | 1833 | 0.059 |
| Lowest (Best) | | | | | Moderate | | | | Highest (Worst) | | | |
| High & Consistent | | | | | Inconsistent | | | | Often over/underestimates | | | |
| Very Low (Stable) | | | | | High in larger instances | | | | Moderate but not reliable | | | |

A more comprehensive comparison of the three learning scenarios reveals several important observations. First, the proposed learning algorithm consistently achieves lower risk values across almost all instances, demonstrating its ability to generate more accurate estimations of the LSP's routing decisions. In contrast, the moving-average learning strategy exhibits noticeably unstable behavior, particularly in larger instances, where its risk values fluctuate significantly. This indicates that relying solely on short-term historical information becomes increasingly unreliable as the network grows and the interdiction environment becomes more complex.

Second, from the interdictor's perspective, the proposed learning algorithm generally results in higher interdicted flows compared with the baseline approaches. This improvement reflects the interdictor's enhanced capability to identify critical arcs once more accurate demand predictions are available. Third, although the proposed method requires slightly higher computational effort than the static non-learning scenario, the additional CPU time remains

marginal relative to the performance gains achieved. Overall, these findings confirm that the proposed learning-based approach provides a more robust and stable decision-making mechanism than both the moving-average strategy and the non-learning model. It offers superior prediction accuracy, more effective interdiction decisions, and reliable performance across small, medium, and large network instances.

In the considered learning-based interdiction framework, the interdictor determines checkpoint locations based on predicted routes that the LSP is expected to select. However, because the LSP observes the checkpoint placements before making its final routing decision, the actual routes may differ from those predicted by the interdictor. This discrepancy introduces a form of decision uncertainty for the interdictor. To explicitly quantify this uncertainty, we introduce a risk metric that measures the discrepancy between the predicted and realized outcomes of interdiction decisions. Specifically, the risk value reflects the difference between the interdicted flow obtained based on the interdictor's predicted routes and the interdicted flow resulting from the actual routes chosen by the LSP. In this context, the risk metric provides a direct indicator of the accuracy of the interdictor's predictions and the effectiveness of the learning mechanism. Lower values of risk indicate a higher consistency between predicted and actual decisions, implying a more reliable learning process. The risk metric used in this study is therefore defined as follows:

$$\boldsymbol{Risk} = \frac{|\boldsymbol{Real} - \boldsymbol{Predicted}|}{\boldsymbol{Real}} \tag{18}$$

As shown in Figure 12, by exploiting more accurate and updated information, the proposed learning algorithm can substantially reduce decision-making risk in several instances compared with both the moving-average-based learning strategy and the non-learning baseline. Overall, the proposed learning algorithm outperforms the other two approaches by reducing losses for the LSP, improving interdiction outcomes, and maintaining a low predictive risk. These results indicate that it is the most effective and reliable option, demonstrating greater robustness, particularly in larger and more complex logistical environments.

In the more complex instances, particularly instances 10-13, the moving-average learning strategy exhibits pronounced spikes in risk, which makes it less dependable for strategic decision-making.

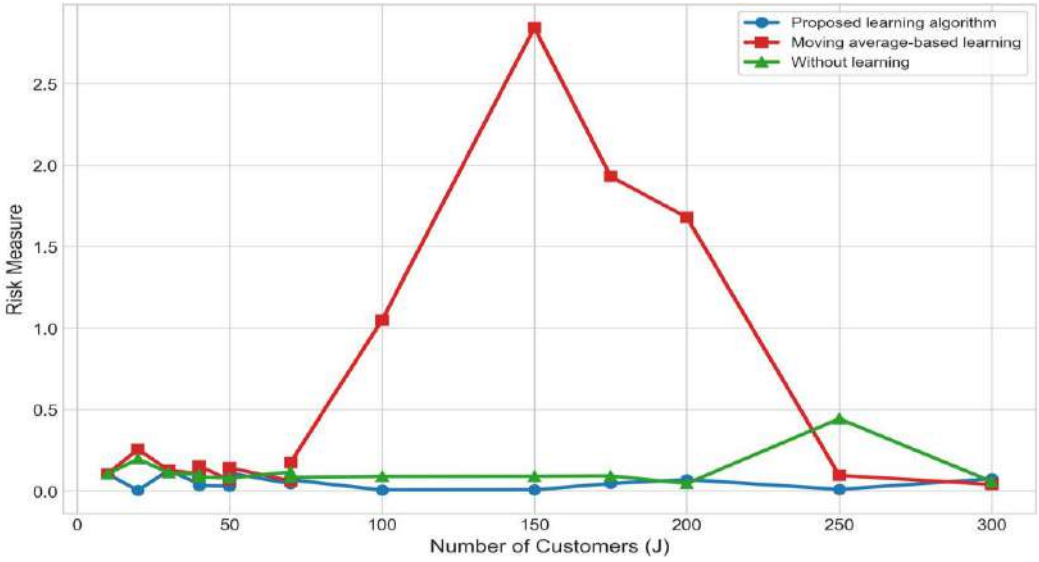



**Figure 12.** Comparative analysis of the learning scenarios based on the obtained risk values.

As illustrated in Figures 13-15, the proposed learning algorithm results in a higher volume

of unmet demands (from the interdictor's perspective) and, consequently, higher operational costs for the LSP. This performance gain stems from the leader's progressively refined knowledge over subsequent rounds; as the leader "learns" more effectively, it identifies more strategic attack points, thereby increasing the volume of interdicted loads and the number of unsatisfied customers.

Furthermore, these sophisticated attacks force the LSP to deviate from primary paths and select more costly alternative routes to bypass interdicted arcs. Consequently, the LSP's total costs under the proposed learning algorithm are higher than those observed in the other two scenarios. While learning-based algorithms typically require a sufficient number of observations to become competitive, they eventually match or exceed the performance of traditional, non-learning methods.

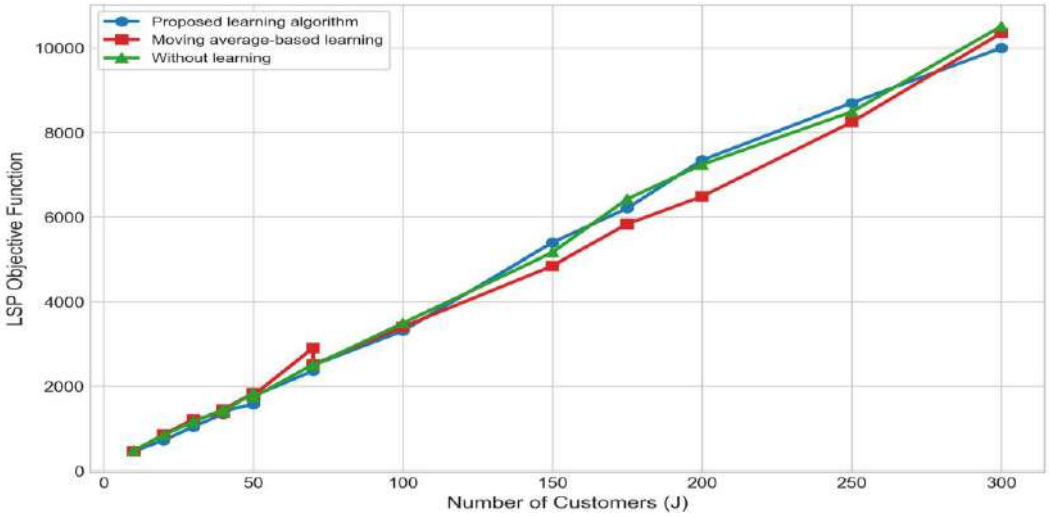



**Figure 13.** Comparison of the learning scenarios based on the LSP objective function.

Notably, the proposed learning algorithm demonstrates superior scalability and adaptability over time compared to the moving-average approach. While established methods, such as that of Bidgoli and Kheirkhah (2018), remain effective in static or well-structured environments, the proposed framework is better suited for dynamic interactions.

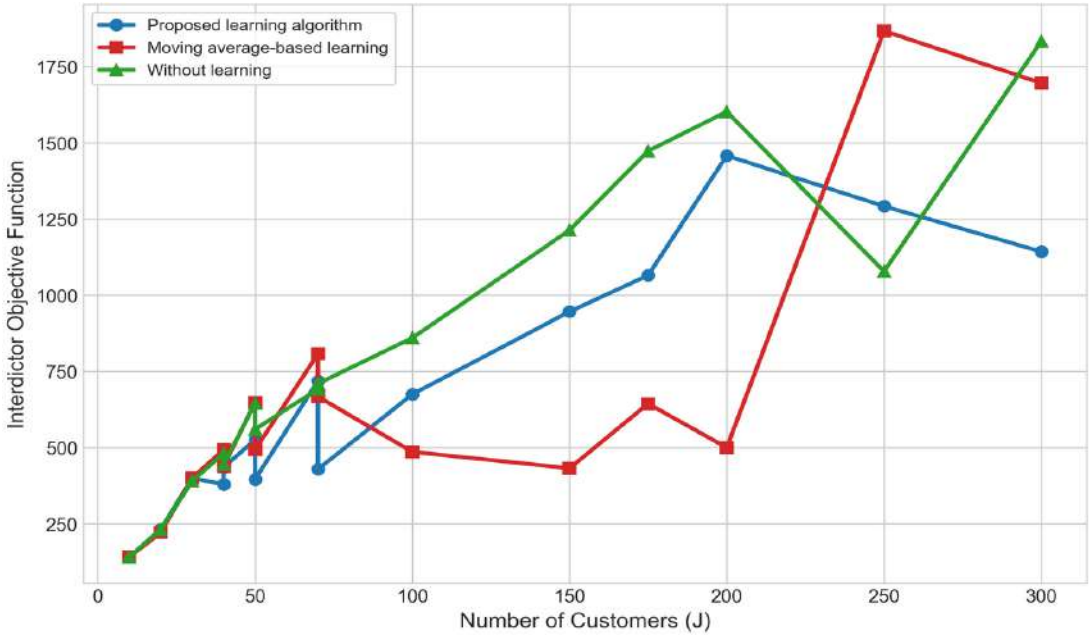



**Figure 14.** Comparison of the learning scenarios based on the interdictor objective function.

Finally, as shown in Figure 15, the computational overhead of the three learning scenarios is comparable; the differences in CPU time are negligible, suggesting that the performance gains of the proposed learning mechanism do not come at a significant computational cost. Although the proposed learning algorithm requires slightly higher computational effort for larger instances, it provides superior solution quality and improved robustness compared with the alternative approaches.

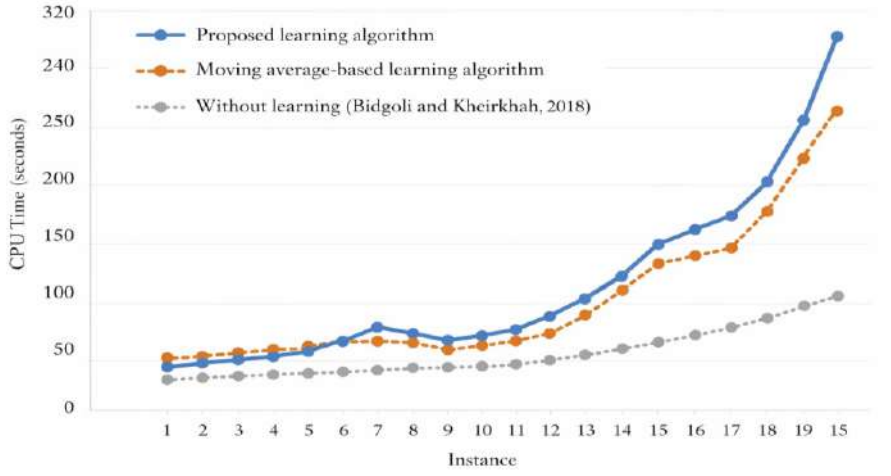



**Figure 15**. Comparison of the learning scenarios based on computational time.

Figure 16 illustrates the progression of the bi-level algorithm under the proposed learning scenario across 20 consecutive periods. In each period, the algorithms at both the upper and lower levels undergo 20 iterations. As shown, the interdictor's objective function consistently improves over time, with the most significant enhancements occurring during the initial periods.

Concurrently, the LSP's reaction to the leader's increasingly strategic interdiction is most pronounced during these early phases. Subsequently, the lower-level genetic algorithm continues to adapt, effectively searching for lower-cost routing solutions for the LSP in every round of the game, thereby responding to the evolving threat posed by the interdictor.

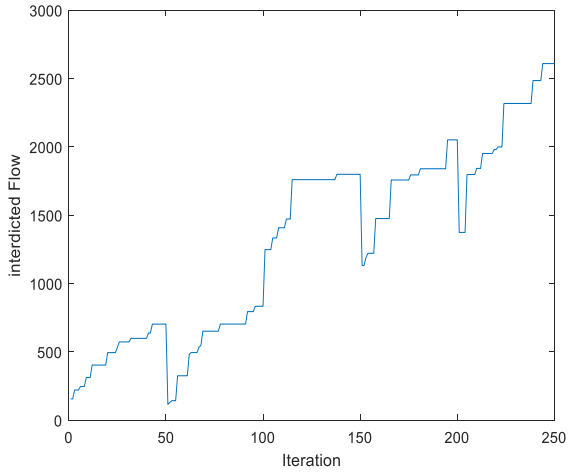



**Figure 16.a**. The upper-level genetic

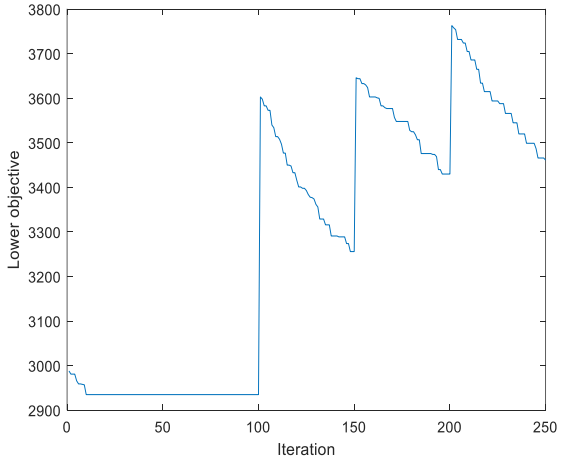



**Figure 16.b**. The lower-level genetic

Figure 16 illustrates the convergence behavior of the bi-level algorithm under the proposed learning scenario across 20 successive periods. In each period, both the upper- and lower-level algorithms perform 20 iterations. As shown in the figure, the interdictor's objective function improves consistently over time, with the most significant gains occurring during the initial

periods. Concurrently, the LSP's reaction to the leader's increasingly strategic interdiction is most pronounced during these early stages. In each round, the lower-level genetic algorithm effectively seeks to identify lower-cost routes for the LSP in response to the interdictor's moves.

Furthermore, an analysis of the results in Table 5 highlights the scalability of the proposed learning algorithm. The problem instances range from 10 to 300 nodes, representing increasingly complex interdiction environments. Despite this substantial growth in problem size, the proposed algorithm maintains consistently low risk values across the larger instances (i.e., 100–300 nodes). By contrast, the moving-average strategy exhibits significant fluctuations and substantially higher risk values as the network grows. This indicates that the proposed learning mechanism remains stable and reliable even as network complexity increases, whereas simpler prediction strategies lose effectiveness. These results underscore the proposed approach's capacity to handle medium- to large-scale network instances, demonstrating promising scalability and robust performance.

### 5.3. Statistical Comparison of Algorithms Based on Risk

To provide rigorous statistical validation of the proposed learning algorithm, we conducted a non-parametric analysis based on the risk metric, which directly quantifies the interdictor's ability to identify arcs with maximum interdicted flows. The Friedman test was applied to compare the performance of the three algorithms, Proposed Learning**,** Moving Average-based Learning, and No Learning, across all 15 instances. The results revealed a statistically significant difference among the algorithms (Friedman test statistic = 10.51, $p$-value = 0.0052), indicating that at least one method outperforms the others in a meaningful way.

To identify specific pairwise differences, a Holm-corrected post-hoc analysis was subsequently performed. The proposed learning algorithm achieves statistically significant reductions in risk compared to both the moving-average learning and the no-learning baselines ($p = 4.5 \times 10^{-5}$ and $p = 0.0079$, respectively). In contrast, no statistically significant difference is observed between the two baseline approaches. These results provide strong empirical evidence that the proposed learning-based method not only achieves lower risk on average but also delivers consistent performance improvements across the evaluated instances.

We focus the statistical analysis on the risk metric because it serves as the most direct indicator of the interdictor's effectiveness. Although other metrics, such as the LSP and interdictor objectives, are also relevant, risk more closely reflects the operational goals of the repeated interdiction game and is therefore the most appropriate measure for rigorous comparative testing. Extending the analysis to incorporate these additional objectives is left for future work.

### 5.4. The Sensitivity Analysis for Time Horizon Length

One of the most important factors affecting learning performance is the length of time periods of demand collected from the past. We analyze this parameter by solving eight instances for three different time periods: 5, 7, and 10 periods. The results are displayed in Table 6 and Figure 17.

$T = 5$ often exhibits larger discrepancies between the predicted and realized interdiction values, indicating lower prediction accuracy. In contrast, $T = 7$ and $T = 10$ show a closer agreement between predicted and actual interdiction outcomes, particularly for the later instances. This pattern suggests that learning accuracy improves as $T$ increases. Consequently, larger values of $T$ yield more accurate interdiction predictions and enable the interdictor to make more effective decisions.

As shown in Figure 17, the risk values decrease markedly as Tincreases. The setting $T = 5$ produces several high-risk cases (e.g., instances 2, 6, and 13), whereas $T = 10$ generally achieves the lowest risk values, indicating more stable and reliable predictions. Overall,

increasing $T$ reduces prediction uncertainty and improves the reliability of the learning process.

**Table 6.** The results of the proposed Learning algorithm for different time horizons

| J_V | $T = 5$ | | | | $T = 7$ | | | | $T = 10$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | LSP | Interdictor objective | | Risk | LSP | Interdictor objective | | Risk | LSP | Interdictor objective | | Risk |
| | | Predicted | Real | | | Predicted | Real | | | Predicted | Real | |
| 10_2 | 462 | 156 | 141 | 0.106 | 457 | 156 | 141 | 0.106 | 459 | 156 | 141 | 0.106 |
| 20_4 | 821 | 249 | 231 | 0.078 | 843 | 214 | 191 | 0.120 | 719 | 233 | 232 | 0.004 |
| 30_5 | 1065 | 321 | 298 | 0.077 | 1091 | 390 | 386 | 0.010 | 989 | 445 | 398 | 0.118 |
| 40_5 | 1671 | 546 | 493 | 0.108 | 1345 | 414 | 434 | 0.046 | 1347 | 396 | 379 | 0.045 |
| 40_6 | 1427 | 413 | 391 | 0.056 | 1453 | 504 | 457 | 0.103 | 1412 | 450 | 435 | 0.034 |
| 50_6 | 1732 | 634 | 566 | 0.120 | 1810 | 664 | 612 | 0.085 | 1572 | 542 | 526 | 0.030 |
| 50_7 | 1776 | 559 | 535 | 0.045 | 1993 | 498 | 475 | 0.048 | 1790 | 501 | 479 | 0.046 |
| 70_6 | 2773 | 804 | 761 | 0.057 | 2283 | 779 | 724 | 0.076 | 2363 | 752 | 719 | 0.046 |
| 70_8 | 2547 | 588 | 557 | 0.056 | 2258 | 536 | 505 | 0.061 | 2495 | 459 | 429 | 0.070 |
| 100_7 | 3483 | 979 | 950 | 0.031 | 3820 | 945 | 922 | 0.025 | 3309 | 680 | 675 | 0.007 |
| 150_7 | 5139 | 1101 | 1068 | 0.031 | 5581 | 1228 | 1187 | 0.035 | 5394 | 954 | 946 | 0.008 |
| 175_8 | 5697 | 1283 | 1175 | 0.092 | 5833 | 1257 | 1159 | 0.085 | 6204 | 1115 | 1065 | 0.047 |
| 200_8 | 7004 | 1089 | 1283 | 0.151 | 6912 | 1339 | 1276 | 0.049 | 7343 | 1561 | 1458 | 0.071 |
| 250_9 | 8744 | 1362 | 1338 | 0.018 | 8508 | 1254 | 1192 | 0.052 | 8697 | 1306 | 1292 | 0.011 |
| 300_9 | 9931 | 1537 | 1494 | 0.029 | 10368 | 1526 | 1426 | 0.07 | 9997 | 1230 | 1143 | 0.076 |
| Highest (least efficient) | | | | Moderate | | | | Lowest (most efficient) | | | | |
| Lower | | | | Improved | | | | Best | | | | |
| Higher (more variability) | | | | Moderate | | | | Lowest (most stable) | | | | |

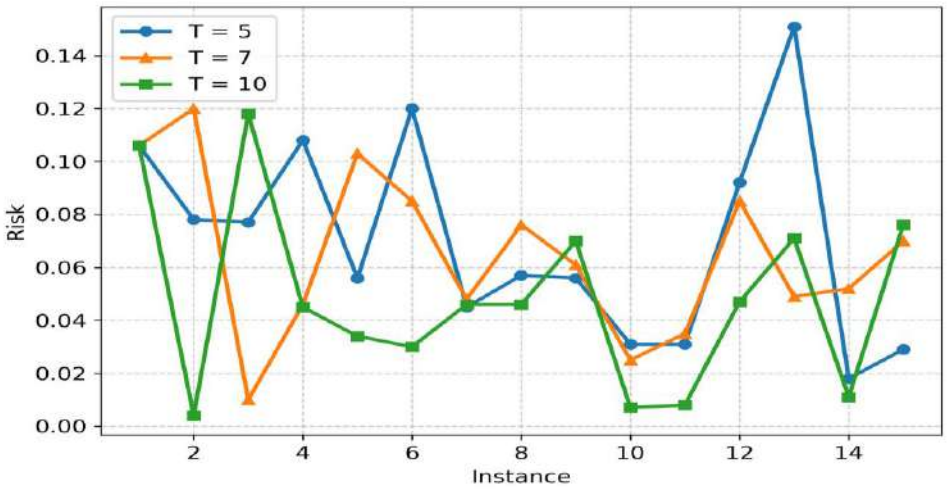



**Figure 17**. The risk for different time horizons (T = 5, 7, and 10).

## 5.5. The effect of interdiction budget on the performance of the bi-level algorithm

To examine whether a direct relationship exists between the interdictor's payoff and the available interdiction budget, we solve multiple instances of different sizes under three budget levels: 160, 300, and 500 dollars.

As shown in Figure 18, increasing the interdiction budget leads to a higher number of unmet demands for the interdictor in the larger network instances. However, this pattern does not consistently hold for smaller networks. In larger networks, allocating a higher budget allows the interdictor to deploy more roadblocks across critical arcs, resulting in more successful interdictions. In contrast, smaller networks offer fewer strategic interdiction opportunities, so additional budget does not necessarily translate into improved interdiction outcomes.

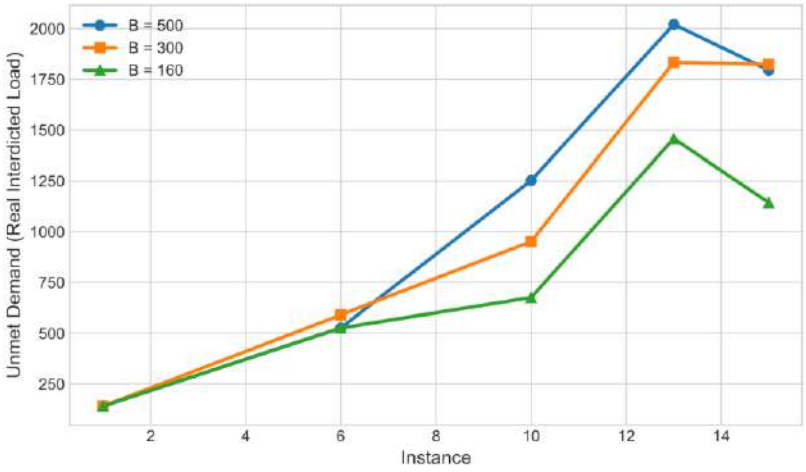



**Figure 18.** The unmet demands for different interdiction budgets.

## 6. Managerial and Practical Insights

From an implementation perspective, the proposed learning-based co-evolutionary framework provides a practical decision-support tool for logistics service providers (LSPs) operating in adversarial environments. The model is particularly relevant for transportation systems in which disruptions, delays, or deliberate interdictions repeatedly affect service performance. In such settings, the ability of the interdictor to update demand estimates over time improves the quality of routing and interdiction decisions without requiring full information in advance.

The computational results indicate that learning can substantially enhance decision quality. In particular, relative to both the non-learning benchmark and the moving-average scenario, the proposed learning scheme reduces the interdictor's risk and increases the amount of unmet demand, while providing a more realistic representation of the interaction between the two players. This suggests that even partial feedback can be sufficient to yield meaningful operational improvements, especially when the system is exposed to repeated interdiction events. The findings also show that the time horizon plays an important role in the performance of the learning process. Longer horizons allow the algorithm to observe more information, update demand estimates more effectively, and reduce the risk associated with inaccurate predictions. Therefore, in practical applications, firms should favor rolling-horizon or multi-period planning frameworks over short-term static decisions, particularly when the network is vulnerable to disruption.

## 7. Conclusions

This study investigated a learning-based NIVRP in which an interdictor seeks to disrupt the operations of a logistics service provider (LSP) under incomplete information about customer demands. To address this challenge, we formulated a repeated bi-level interdiction-routing model and developed a co-evolutionary algorithm that enables the interdictor to gradually learn unknown demand parameters from observed routing and interdiction outcomes.

Beyond the numerical superiority of the proposed method, several managerial and methodological insights emerge from the results. First, the learning mechanism is most beneficial in settings characterized by high demand uncertainty, limited prior information, and repeated interactions between the interdictor and the LSP. In such environments, static or short-sighted interdiction strategies perform poorly, whereas the proposed learning-based framework allows the interdictor to progressively concentrate resources on structurally critical arcs, leading to more effective interdiction outcomes over time. Second, the results reveal a clear trade-off between the interdiction budget and the LSP's operational costs. Increasing the interdiction

budget improves the interdictor's effectiveness but yields diminishing marginal returns beyond a certain threshold, as the LSP adapts by rerouting through less vulnerable paths. This finding underscores the importance of strategic budget allocation over indiscriminate interdiction, suggesting that learning enables regulators to achieve comparable disruption levels at lower enforcement costs.

From a policy perspective, the proposed framework provides a decision-support tool for regulators and enforcement agencies operating with limited information. The results suggest that adaptive, data-driven interdiction policies can outperform rigid enforcement schemes, particularly in large-scale or dynamically evolving logistics networks. The model can be interpreted as a generic adversarial framework, with government agencies serving as a representative real-world application.

Despite these contributions, several limitations should be acknowledged. The learning process assumes that interdiction outcomes provide sufficiently informative signals about customer demands, which may be noisy or incomplete in practice. Moreover, only synthetic instances are considered, and real-world operational constraints, such as delayed observations or strategic misinformation, are not modeled. These limitations open promising avenues for future research. Extensions may include incorporating noisy or partially observable feedback, considering alternative interdiction strategies such as penalty-based enforcement, and integrating advanced learning paradigms such as reinforcement learning or Bayesian updating. Adapting benchmark VRP instances to interdiction settings and studying multi-interdictor or coalition-based scenarios also represent fruitful directions for further investigation.

## References

Bertsimas, D., & Kallus, N. (2020). From predictive to prescriptive analytics. Management Science, 66(3), 1025–1044. https://doi.org/10.1287/mnsc.2018.3253

Bidgoli, M. M., & Kheirkhah, A. (2018). An arc interdiction vehicle routing problem with information asymmetry. Computers & Industrial Engineering, 115, 520–531. https://doi.org/10.1016/j.cie.2017.11.019

Farrokhi-Asl, H., Tavakkoli-Moghaddam, R., Asgarian, B., & Sangari, E. (2017). Metaheuristics for a bi-objective location-routing-problem in waste collection management. Journal of Industrial and Production Engineering, 34(4), 239–252. https://doi.org/10.1080/21681015.2016.1253619

Gendreau, M., Laporte, G., & Séguin, R. (1996). Stochastic vehicle routing. European Journal of Operational Research, 88(1), 3–12. https://doi.org/10.1016/0377-2217(95)00050-X

Guo, Q., An, B., & Tran-Thanh, L. (2017). Playing repeated network interdiction games with semi-bandit feedback. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17) (pp. 3682–3690). https://doi.org/10.24963/ijcai.2017/515

Haskell, W. B., Kar, D., Fang, F., Tambe, M., Cheung, S., & Denicola, E. (2014). Robust protection of fisheries with COmPASS. Proceedings of the AAAI Conference on Artificial Intelligence, 28(2), 2978–2983. https://doi.org/10.1609/aaai.v28i2.19018

Kakade, S. M., Kalai, A. T., & Ligett, K. (2009). Playing games with approximation algorithms. SIAM Journal on Computing, 39(3), 1088–1106. https://doi.org/10.1137/070701704

Kheirkhah, A., Navidi, H., & Bidgoli, M. M. (2016b). An improved Benders decomposition algorithm for an arc interdiction vehicle routing problem. IEEE Transactions on Engineering Management, 63(2), 259–273. https://doi.org/10.1109/TEM.2016.2542849

Kheirkhah, A., Navidi, H., & Messi Bidgoli, M. (2016a). A bi-level network interdiction model for solving the hazmat routing problem. International Journal of Production Research, 54(2), 459–471. https://doi.org/10.1080/00207543.2015.1084061

Legillon, F., Liefooghe, A., & Talbi, E. G. (2012). CoBRA: A cooperative coevolutionary algorithm for bi-level optimization. In 2012 IEEE Congress on Evolutionary Computation (pp. 1–8). IEEE. https://doi.org/10.1109/CEC.2012.6256620

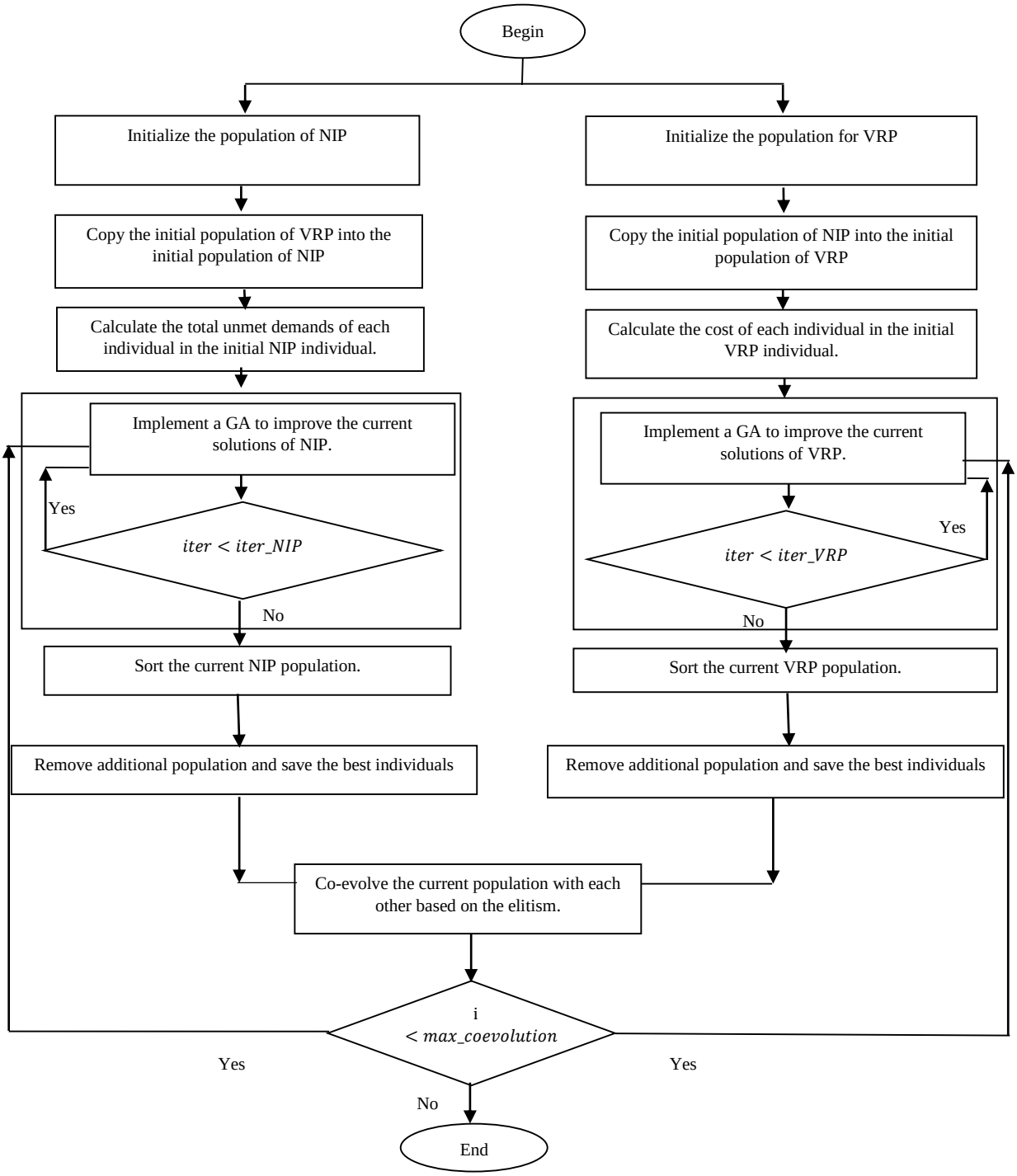
Nadizadeh, A., & Sabzevari Zadeh, A. (2021). A bi-level model and memetic algorithm for arc interdiction location-routing problem. Computational and Applied Mathematics, 40. https://doi.org/10.1007/s40314-021-01453-2

EurekaMag

Sadati, M. E. H., Aksen, D., & Aras, N. (2020a). The r-interdiction selective multi-depot vehicle routing problem. International Transactions in Operational Research, 27(2), 835–866. https://doi.org/10.1111/itor.12669

Wiley Online Library
Sadati, M. E. H., Aksen, D., & Aras, N. (2020b). A trilevel r-interdiction selective multi-depot vehicle routing problem with depot protection. Computers & Operations Research, 123, 104996. https://doi.org/10.1016/j.cor.2020.104996
OpenAlex
Tambe, M. (2011). Security and game theory: Algorithms, deployed systems, lessons learned. Cambridge University Press. https://doi.org/10.1017/CBO9780511973031

## Appendix A:




This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license.