International Journal of Networked and Distributed Computing (2025) 13:4
https://doi.org/10.1007/s44227-024-00048-7

RESEARCH ARTICLE

Deep Learning Based Multi Constraint Hybrid Optimization Algorithm for Transshipment-Based Inventory Routing with Dynamic Demands

Sreeparnesh Sharma Sivadevuni[1] · Naveen J[1]

Received: 5 July 2024 / Accepted: 21 November 2024

Abstract
The Inventory-Routing Problem (IRP) is considered a major issue in supply chain management as it comprises two areas: vehicle routing and inventory control. The existing techniques were unable to incorporate location details for enhancing the decision-making and it failed to consider the uncertainty of the demands. Hence to solve this issue, a Snake Artificial Ecosystem Optimization (SAEO) algorithm is proposed in this paper. The SAEO algorithm is developed to address the transshipment IRP with dynamic demands by combining the AEO model and SO to enhance the optimizer's performance. Further, a penalty strategy is proposed, where Deep Quantum Neural Network (DQNN) is employed for calculating the penalty for verifying the feasibility of the solution generated in case of violations in model constraints. In addition, the efficiency of the proposed SAEO-DQNN technique is examined by considering metrics, like transportation cost, transshipment cost, and total cost, and it achieved improved values of 0.391, 0.518, and 1.012 when compared to existing techniques such as Genetic Algorithm with Deep Reinforcement Learning (GA + Deep RL) and Kernel Search Multi-vehicle IRP (KSMIRP).

Keywords Dynamic demands · Inventory routing · Quantum neural network · Snake artificial ecosystem optimization · Transshipments

1 Introduction

One of the most recent instances of value-added logistics is Vendor-Managed Inventory (VMI), which comprises a well-organized technique for managing inventory where the replenishment assessment is made by the supplier by considering the supply chain policy and the inventory [1]. The main issue faced by VMI is that it needs to solve an extremely challenging optimization issue named IRP, which comprises two commonly researched issues such as vehicle routing and inventory management [2]. Over the past decades, the research community has developed a huge interest in IRPs, owing to the application of IRPs in real-time scenarios like supply chain management and integrated logistics and the problems illustrated in their studies [3]. In addition, researchers in IRPs have been greatly inspired by the possible advantages of integrating routing decisions with inventory management. Determining the solution for two individual optimization issues considering the various factors in routing and inventory management results in an integrated issue with a suboptimal solution [4]. IRP designates a notably significant advancement in Vehicle Routing Problem (VRP) that aims at performing routing decision-making and inventory control simultaneously [5, 6]. The major intent of performing IRP is to identify the vehicle routing strategies or sequence of visiting customers in every period. So that the overall cost is reduced depending on the customer's storing capability of the distributed products without allowing a shortage of inventories [7].

The supply chain in IRP comprises a supplier and numerous traders located at various distributed geographical locations with varied demands [8]. The supplier must ship the products to the vendors/retailers within a specific time frame. Therefore, all customers' demands are satisfied with the help of vehicles in a distinct time frame [9, 10]. The overall cost comprises the routing and holding costs. IRP aims to capitalize on achieving high performance by considering inventory management and vehicle routing in

✉ Sreeparnesh Sharma Sivadevuni
Sreeparnesh.s@res.christuniversity.in

Naveen J
Naveen.j@christuniversity.in

1 Department of Computer Science and Engineering, School of Engineering and Technology, Christ University, Bangalore, Karnataka 560001, India

Published online: 05 December 2024

the supply chain [11]. Under-discussed issues include real-time data's role in customer interaction and demand management, data ownership, sharing, and privacy, technical data and infrastructure security, utility personnel skills, and deployment costs and benefits [12]. Furthermore, the supplier ensures product delivery in the VMI model by identifying the routes, finding the vendor's replenishment strategy, and avoiding any shortages [13]. IRP tries to reduce the total cost of logistics to ensure that the vendor's demands are satisfied during the planning horizon. Transshipments offer an efficient strategy for rectifying the variations in the inventory available and the demand experienced in any location [14]. Thus, transshipments result in minimized costs and enhanced services with no increase in inventories in the system. Conventionally, electronic manufacturers stock their spare parts themselves rather than distributing them in between the manufacturers [15]. In this paper, a novel SAEO-DQNN technique is developed for addressing transshipment IRP with dynamic demands. Here, a supply chain is considered which comprises of a central warehouse, vehicles, and retailers. The IRP is handled using two processes, such as transshipments and substitution. The key contribution of this paper is given below.

- The SAEO algorithm is developed to address the transshipment IRP with dynamic demands by combining the AEO model and SO in order to enhance the performance of the optimizer.
- A penalty strategy is proposed, where DQNN is employed for calculating the penalty for verifying the feasibility of the solution generated in case of violations in model constraints.
- The SAEO-DQNN for inventory routing performance is estimated by transportation cost, transshipment cost and total cost based on the four instances in IRP, such as high cost_H3, low cost_H3, high cost_H6, and low cost_H6.

The rest of the paper is structured in the following order: Sect. 2 elaborates on the various prevailing works in IRP, the devised SAEO-DQNN approach is elaborated in Sect. 3, followed by the experimental outcomes and assessment in Sects. 4, and 5 concludes the work.

2 Literature Review

Various works have addressed IRP in the past, among which seven works are considered here for analysis.

Achamrah et al. [16] presented a GA and deep RL model for solving IRP. A threshold was utilized for facilitating a re-ordering policy of pre-determined dimensions to restock the inventory. Further, GA and Deep RL were integrated to minimize the shipping cost, with operations like mutations and crossover. This method effectively minimized the computational cost but failed to address the production rate to obtain superior efficacy. Archetti et al. [17] devised a KSMIRP for minimizing the overall distribution cost. This approach was based on the Kernel Search (KS) technique, in which the input variables were split into numerous kernels (subsets) to solve the MIP issue. Here, the TS algorithm was used to identify the parameters of the restricted issue. The Multi-Vehicle IRP (MIRP) issue was addressed by formulating it as a Mixed-Integer Linear Programming (MILP) for building the MIP problem sequence. The KSMIRP successively provided a trade-off between computational complexity and solution quality. However, it did not consider utilizing effective techniques to address the routing issues. Alinaghian et al. [18] presented an Augmented TS Algorithm for addressing IRP. Here, a MILP scheme was devised to reduce fuel utilization in the Green Inventory-Routing Problem with Time Windows (GIRP-TW). The problem was addressed with the help of three algorithms: Differential Evolution (DE), original TS, and augmented TS. Furthermore, an improved Push-Forward Insertion Heuristic (PFIH) algorithm was employed for handling the routing structure. The speed optimization and route design were carried out using the heuristic speed optimization algorithm. An improved Clarke–Wright algorithm was used to construct the vehicle routing strategy devoid of the time window. This scheme successfully minimized usage, inventory, driver, and consumption costs, although it obtained different results in every execution due to the random operators.

Liu et al. [19] devised an integrated location-inventory-routing model for minimizing economic costs in transporting perishable products. Various parameters, like freshness, carbon emission levels, and economic cost, were initially evaluated. Then, constraints were recognized in the basis of the location-inventory-routing scenario, and a multi-objective planning scheme was designed. Finally, the optimal solution was computed using the YALMIP toolbox. This technique effectively reduced the carbon emission levels and manufacturing costs; however, it failed to explore the transportation period. Gruler et al. [20] presented a Variable Neighborhood Search (VNS) technique for solving multiperiod IRP considering the stochastic demands of the consumers. This approach was devised by integrating the VNS method with Monte Carlo Simulation (MCS) for constructing the initial solutions. Further, a local search algorithm was employed to explore the neighborhood of the solution, and refinement was performed to determine the best solutions. Although this approach effectively addressed the complexities of the issues concerning the relationship among the successive periods owing to the arbitrary demands, it failed to consider variables, such as distribution costs.

Friske et al. [21] introduced a Relax-and-Fix (R&F) algorithm and Fix-and-Optimize (F&O) algorithm for maritime

IRP. Here, the overall count of the constraints and variables were minimized with the help of simplification and pre-processing. The initial solution was computed by the R&F algorithm, which was later enhanced by the F&O algorithm. The approach effectively reduced the search space and dimension of the system; however, it suffered from the dependency of algorithmic parameters on instances. Mahjoob et al. [22] presented a Modified Adaptive GA (MAGA) to find the solution to various instances. Here, a multi-product multi-period IRP (MMIRP) was formulated with a heterogeneous vehicle set and customers and suppliers dispersed in a geographical area. The MAGA was implemented in two stages, construction and improvement. Here, the GA was modified using the Lin-Kernigan Helgaun heuristic for obtaining the sub-optimal or optimal Travelling Salesman Problem (TSP) tour. This technique attained higher closeness rate, but did not consider the uncertainties of consumer demands.

Not being able to apply non-parametric procedures that assess customer demand based on the empirical demand distribution rather than utilizing the data directly is one of the limitations of the methods that are currently in use. It is not possible to include geographical information to improve decision-making capacity [23]. In addition to this, it does not consider the inherent unpredictability of the need. I failed to consider the need to use an algorithm selection approach to ascertain the ideal parameters for each instance of the challenge. The disregard for the unmet and ever-changing expectations of the customer, in addition to the degradation of the items while they are being transported about[8].

3 Introduced SAEO-DQNN for Transshipments-Based IRP

The devised SAEO-DQNN approach considers a supply chain comprising a central warehouse with multiple products that serve many retailers with or without transshipment. Moreover, the retailers may be customers or other agents with a specific quantity of products that must be replenished to alleviate the product shortage. Initially, the products are collected in vehicles from the manufacturer/other locations and stored in the central warehouse, which are later transshipped to the retailers for replenishing the products at the retailers. This problem can be modelled as an IRP with dynamic demands and can be considered as an optimization problem with the objective of minimizing the cost of inventory holding lost sales, and transshipment using multiple constraints. Hence to solve this issue, a novel multi constraint-based SAEO algorithm is devised, which is developed by merging the AEO model and the SO. Further, a constraint violation policy is designed based on the violation penalty, which is determined with the help of the DQNN. Figure 1 depicts the schematic representation of the introduced SAEO-DQNN technique for transshipment IRP.

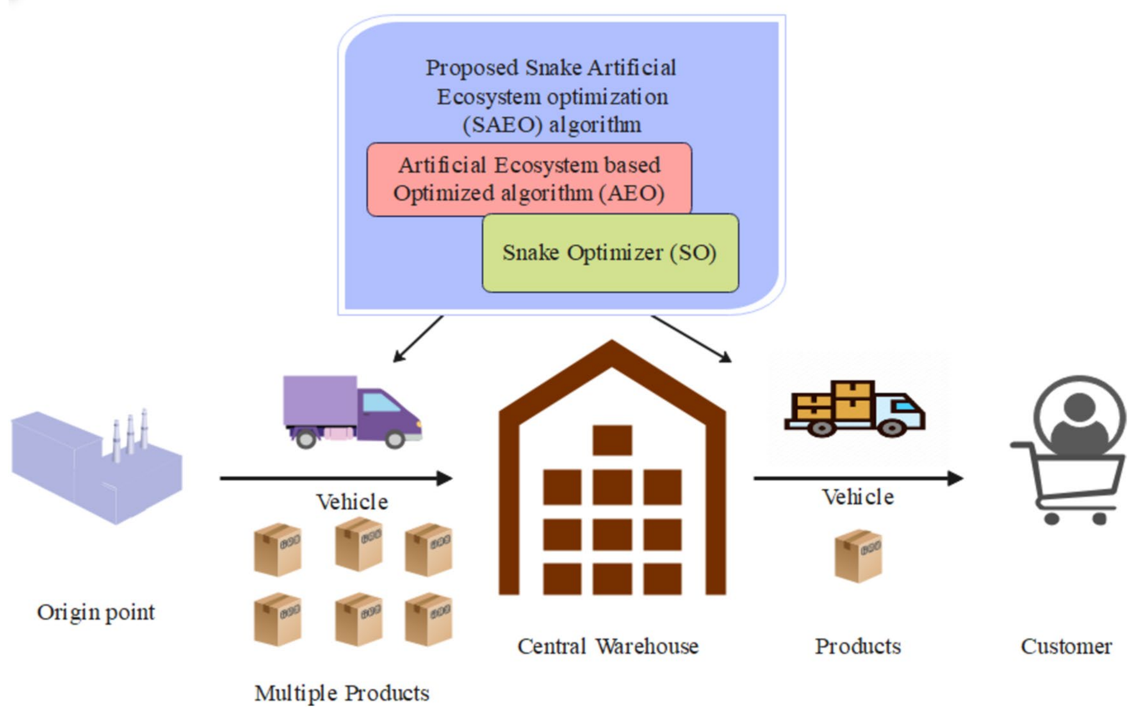

Fig. 1 Schematic representation of SAEO-DQNN technique for transshipment IRP

3.1 Data Description

The IRP is considered to an NP-hard problem owing to the vehicle routing problem and the developed SAEO-DQNN approach is evaluated by considering four different classes of instances considering high cost and low cost for two different planning horizons.

3.2 Inventory Routing Optimization Model

The devised inventory routing model is elaborated in this section, wherein the optimization of the routing process is carried out with the help of the developed SAEO algorithm. Further, the multi-constraint objective function used to reduce the overall cost considering the inventory holding cost, transshipment cost, lost sales cost, and transportation cost is also presented. Moreover, a detailed explanation of the devised SAEO algorithm is also elucidated.

3.2.1 Solution Encoding

Inventory routing permits the supplier to ship goods to various retailers by optimizing vehicle routing and inventory management. Here, the IRP problem aims at determining the optimal routing to deliver products to various retailers, with minimum cost and minimum number of vehicles. The solution is to determine the order in which the products have to be delivered from the central warehouse or the supplier to the retailers through vehicles. The solution vector has a dimension of $1 \times (e \times h)$, where e denotes the number of retailers, and h represents the total number of vehicles used.

3.2.2 Multi-Constraint Objective Formulation

The IRP is addressed by considering a multi-constraint objective function to reduce the overall cost considering the inventory holding cost, transshipment cost, lost sales cost, and transportation cost along with the penalty. The multi-constraint objective function is given by Eq. (1),

$$TC = \sum_{j=1}^{e} \left(\frac{C_{hd_{sj}} + I_{zjt} + C_{trans} + C_{trship}}{4} \right) \lambda \tag{1}$$

where, $C_{hd_{sj}}$ indicates the inventory holding cost, I_{zjt} refer the lost sales cost, C_{trans} denotes the transportation cost, C_{trship} refer to the transshipment cost, and λis the penalty, which are briefed in the following section.

(i) Inventory holding cost $\boldsymbol{C_{hd_{sj}}}$

This cost designates the cost incurred for the storage of products at the retailer, and is given in Eq. (2),

$$C_{hd_{sj}} = \sum_{j=1}^{e} hd_{sj} \tag{2}$$

where, hd_{sj} designates the inventory holding cost for the j th retailer, and is expressed in Eq. (3),

$$hd_{sj} = inventoryholdingat\,retailer\,j\,attimet \times C_{uirj} \tag{3}$$

where, C_{uirj} indicates the unit inventory cost.

(ii) Last sales amount $\boldsymbol{I_{zjt}}$

The lost sales cost is linked to product shortage and computed by finding the difference between expected sales and actual sales. It is expressed in Eq. (4),

$$I_{zjt} = Ex_j - Ac_j \tag{4}$$

where, Ex_j represents the expected sales, and Ac_j designates the actual sales.

(iii) Transportation cost $\boldsymbol{C_{trans}}$

Transportation cost is related to the fuel consumption of the vehicles and is based on the distance between the subsequent retailer inventories.

(iv) Transshipment cost $\boldsymbol{C_{trship}}$

Transshipment is employed to reduce the overall shipping cost and is computed with the help of Eq. (5),

$$C_{trship} = \frac{\sum_{i=1}^{h} d_s}{G_{trans}} \tag{5}$$

where, d_s represents the s th vehicle and G_{trans} denotes the transportation capacity.

(v) Penalty calculation using DQNN

The penalty violation policy is imposed if the supplier fails to deliver the product on time, thus violating a time window constraint. For example, consider that a retailer places a demand for products that have to be delivered in a specific duration; if it is violated shortage of product occurs. Thus, to improve the constraints, a penalty violation policy is considered, and a penalty is imposed when the policy is violated. The penalty λ is determined using the DQNN, considering inventory holding, transshipment, lost sales, and transportation costs.

DQNN Architecture:

DQNN is a generalization of the Deep Neural Network (DNN) that is used to effectively solve numerous complex issues. DQNN was created by fusing the principles of DNN with quantum computing. Like the DNN, the DQNN comprises input, hidden, and output layers, where the quantum perceptron and its analog form the basic building block. The quantum perceptron can perform quantum computations and random unitary operators fed with k input and l output qubits, thus they have $k + l$ qubit unitarizes. The perceptron depends on $(2^{k+l})^2 - 1$ factors comprising of the weight and biases. Initially, the input and output qubits are set with an indefinite state λ and fiducial multiplicative state $|0\ldots0\rangle_{out}$. Further, the DQNN is considered to have E hidden layers which contain a circuit with quantum perceptron's, which act on the input state λ_{in} and generate the output state λ_{out} given in Eq. (6),

$$\lambda_{out} = tr_{in,out}\left(H\left(\lambda_{in} \otimes |0\ldots0\rangle_{hid,out}\langle 0\ldots0|\rangle\right)H^{+}\right) \tag{6}$$

where, λ_{in} denotes input state, λ_{out} denotes output state, $|0\ldots0\rangle_{hid,out}$ denotes fiducial multiplicative state and H^{+} denotes quantum circuit. The equation gives the penalty computed by the DQNN. In Fig. 2, the architecture of DQNN is displayed.

3.2.3 Devised SAEO Algorithm for Optimizing IR

This section describes the developed SAEO algorithm for optimizing the inventory routing process. The AEO model is modified using the SO to obtain the devised SAEO algorithm employed to obtain the optimized inventory routing. This algorithm imitates the three distinct behaviors of living things comprising production, consumption, and decomposition. An ecosystem refers to a group of organisms living in a certain area and having a mutual relationship with one another. The AO algorithm offers the benefit of easy implementation and is effective in handling real-time issues with a high converging speed. However, the complexity of the problem limits the computational efficiency of the AEO algorithm. The SO is devised from the inspiration of the mating behavior of snakes. The SO assumes that the male and female snakes are equal in number during the updating process. In the exploration phase, the snakes search for food ad in the exploitation phase, in case of ambient temperature, the snakes either fight. Thus, by combining the two algorithms, the devised SAEO algorithm achieves a fast convergence rate with minimum computational complexity. The steps of the developed SAEO algorithm are listed as follows.

Step 1: Initialization

An ecosystem V comprising m individuals is initialized at the start, wherein each individual corresponds to potential solutions, and it can be expressed in Eq. (7),

$$V = \left\{V_1, V_2, \ldots, V_i, \ldots, V_m\right\} \tag{7}$$

where, V_i symbolizes the i th individual.

Step 2: Fitness Computation

Once the population is initialized, the fitness of each individual is computed with the help of Eq. (1), wherein the fitness corresponds to the total cost. The individuals are arranged in descending order based on the fitness, and the individual with highest fitness is considered the worst individual or the producer (V_1). On the other hand, the lowest fitness corresponds to the decomposer and is considered the best individual (V_m).

Step 3: Production

The producer normally creates energy in an ecosystem utilizing the nutrients given by the decomposer, sunlight, water and carbon dioxide. This behavior is considered for updating the producer based on the decomposer, as well as the bounding limits of the search space and the updated producer directs the omnivores and herbivores to various areas. This enables the algorithm to generate a new organism randomly substituting the prior one between the decomposer (V_m) and an arbitrarily individual (V_{rand}) in the search space. This producer operator is be modelled in Eq. (8),

$$V_1(x+1) = (1-\eta)V_m(x) + \eta V_{rand}(x) \tag{8}$$

where, x indicate the current iteration, η is a weight coefficient, V_m and V_{rand} are a decomposer and arbitrarily individual, V_1 is an arbitrary number with value in the range [0,1].

Step 4: Consumption

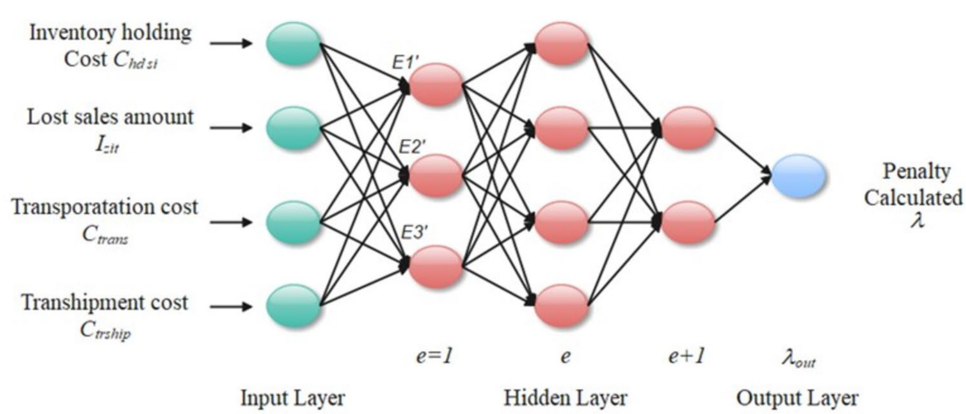

Fig. 2 Architecture of DQNN

Once the producer operation is accomplished, the process of consumption is carried out by the consumers using the consumption operator. The consumers gather energy by eating a producer or another consumer with low fitness or both. The process of searching food is established by using Levy Flight, which also enhances the exploration capability. However, the behavior modelling using Levy Flight suffers from the need of tuning numerous parameters and high complexity. So, to overcome this, a consumption factor is introduced, which is a random walk devoid of any parameter with the Levy Flight feature. The consumption factor helps the consumers in acquiring food based on various hunting techniques.

Herbivore:

When the consumer chosen arbitrarily is an herbivore, its consumer is only the producer. This behavior can be represented in Eq. (9),

$$V_i(x+1) = V_i(x) + P.\left(V_i(x) - V_1(x)\right), i \in [2, \ldots, m] \tag{9}$$

Carnivore:

In the case of a carnivore being selected randomly, it feeds on other consumers with lower fitness values and this is modeled using Eq. (10),

$$V_i(x+1) = \frac{V_{food}(1 + p3 \times Temp \times rand)(1 + p) - P.V_0(x)(p3 \times temp \times rand)}{(p3 \times temp \times rand) + (1 + p)} \tag{10}$$

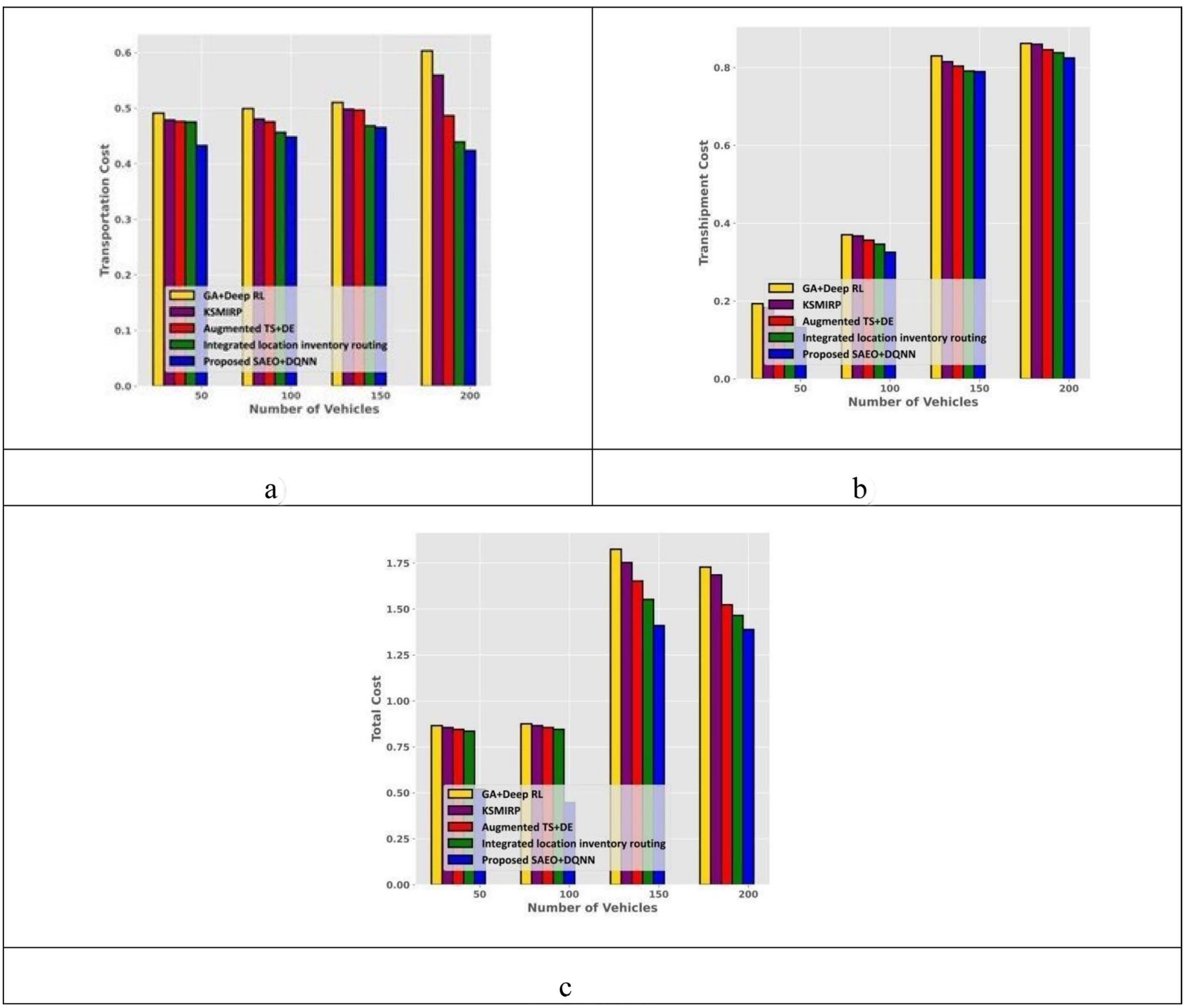

Fig. 3 Assessment of the presented SAEO-DQNN for inventory routing based on high cost_H3 using **a** Transportation cost, **b** Transshipment cost, and **c** Total cost

where, $V_i(x + 1)$ is a i th solution position in the $(x + 1)$ th iteration.

Omnivore:

While a consumer selected arbitrarily is an omnivore, it can feed on the producer or any other consumer with less fitness value, and this is expressed in Eq. (11),

$$V_i(x + 1) = V_i(x) + P.\left(v2 \bullet \left(V_i(x) - V_1(x)\right)\right) + (1 - v2) \bullet \left(V_i(x) - V_0(x)\right) i \in [3, \ldots, m] \tag{11}$$

where, arbitrary number with value in between 0 and 1.

Step 5: Decomposition

The decomposers contribute a major part in the ecosystem, as they decompose the dead organisms into nutrients that are utilized by the producer. This process can be formulated by considering two weight variables y and w, decomposition factor W. The location of the individuals is updated based on the decomposer V_m, which is given in Eq. (12),

$$V_i(x + 1) = V_m(x) + W.\left(yV_m(x) - wV_i(x)\right), i = 1, \ldots, m \tag{12}$$

where, V_i is an arbitrary number in the range [0, 1].

Step 6: Feasibility Evaluation

The optimal solution corresponds to the solution with the minimum fitness, and once the individual position is updated, the fitness is recomputed and if a solution is found to have a high fitness, then its position is updated using Eq. (12).

Step 7: Termination

The above process is kept reiterated till the maximum number of iterations is reached. The best solution obtained V_{best} correspond to the optimized inventory routing and thus, the presented SAEO algorithm is highly efficient in attaining the optimal solution with a fast convergence and minimum computational burden. The pseudocode of SAEO algorithm is given as follows:

Figa.

4 Results and Discussion

This section elucidates the experimental outcomes of the devised SAEO-DQNN technique for transshipment IRP. Further, the experimental set-up, dataset, evaluation metrics, and the analysis of the devised SAEO-DQNN technique are elaborated.

4.1 Experimental Setup

The implementation of the devised SAEO-DQNN approach is executed on a PC having Intel core i-3 processor, 2 GB RAM, and Windows 10 OS using Python tool.

4.2 Evaluation Measures

The developed SAEO-DQNN is analyzed for its efficiency based on metrics, such as transportation cost, transshipment cost and total cost, which are already elaborated in Sect. 3.2.2.

4.3 Comparative Assessment

This section portrays the comparative assessment of the presented SAEO-DQNN for inventory routing considering various performance metrics, based on the four instances in IRP, such as high cost_H3, low cost_H3, high cost_H6, and low cost_H6., considering different numbers of vehicles.

(a) Analysis using high cost_H3

Figure 3 depicts the analysis of the presented SAEO-DQNN using high cost_H3, with varying vehicle numbers. Figure 3a displays the analysis of the devised SAEO-DQNN based on transportation cost. With 50 vehicles, the various inventory routing techniques, such as GA + Deep RL, KSMIRP, Augmented TS + DE, Integrated location inventory routing, and the devised SAEO-DQNN technique, computed transportation costs of 0.491, 0.479, 0.476, 0.475 and 0.433, respectively. In Fig. 3b, the evaluation of the inventory routing schemes considering transshipment cost is portrayed. For 150 vehicles, the value of the transshipment cost achieved by the different inventory routing methods, like GA + Deep RL is 0.830, KSMIRP is 0.815, Augmented TS + DE is 0.804, Integrated location inventory routing is 0.791, and the devised SAEO-DQNN approach is 0.789. The assessment of the developed SAEO-DQNN for inventory routing based on the total cost is displayed in Fig. 3c. The total cost calculated by the various techniques is 0.875 for GA + Deep RL, 0.865 for KSMIRP, 0.855 for Augmented TS + DE, and 0.845 for Integrated location inventory routing. In contrast, the presented SAEO-DQNN achieved a low value of total cost at 0.447 with 100 vehicles.

(b) Analysis using high cost_H6

This section elucidates the comparative analysis of the presented SAEO-DQNN scheme for inventory routing using high cost_H6, which is portrayed in Fig. 4. In Fig. 4a, the examination of the proposed SAEO-DQNN concerning the transportation cost is shown. The transportation cost measured with 100 vehicles by the inventory routing techniques is 0.538 for GA + Deep RL, 0.515 for KSMIRP, 0.487 for Augmented TS + DE, and 0.465 for Integrated location inventory routing, and 0.460 for the introduced SAEO-DQNN technique. Figure 4b exhibits the examination of the developed SAEO-DQNN concerning transshipment cost. For 50 vehicles, the value of the transshipment cost attained by the various inventory routing schemes, like GA + Deep RL, KSMIRP, Augmented TS + DE, Integrated location inventory routing, and the devised SAEO-DQNN is

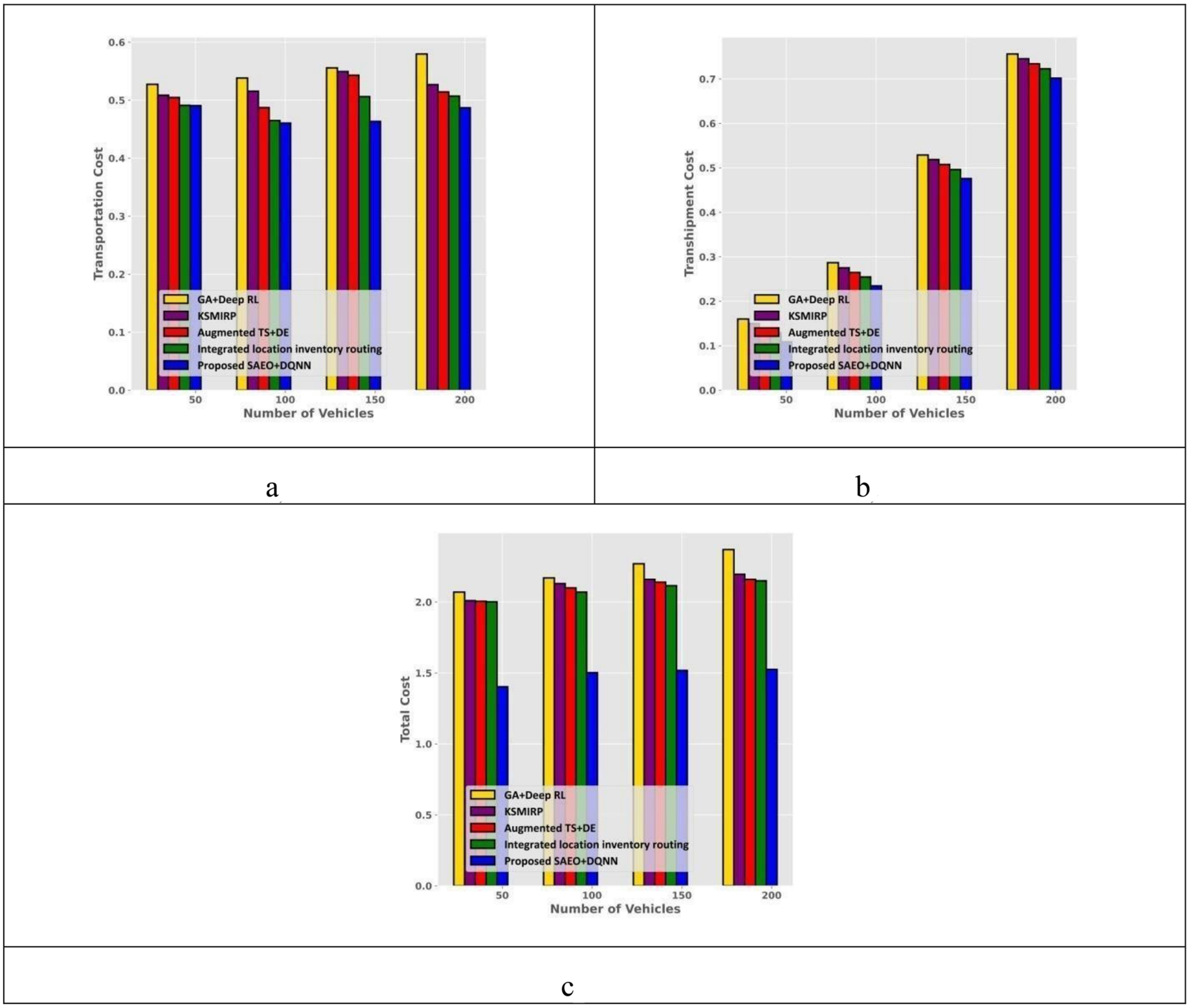

Fig. 4 Assessment of the developed SAEO-DQNN for inventory routing considering high cost_H6 using **a** Transportation cost, **b** Transshipment cost, and **c** Total cost

0.160, 0.150, 0.139, 0.129 and 0.109, correspondingly. Figure 4c presents the analysis of the proposed SAEO-DQNN based on the total cost by varying vehicle numbers. The different inventory routing schemes computed the total cost of 2.270 for GA + Deep RL, 2.159 for KSMIRP, 2.139 for Augmented TS + DE, 2.115 for Integrated location inventory routing, and 1.517 for the developed SAEO-DQNN, with 150 vehicles.

(c) Analysis using low cost_H3

Taking into consideration the total number of vehicles, this part investigates the SAEO-DQNN algorithm for inventory routing with low cost_H3. The findings of the evaluation are shown in Fig. 5, which also includes the assessment. A representation of the evaluation of the SAEO-DQNN based on the expenses of transportation is shown in Fig. 5a. If 150 vehicles are taken into consideration, the transportation cost of the SAEO-DQNN that has been developed is 0.485. When compared to the costs of 0.534, 0.532, 0.500, and 0.498 that were reached by other approaches, such as GA + Deep RL, KSMIRP, Augmented TS + DE, and Integrated location inventory routing, respectively, this cost is lower. Figure 5b is a graphical representation of the assessment of the SAEO-DQNN system based on the expenses of transshipment. The cost of transshipment is decided by several different inventory routing schemes, including GA + Deep RL, which has a value of 0.870, KSMIRP, which has a value of 0.859, Augmented TS + DE, which has a value of 0.844, integrated location inventory routing, which has a value of 0.832, and the one that was designed. Figure 5c depicts the evaluation that considers the whole cost of the

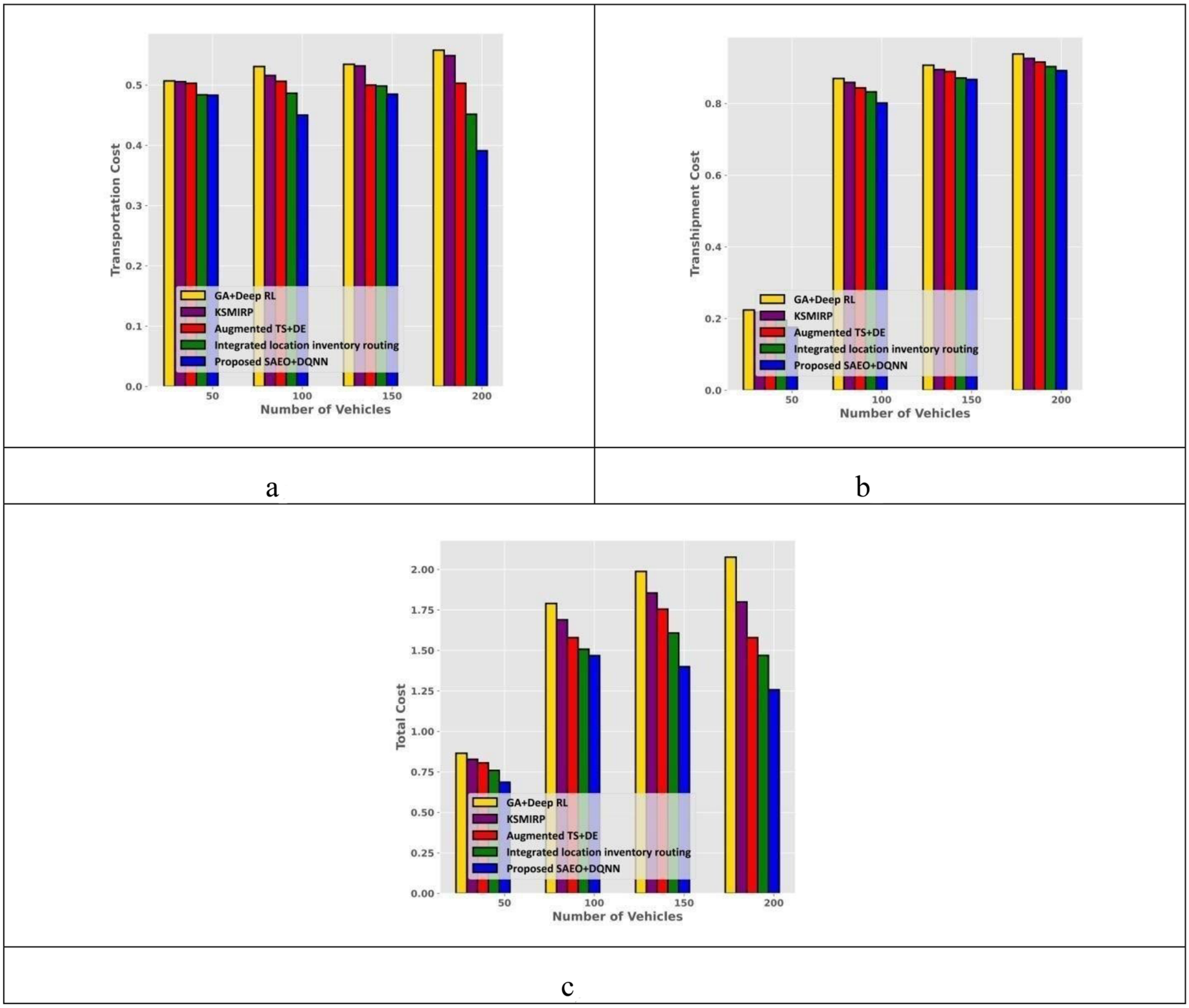

Fig. 5 Assessment of the devised SAEO-DQNN for inventory routing concerning low cost_H3 based on **a** Transportation cost, **b** Transshipment cost, and **c** Total cost

project. The total expenses that relate to fifty Vehicles are 0.865, 0.828, 0.806, 0.759, and 0.686, respectively, according to several inventory routing techniques. These strategies include GA + Deep RL, KSMIRP, Augmented TS + DE, Integrated location inventory routing, and the SAEO-DQNN that was constructed.

(d) Analysis using low cost_H6

Figure 6 depicts the comparative evaluation of the SAEO-DQNN using numerous assessment criteria. The low cost_H6 is altered by modifying the number of cars, as shown in the figure. Figure 6a depicts an analysis of the SAEO-DQNN that was produced in terms of the expenses associated with transportation. There are many distinct inventory routing systems, and the estimated transportation costs for each of them are as follows: 0.520 for GA + Deep RL, 0.495 for KSMIRP, 0.493 for Augmented TS + DE, and 0.482 for Integrated Location Inventory Routing with 100 vehicles. An extremely low transit cost of 0.477 was attained by the SAEO-DQNN that was designed. A representation of the evaluation of the developed SAEO-DQNN about the cost of transshipment is shown in Fig. 6b. Five different inventory routing procedures achieved transshipment costs of 0.558, 0.546, 0.524, 0.519, and 0.507 for a total of 150 automobiles. These prices are equal to GA + Deep RL, KSMIRP, improved TS + DE, Integrated Location Inventory Routing, and the proposed SAEO-DQNN techniques. Figure 6c depicts the findings of the research conducted on the SAEO-DQNN that was produced about the total cost. However, the other methods provide higher total costs: 0.906 for GA + Deep RL, 0.886 for KSMIRP, 0.856 for Augmented

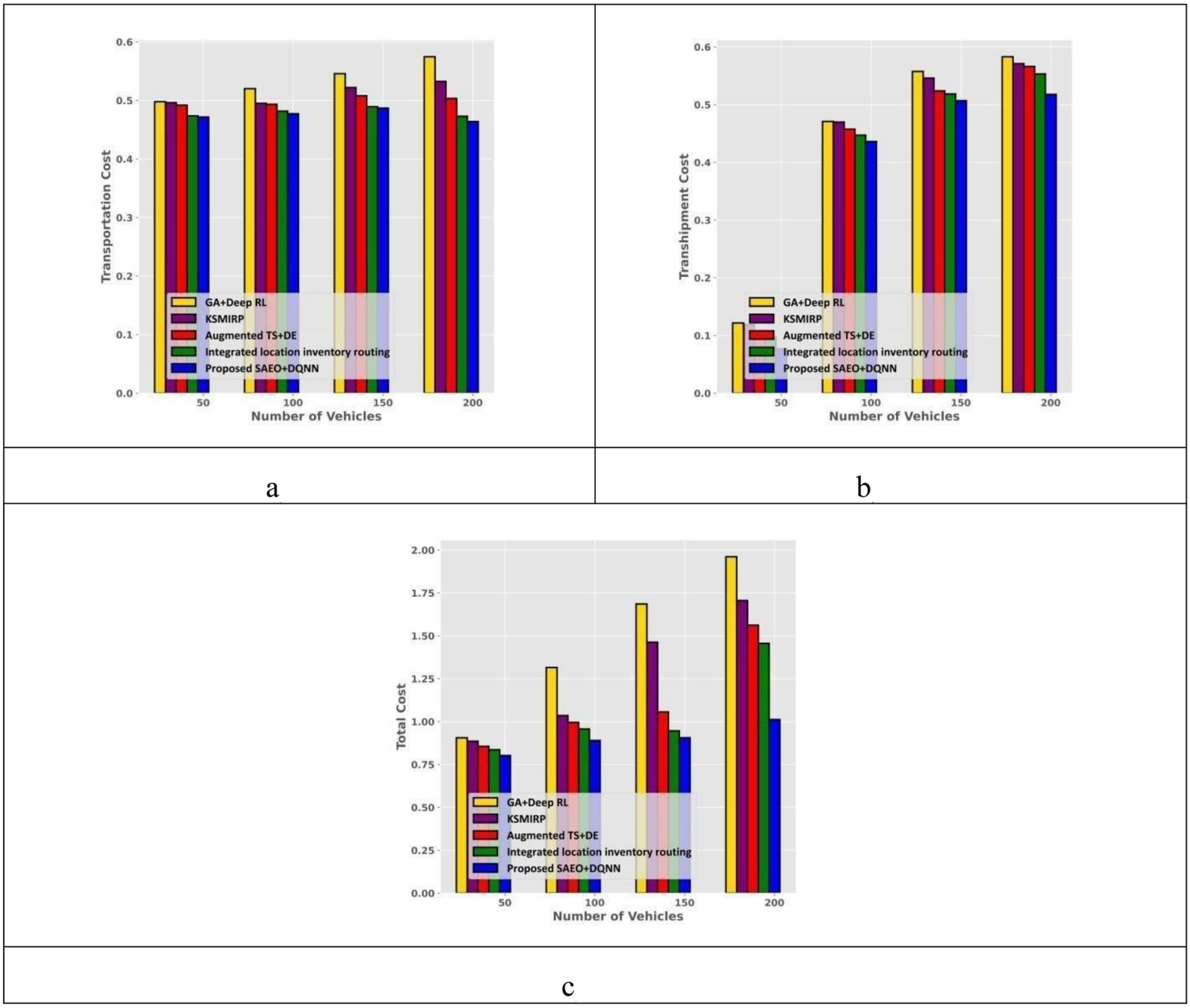

Fig. 6 Analysis of the presented SAEO-DQNN for inventory routing based on low cosy_H6 using **a** Transportation cost, **b** Transshipment cost, and **c** Total cost

TS + DE, and 0.836 for Integrated location inventory routing. The overall cost calculated by the SAEO-DQNN with 50 automobiles is 0.802, whereas the other methods produce higher total costs.

4.4 Comparative Analysis

This section aims to examine the comparative analysis of the SAEO-DQNN proposed for inventory routing. The efficacy of the devised routing system is assessed by comparing it with other inventory routing methodologies that use various criteria, such as transportation cost, transshipment cost, and overall cost. This is achieved by using four instances: High cost_H3, High cost_H6, Low cost_H3, and Low cost_H6, while altering the number of cars. Table 4 presents the comparative analysis of the described SAEO-DQNN approach. The values listed in the table correspond to the total number of vehicles at 200. The efficacy of the presented SAEO-DQNN for inventory routing is investigated by comparing it with other inventory routing techniques, like GA + Deep RL [16, 20], KSMIRP [17], Augmented TS + DE [16], and Integrated location inventory routing [19]. Table 1 presents the comparative analysis.

4.5 Dataset Description

The analysis of the presented SAEO-DQNN for inventory routing is executed using the Inventory-Routing dataset [24].

Table 1 Comparative Analysis

Instances	Metrics	GA + Deep RL [16]	KSMIRP [17]	Augmented TS + DE [18]	Integrated location inventory routing [19]	Proposed SAEO + DQNN
High cost_H3	Transportation Cost	0.603	0.559	0.487	0.439	0.424
	Transshipment Cost	0.862	0.860	0.846	0.838	0.825
	Total Cost	1.728	1.685	1.523	1.465	1.388
High cost_H6	Transportation Cost	0.580	0.527	0.514	0.507	0.487
	Transshipment Cost	0.756	0.745	0.734	0.723	0.702
	Total Cost	2.370	2.195	2.160	2.149	1.525
Low cost_H3	Transportation Cost	0.558	0.549	0.503	0.451	0.391
	Transshipment Cost	0.938	0.926	0.916	0.903	0.892
	Total Cost	2.076	1.799	1.579	1.469	1.257
Low cost_H6	Transportation Cost	0.575	0.532	0.503	0.473	0.464
	Transshipment Cost	0.583	0.571	0.566	0.553	**0.518**
	Total Cost	1.961	1.706	1.562	1.456	**1.012**

The dataset comprises four instance classes based on the cost and length of the planning horizon, such as high cost (horizon, H = 3), low cost (horizon, H = 3), high cost (horizon, H = 6), and low cost (horizon, H = 6). Here, low cost corresponds to the retailer's inventory cost arbitrarily produced in the interval (0.01, 0.05) and the supplier's inventory cost having a value of 0.03. On the other hand, high cost corresponds to the retailers. inventory cost arbitrarily produced in the interval (0.1, 0.5), and the supplier's inventory cost has a value of 0.3. The dataset contains data and solution files. The data file is available in the (*.dat) format, which constitutes information regarding the retailer count, discrete time instant count of the planning time horizon, transportation capacity, production facility, and retailer information.

4.6 Discussion

The problems in existing techniques and advantages of the proposed method is discussed in this section. The GA + Deep RL [16] failed to address the production rate to obtain superior efficacy. The KSMIRP [17] unable to consider utilizing effective techniques to address the routing issues. The Augmented TS + DE [18] unable to incorporate location details for enhancing the decision-making. Moreover, it failed to consider the uncertainty of the demands. The Integrated location inventory routing [19] it failed to explore the transportation period. The SAEO algorithm is proposed for addressing the transshipment IRP with dynamic demands by combining the AEO model and SO in order to enhance the performance of the optimizer. A penalty strategy is proposed, where DQNN is employed for calculating the penalty for verifying the feasibility of the solution generated in case of violations in model constraints. The table indicates that the SAEO-DQNN achieved low values for transportation cost, transshipment cost, and total cost, recorded at 0.391, 0.518, and 1.012, respectively. Transportation and transshipment costs were reduced using the SAEO algorithm, leading to an improvement in inventory routing. Furthermore, the total must remain constant. To determine punishments and uses the SAEO in combination with the DQNN. In the subsequent role, SEAO + DQNN was responsible for coordinating the movement of perishable goods throughout a supply chain.

5 Conclusions and Future Work

In this paper, a supply chain comprising a central warehouse, retailers, multiple vehicles, and multiple products is considered. The retailers have to be replenished to alleviate the shortcomings of products, and this has to be executed with low cost and the minimum number of vehicles. This problem is considered an IRP with variable demands from the retailers. A novel SAEO algorithm is presented here to optimize the inventory routing process considering various costs, like inventory holding, lost sales, transshipment, and transportation costs. Further, a simple penalty approach is considered, wherein a penalty is introduced when the constraints are violated. The different costs along with the penalty decide the fitness of the solution generated, where the best routing corresponds to minimum fitness. Here, a DQNN is employed for estimating the penalty. The developed SAEO-DQNN is analyzed for its effectiveness based on transportation, transshipment, and total costs, and is observed to have achieved values of 0.391, 0.518, and 1.012. A further course of research could be the inclusion of product substitutions to alleviate product shortcomings by substituting one product by another. IRP for special goods, like perishable or hazardous materials can be considered in the future.

Acknowledgements Not Applicable.

Author Contributions Sreeparnesh Sharma S, and Dr.Naveen J contributed to the technical design of the proposal outlined in the study. Sreeparnesh Sharma S contributed to data acquisition, analysis, and interpretation through the libraries included in this work. Sreeparnesh Sharma S drafted the manuscript initially; Dr.Naveen J critically revised its content. All authors read and approved the final manuscript and agree to be accountable for all aspects of the work, ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

Funding This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability The dataset that is adopted in our experiments is publicly available and can be accessed as follows: Dataset is available for the public and can be accessed via the following link: http://www.leandro-coelho.com/instances/inventory-routing/. The DOI for the dataset used is https://doi.org/10.1016/j.cor.2011.12.020. The dataset was accessed on February 15, 2023.

Declarations

Conflict of Interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Consent for Publication The authors provide consent to publish the images in the manuscript. The data used in the publication is publicly available. We provide respective citations for each of the data sources.

Consent to Participate The authors provide the appropriate consent to participate.

Ethics Approval and Consent to Participate Not applicable.

References

1. Antic S, Djordjevic Milutinovic L, Lisec A (2022) Dynamic discrete inventory control model with deterministic and stochastic demand in pharmaceutical distribution. Appl Sci 12(3):1536. https://doi.org/10.3390/app12031536
2. Liu P, Hendalianpour A, Razmi J, Sangari MS (2021) A solution algorithm for integrated production-inventory-routing of perishable goods with transshipment and uncertain demand. Complex Intell Syst 7(3):1349–1365. https://doi.org/10.1007/s40747-020-00264-y
3. Achamrah FE, Riane F, Limbourg S (2022) Spare parts inventory routing problem with transshipment and substitutions under stochastic demands. Appl Math Model 101:309–331. https://doi.org/10.1016/j.apm.2021.08.029
4. Abdollahi M, Yang X, Nasri MI, Fairbank M (2023) Demand management in time-slotted last-mile delivery via dynamic routing with forecast orders. Eur J Oper Res 309(2):704–718. https://doi.org/10.1016/j.ejor.2023.01.023
5. Sakhri MSA, Tlili M, Korbaa O (2022) A memetic algorithm for the inventory routing problem. J Heuristics 28(3):351–375. https://doi.org/10.1007/s10732-022-09497-1
6. Ab Halim HZ, Mohd Azliana NN, Baharom N, Fauzi NF, Ahmad Bakhtiar NS, Khairudin NI (2021) Green inventory routing problem using hybrid genetic algorithm. J Comput Res Innov 6(4):10–19. https://doi.org/10.24191/jcrinn.v6i4.254
7. Alarcon Ortega EJ, Schilde M, Doerner KF (2020) Matheuristic search techniques for the consistent inventory routing problem with time windows and split deliveries. Oper Res Persp 7:100152. https://doi.org/10.1016/j.orp.2020.100152
8. Sivadevuni SS, Kumar Ravichandran S (2023) Systematic literature review—IoT-based supply chain management in industry 4.0. Springer, Cham, pp 291–302. https://doi.org/10.1007/978-3-031-18497-0_22
9. Touzout FA, Ladier AL, Hadj-Hamou K (2022) An assign-and-route matheuristic for the time-dependent inventory routing problem. Eur J Oper Res 300(3):1081–1097. https://doi.org/10.1016/j.ejor.2021.09.025
10. Song L, Wu Z (2023) An integrated approach for optimizing location-inventory and location-inventory-routing problem for perishable products. Int J Transport Sci Technol 12(1):148–172. https://doi.org/10.1016/j.ijtst.2022.02.002
11. Archetti C, Ljubić I (2022) Comparison of formulations for the inventory routing problem. Elsevier B.V, Amsterdam. https://doi.org/10.1016/j.ejor.2021.12.051
12. Sivadevuni SS, Maiya R, Ganesh D, Krishna AR (2024) Intelligent metering innovations for urban utility management: a review. Inst Electr Electron Eng (IEEE). https://doi.org/10.1109/icacrs58579.2023.10405137
13. Ghasemkhani A, Tavakkoli-Moghaddam R, Rahimi Y, Shahnejat-Bushehri S, Tavakkoli-Moghaddam H (2022) Integrated production-inventory-routing problem for multi-perishable products under uncertainty by meta-heuristic algorithms. Int J Prod Res 60(9):2766–2786. https://doi.org/10.1080/00207543.2021.1902013
14. Sharma Sivadevuni S, Ravichandran SK (2023) Tracking and localization of devices—an IoT review. In: 2023 International Conference on Inventive Computation Technologies (ICICT), IEEE, 2023;1321–1325. https://doi.org/10.1109/ICICT57646.2023.10134278.
15. Imran M, Habib MS, Hussain A, Ahmed N, Al-Ahmari AM (2020) Inventory routing problem in supply chain of perishable products under cost uncertainty. Mathematics 8(3):382. https://doi.org/10.3390/math8030382
16. Achamrah FE, Riane F, Limbourg S (2022) Solving inventory routing with transshipment and substitution under dynamic and stochastic demands using genetic algorithm and deep reinforcement learning. Int J Prod Res 60(20):6187–6204. https://doi.org/10.1080/00207543.2021.1987549
17. Archetti C, Guastaroba G, Huerta-Muñoz DL, Speranza MG (2021) A kernel search heuristic for the multivehicle inventory routing problem. Int Trans Oper Res 28(6):2984–3013. https://doi.org/10.1111/itor.12945
18. Alinaghian M, Tirkolaee EB, Dezaki ZK, Hejazi SR, Ding W (2021) An augmented Tabu search algorithm for the green

inventory-routing problem with time windows. Swarm Evol Comput 60:100802. https://doi.org/10.1016/j.swevo.2020.100802
19. Liu A, Zhu Q, Xu L, Lu Q, Fan Y (2021) Sustainable supply chain management for perishable products in emerging markets: an integrated location-inventory-routing model. Transp Res E Logist Transp Rev 150:102319. https://doi.org/10.1016/j.tre.2021.102319
20. Gruler A, Panadero J, de Armas J, Pérez JAM, Juan AA (2020) A variable neighborhood search simheuristic for the multiperiod inventory routing problem with stochastic demands. Int Trans Oper Res 27(1):314–335. https://doi.org/10.1111/itor.12540
21. Friske MW, Buriol LS, Camponogara E (2022) A relax-and-fix and fix-and-optimize algorithm for a maritime inventory routing problem. Comput Oper Res 137:105520. https://doi.org/10.1016/j.cor.2021.105520
22. Mahjoob M, Fazeli SS, Milanlouei S, Tavassoli LS, Mirmozaffari M (2022) A modified adaptive genetic algorithm for multi-product multi-period inventory routing problem. Sustain Oper Comput 3:1–9. https://doi.org/10.1016/j.susoc.2021.08.002
23. Sivadevuni SS, Naveen J (2024) WSETO: wild stock exchange trading optimization algorithm enabled routing for NB-IoT tracking system. Int J Inf Technol. https://doi.org/10.1007/S41870-024-02130-3/METRICS
24. “Inventory-Routing—Prof. Leandro C. Coelho, Ph.D.” Accessed: Jul. 03, 2023. [Online]. Available: https://www.leandro-coelho.com/instances/inventory-routing/