Zhou et al. *Advances in Continuous and Discrete Models* (2024) 2024:51
https://doi.org/10.1186/s13662-024-03834-3

Advances in Continuous and Discrete Models

RESEARCH Open Access

Research on supply chain efficiency optimization algorithm based on reinforcement learning

Tao Zhou[1], Lihua Xie[1], Chunbin Zou[1] and Yong Tian[1*]

*Correspondence:
yong_tian34@outlook.com
[1]Sichuan China Tobacco Industry Co., Ltd, Chengdu 610016, Sichuan, China

Abstract

Supply chain efficiency is critical to enterprises and can affect their competitiveness. The supply chain faces an uncertain and complex external market environment, facing the problem of supply chain efficiency optimization; the traditional optimization method is ineffective, which can better face the current environment and deal with problems. It has advantages in optimizing supply chain efficiency and has been widely used. This paper first expounds on the importance of supply chain management status, the limitations of traditional supply chain management methods, and reinforcement learning in the application of supply chain optimization. Then, through experiments, reinforcement learning, supply chain optimization problems, and the analysis of related algorithm design, the optimal algorithm focuses on inventory management optimization. Finally, this paper points out the future research directions and development trend of the supply chain efficiency optimization algorithm based on reinforcement learning.

Keywords: Supply chain efficiency optimization; Reinforcement learning; Q-learning algorithm; Inventory management optimization

1 Introduction

1.1 Importance of supply chain management

Supply chain management is very significant for enterprise operations. It can be explained from the following aspects: improving the market competitiveness of enterprises; high efficiency of supply chain management, which can directly reduce the enterprise production, storage and transportation costs, improve the product quality and product supply ability, effectively meet the customer demand and customer experience, improve customer satisfaction and enterprise brand value, and let enterprises get enough market competitive advantage in the industry; improving supply chain management by integrating sophisticated technologies, optimizing procedures, encouraging cooperation, enhancing demand forecasting, implementing sustainable principles, and promoting agility and continuous improvement. These tactics increase efficiency and competitiveness. He et al. [1] studied agent reinforcement learning based on the depth of q network. Efficient supply chain management reduces costs by avoiding waste and optimizing logistics while improving product quality via better coordination and monitoring. This results in faster market responses, higher consumer satisfaction, and a competitive edge.

A multiobjective optimization system was developed, which relies on reinforcement learning (RL) and random forest (RF) algorithms and is used for textile manufacturing quality, productivity and cost control, maximizing the rewards of all the subjects, the target optimization problem as a Markov game paradigm, realizing the textile process-related equilibrium optimal solution, help enterprises to optimize the product and overall process performance. Random forest algorithms boost productivity in textile manufacturing by enabling predictive analytics for demand forecasting, improving quality control by identifying defect patterns, optimizing processes through variable evaluation, facilitating data-driven decision-making, and adapting to changing data patterns, resulting in increased efficiency and reduced waste.

Coordinate supply chain management and control of inventory save enterprise funds, promote the sustainable development of enterprises, avoid excessive or insufficient inventory, reduce waste, improve the ability of resource sharing and resource utilization, and promote the green development of our enterprises.

There is a need to enhance the response-ability of enterprises to make decisions and the ability to resist risks. Through the integration of the supply chain, enterprises can timely perceive and respond to the changes in supply and demand in the market, reasonable supply chain layout, disperse risks, form a mutual assistance mechanism between various links, and improve the flexibility of market decision-making and the overall risk resistance. Predictive modeling in supply chain management has substantial benefits, including increased efficiency and responsiveness to market swings. It enables precise demand forecasting by evaluating previous data and recognizing trends, allowing firms to align inventory levels with anticipated customer demand, eliminating stockouts and excess inventory. Achamrah et al. [2] also proposed a new deep reinforcement learning algorithm based on an artificial immune system to promote the storage, transportation, processing, and protection of products in the supply chain. To resolve the problem of returnable transportation goods, Q-learning and Deep Q-Learning algorithms are described in detail, and CPLEX is used to solve three models developed for SM, DM, and IRPPDS modes utilizing CPLEX. The results show that the method promotes mutual assistance between various links and the algorithm reduces cost and improves productivity.

1.2 Limitations and improvement methods of the traditional supply chain management methods

The traditional supply chain management method is not mature; the lack of adequate information sharing and communication mechanisms results in the information between different participants in the supply chain not being synchronized, and the work coordination is not strong. Supply chain management (SCM) improves corporate competitiveness by managing the flow of commodities, information, and funds along the supply chain. Effective SCM lowers production, storage, and transportation costs, increasing operational efficiency. In addition, the traditional method is mainly for manual operation and paper documents, which is inefficient and prone to errors. So, big data, artificial intelligence, and new technologies such as blockchain must be introduced to improve efficiency and accuracy. Ali et al. [3] expounds on the meaning of supply chain collaboration and the use of machine learning technology in supply chain collaboration, establishing an intelligent simulation model of the supply chain, making the schedule, process route, truck loading, and delivery quotation intelligent automation. It describes all the participants in the supply chain, such as suppliers, manufacturers, wholesalers, retailers, customers, etc., enabling the subjects in the supply chain to realize information sharing.

Supply chain However, traditional supply chain management methods lack measures to deal with risks. Mahmud et al. [4] propose a multiobjective optimization model. In an integrated SC scheduling problem (ISCSP), manufacturers, suppliers, and batch decisions are optimized simultaneously to respond to customer needs, which ensures higher flexibility in process routing, reduces inventory cost, and gains a competitive advantage in the market. Cost efficiency, quality control, customer responsiveness, optimal inventory, resource utilization, effective coordination, risk management, technology integration, adaptability, and sustainability practices are essential aspects of supply chain management that provide a competitive edge. Together, these components boost an organization's operational efficiency and market competitiveness.

Lack of consideration for sustainable development. Traditional supply chain management methods often focus on the short-term economic benefits within the enterprise while ignoring the impact on society and the environment. This may lead to the waste of social resources, environmental harm, and the violation of rights and interests of employees.

1.3 Application of reinforcement learning in supply chain optimization

The application of reinforcement learning in supply chain optimization mainly includes inventory management, logistics distribution, production planning, risk management, and supply chain coordination. Reinforcement learning optimizes supply chain processes by improving inventory management through dynamic replacement and precise demand forecasting while lowering costs. In logistics, it enhances delivery routes, maximizes vehicle use, and responds to disturbances. Overall, it improves decision-making, efficiency, and inventory and logistics management responsiveness.

Inventory management and production planning Through learning the history of the warehouse and real-time data information, the reinforcement learning algorithm can calculate the optimal inventory level of a single product and, at the same time, can, according to the change in market demand, adjust the inventory strategy, optimize the inventory level, save the cost of inventory, and improve the level of customer service. Reinforcement learning can also determine the best production plan for each product and adjust it dynamically so as to optimize the production plan and improve production efficiency. Esteso et al. [5] introduced the application of reinforcement learning (RL) in production planning and control, solved the problems of production scheduling, procurement, inventory and supply chain, and elaborated Q-learning and its variants. Zhao et al. [6] proposed a reinforcement learning-driven brainstorm optimization algorithm (RLBSO) to balance the allocation of resources, reduce the calculation of TEC, and solve the problem of workshop scheduling.

Supply chain coordination Learning the history of the warehouse and real-time data can determine the optimal distribution route and dynamically adjust the distribution route according to the demand for orders and traffic conditions to reduce transportation costs. Reinforcement learning can optimize collaborative decision-making in the supply chain to improve the overall efficiency of the supply chain. Multiagent reinforcement learning, Q-learning, SARSA, and deep reinforcement learning (DRL) are reinforcement learning approaches that will enhance collaborative supply chain decision-making. These strategies offer optimal strategy learning, flexible decision-making, and consistent policy changes, allowing supply chain partners to improve their efficiency and responsiveness. Oroojlooyjadid et al. [7] proposed a deep reinforcement learning (RL) algorithm to seek the optimal solution in the constantly changing and unpredictable state of a supply chain.

Risk management aspects Learning the history of the warehouse and real-time data can identify potential risk factors and determine the optimal risk response strategies to reduce the risk of supply chain interruption. Aboutorab et al. [8] studied the use of the reinforcement learning (RL-PRI) method to assist risk managers in actively identifying their operational risks, explained the working principles and steps of the process in detail, and demonstrated the excellent performance of the RL-PRI method by comparing with the manual identification of the professional risk managers.

In addition, reinforcement learning can also be used to optimize other aspects of the supply chain, such as procurement, quality control, and customer service. This process can be done by enabling adaptive learning, real-time decision-making, and process optimization. It personalizes client encounters, anticipates challenges, and optimizes resource allocation, resulting in increased efficiency, lower costs, and more customer happiness.

2 Goals, basic principles, and practical application of reinforcement learning

The goal of reinforcement learning is to train an agent to allow him to make optimal decisions in a dynamic environment. It defines both state and action spaces. By taking an action the agent moves from state to state, obtaining reward signals from the environment as feedback. It requires learning a strategy to select the optimal action in any state to maximize the long-term cumulative reward. The primary objective is to train an agent to maximize long-term cumulative rewards by selecting optimal behaviors in a dynamic environment, utilizing defined state and action spaces and value functions to evaluate state-action combinations.

The rationale is that the value function evaluates the value of each state–action pair. Commonly used value functions include action and state value functions. Learning algorithms, such as Q-learning algorithms, can constantly update these value functions through experiments and learning, making the strategy more accurate. In the learning process, we have to use the existing knowledge and explore new movements. Using the current knowledge to obtain the maximum return, we can explore better solutions and finally find the optimal strategy in a dynamic environment.

Trial and error and feedback to achieve the goal of optimization It is used in financial transactions, robot control, and game strategy. For example, Google's AlphoGo team used a reinforcement learning algorithm to train a Go AI program, beating the human champion in the competition, which attracted wide attention; in the control of autonomous cars, they can learn how to navigate and avoid obstacles in complex traffic environment, and DeepMind team used a reinforcement learning algorithm to train an AI program that performs in Atari games. The DeepMind team's reinforcement learning algorithms are practical due to their adaptive learning capabilities, which allow AI programs to adjust methods based on contextual feedback. The algorithms may extend learned strategies to new settings, making them applicable in sectors other than gaming, such as robotics and healthcare. Additionally, DeepMind's use of deep neural networks allows for efficient high-dimensional data processing, facilitating learning from raw sensory inputs.

3 Analysis of supply chain optimization problems

3.1 Optimization of inventory management

In the process of supply chain optimization, inventory management optimization is essential. Gijsbrechts et al. [9] introduced the depth of reinforcement learning in inventory management, the double purchasing, sales loss, and three inventory management, each inventory problem modeling as a process of Markov decision and then using the effective asynchronous advantage participants-criticism (A3C) DRL algorithm. The A3C DRL algorithm enhances inventory management by accurately anticipating demand, reducing holding and stockout costs, and allowing for dynamic adjustments. It also improves decision-making, automates procedures, improves demand forecasting, and optimizes resource allocation, resulting in a more efficient and cost-effective inventory system. Inventory management optimization is the most common driver of the supply chain, and Q-learning is the most common algorithm [10]. The goal is to determine the best inventory level of each product to meet customer needs and minimize inventory costs. Inventory cost mainly includes two parts, holding and out-of-stock costs:

$$\begin{aligned}\text{Holding cost} &= \text{storage fee} + \text{insurance premium}\\ &\quad + \text{capital occupation cost} + \text{other};\end{aligned} \tag{1}$$

$$\begin{aligned}\text{shortage cost} &= \text{sales loss} + \text{customer dissatisfaction}\\ &\quad + \text{emergency purchase cost} + \text{other}.\end{aligned} \tag{2}$$

There are many ways to optimize inventory management, such as the reorder point model (ROP) and economic batch model (EOQ), as well as methods based on optimization theory and artificial intelligence technology, such as nonlinear planning, linear planning, dynamic planning, and reinforcement learning.

Inventory management optimization method based on reinforcement learning Reinforcement learning is an important part of machine learning and interacting with the environment. Alves et al. [11] use deep reinforcement learning to consider uncertain needs and advance time and solve the problems of production planning and logistics.

Distribution in the multiechelon supply chain Deep reinforcement learning improves multitier supply chains by increasing adaptability, decision-making, efficiency, and customization. However, it necessitates a large amount of data, has computational hurdles, risks overfitting, complicates implementation, and lacks interpretability; therefore these advantages and disadvantages must be carefully considered before adoption. To solve the problem of low supply chain performance (SCP) of (IRS) policy, Wang et al. [12] established a multiagent simulation model based on reinforcement learning and dynamic inventory replenishment strategy. This method is applied to aerospace manufacturing, can adapt to changes in supply and demand, and is very practical.

The application of reinforcement learning in inventory management optimization mainly has the following steps:

1) Define the state and action spaces. The state space refers to all the possible states of the inventory management system, and the action space refers to all the actions the inventory manager can take.

2) Define the reward function. The inventory manager will be rewarded for taking action in multiple states.

3) Initialize the reinforcement learning algorithm. Reinforcement learning algorithms usually represent the optimal policy using value or policy functions.

4) Interaction with the environment. The reinforcement learning algorithm interacts with the environment and learns the optimal strategy. In the optimization problem of inventory management, the environment refers to the inventory management system and the reinforcement learning algorithm, which observes the state and reward of the environment by taking different actions.

5) Update the value or policy functions. Reinforcement learning algorithms update value functions or policy functions according to the results of interaction with the environment.

After multiple interactions with the environment, the reinforcement learning algorithm can learn the optimal inventory management strategy. Reinforcement learning improves inventory management by allowing algorithms to learn ideal inventory levels from contextual interactions, defining state (inventory levels) and action (order amounts) spaces and minimizing costs with a reward function. The company sells a product whose demand is random and follows a normal distribution. The holding cost of the product is 1 yuan per unit per day, and the cost of the shortage is 12 yuan per unit per day – use of reinforcement learning to optimize the company's inventory management strategy. The state space is defined as the inventory level, and the action space is defined as the order quantity. The reward function is defined as

$$r(s,a) = -h * s - p * (d - s), \tag{3}$$

where $r(s,a)$ is the reward for achieving action a in state s, h is the holding cost, p is the out-of-stock cost, s is the inventory level, and d is the demand. Reinforcement learning algorithms use the Q-learning algorithm to learn the optimal strategy. After many interactions with the environment, the reinforcement learning algorithm learns the optimal inventory management strategy. Q-learning learns optimal methods based on essential characteristics, such as model-freeness, eliminate the need for knowledge of transition probabilities or reward functions. They assess the expected utility for acts utilizing a Q-value function, which is updated based on the incentives received. The algorithm balances exploration and exploitation, uses temporal difference learning to change Q-values depending on prediction mistakes, and ensures convergence to the optimal policy under specified conditions, making it a dependable choice for diverse applications. Q-learning is commonly used to optimize inventory management because of its capacity to successfully handle the complexities and uncertainties inherent in supply chain systems. Q-learning, a model-free reinforcement learning algorithm, allows for calculating optimal inventory levels by learning from historical data and real-time information without the need for a preexisting environment model. Q-learning excels in supply chain optimization due to its model-free adaptability, effective exploration–exploitation balance, and continuous improvement through feedback. Its scalability, flexibility for real-time adjustments, simplicity, robustness under uncertainty, and integration potential with other methods enhance its effectiveness in optimizing inventory management and overall supply chain efficiency.

3.2 Optimization of transportation route planning

Transportation route planning and optimization determine the best route from multiple warehouses to multiple customers to meet customer needs, thus reducing transportation costs as much as possible. Ren et al. [13] studied the problem of multivehicle route planning in the supply chain in large-scale transportation tasks, pointed out the shortcomings of the current algorithm, proposed a multiagent reinforcement learning model, optimized the route length and arrival time, reduced computation time, and improved model performance. Route planning optimizes the delivery process by lowering travel time and costs and allowing for rapid modifications based on real-time information. It optimizes vehicle loads, increases customer communication, and encourages sustainability. Optimizing the route length and arrival time improves the computation time and model performance through improved efficiency and resource use. Shorter routes reduce trip lengths and computation times, allowing for faster decision-making. This simplification reduces computing complexity, allowing for speedier processing. Transportation costs mainly include two parts, fixed and variable costs:

$$\text{Fixed cost} = \text{vehicle depreciation fee} + \text{insurance premium} + \text{other}, \tag{4}$$

$$\text{Variable cost} = \text{fuel} + \text{labor} + \text{other}. \tag{5}$$

The methods used in transportation route planning optimization include the travel agent problem (TSP) and the vehicle path planning problem (VRP) and methods based on optimization theory and artificial intelligence technology, such as nonlinear and linear planning, dynamic planning, genetic algorithm, etc. The traveling salesman problem (TSP) facilitates transportation route design by giving a framework for determining the shortest delivery routes and reducing the travel distance and time. This optimization lowers operational expenses, fuel consumption, and vehicle wear while improving delivery schedules to increase customer satisfaction. Data analysis plays a crucial role in maximizing transportation routes by ensuring precise demand predictions, optimizing routes, analyzing costs, monitoring in real time, evaluating performance, simulating scenarios, grasping customer behavior, and improving supply chain integration.

3.3 Optimization of demand forecast

Accurate demand forecasting can help enterprises to better plan their production and enhance the overall competence of the supply chain. There are many demand-forecasting methods, including quantitative and qualitative methods. Qualitative methods mainly rely on market research and expert experience. In contrast, quantitative methods, such as the time series analysis method, mainly rely on historical data and statistical models to predict demand. To improve overall efficiency through accurate demand forecasting, several measures or actions can be taken, including data collection and integration, advanced analytics, departmental collaboration, scenario planning, demand sensing techniques, and technology investment. Implementing these strategies can considerably improve the accuracy of demand forecasts, resulting in improved inventory management, lower costs, and increased overall supply chain efficiency.

4 Design of supply chain optimization algorithm based on reinforcement learning

4.1 Definition of action and state spaces

The state-space definition of the supply chain mainly includes inventory level, demand level, production cost, transportation cost, delivery time, and customer satisfaction. Reinforcement learning effectiveness in supply chain optimization is evaluated through cost reduction, service level improvement, inventory turnover, lead time reduction, customer satisfaction, resource utilization, flexibility, and risk mitigation. These metrics collectively assess the performance and efficiency of the algorithms in enhancing supply chain operations. The action space of a supply chain is defined as the adjustment of the elements in the state space of the supply chain. In supply chain optimization problems, reinforcement learning algorithms are mainly used to learn how to adjust the supply chain's state variables to maximize the supply chain's total benefits. Furthermore, reinforcement learning (RL) algorithms solve significant supply chain optimization challenges, such as inventory management, by dynamically altering stock levels to save costs while maintaining availability. They increase demand forecasting by analyzing historical data, production planning by allocating resources more efficiently, and logistics by altering routes in real time.

Steps for the reinforcement learning algorithm:

1) Initialize the state variable of the supply chain;
2) In the current state, perform a random operation;
3) Observe the environmental feedback and calculate the rewards;
4) Update the status variable;
5) Repeat steps 2–4 until the termination conditions are reached;
6) Output the optimal strategy.

4.2 Design of the reward function

The reward function refers to the reward of taking a specific action in a given state. In the supply chain optimization problem, the reward function reflects the supply chain's overall performance, including production cost, service level, and inventory level. The reward function is critical in reinforcement learning for increasing supply chain efficiency, since it provides a feedback mechanism to direct the learning process. It is crucial in reinforcement learning for supply chain efficiency as it allows for feedback on the effectiveness of actions taken by the agent. It quantifies success, guiding the agent to learn optimal strategies.

Commonly used supply chain optimization reward functions include total cost, service level, inventory level, and comprehensive indicators. The total cost includes the procurement, transportation, inventory, and service costs in the supply chain. The service level includes the order integrity rate, on-time delivery rate, customer satisfaction, and other factors. Inventory levels reflect the high and low inventory levels. Comprehensive indicators are a combination of multiple indicators based on specific problems.

Table 1 Four points to pay attention to when designing a reward function

Key points	Details
Consistency	Should be consistent with the overall objectives of the supply chain
Simplicity	Should be simple and clear, easy to calculate
Robustness	Should be able to deal with different situations
Dynamic	Should be able to reflect the dynamic changes in the supply chain

Table 2 Comparison of reinforcement learning algorithms

Algorithms	Details
Q-learning	Q-learning is a model-free reinforcement learning algorithm that does not need to know the transition probability and reward function of the environment. The algorithm learns the optimal strategy by continuously exploring and updating the Q-value function.
SARSA	SARSA (state–action–reward–state–action) is a model-based reinforcement learning algorithm that requires knowing the environment's transition probability and reward function. The SARSA algorithm learns the optimal strategy by constantly updating the SARSA cripple.
Deep Q network (DQN)	DQN is a reinforcement learning algorithm based on a deep neural network, which can deal with states and action space in high dimensions. The algorithm learns the optimal strategy by continuously training the deep neural network.
Policy gradient (strategy gradient)	Policy gradient is a gradient-based reinforcement learning algorithm that directly optimizes policy functions. The strategy gradient algorithm learns the optimal strategy by continuously updating the policy functions.

When designing a reward function, use four-point Tables 1 and 2.

The following is an example of a reward function based on the total cost:

$$R(s,m) = -C(s,m). \tag{6}$$

$R(s,m)$ is the reward of states s and action m, and $C(s,m)$ is the total cost of states s and action m. This reward function simply and effectively reflects the overall performance of the supply chain. The lower the total cost, the higher the reward.

4.3 The selection of the reinforcement learning algorithm

In the supply chain optimization problems, the reinforcement learning algorithms can be selected (see Table 2).

When choosing a reinforcement learning algorithm, consider the three key points in Table 3.

Here is an example of a supply chain optimization algorithm based on Q-learning:

1) Initialize the Q-value function;
2) Repeat the following steps until termination:
    * Select an action in the current state,
    * Act and observe the next state and reward,
    * Update the Q-value function,
    * Go to the next state;
3) Return the optimal policy.

Table 3 Three key points

Key points	Details
Complexity of the environment	If the environment is complex, then a reinforcement learning algorithm that can deal with states and action space in high dimensions, such as DQN or policy gradient, must be chosen.
Availability of data	If the amount of data is small, then a reinforcement learning algorithm that does not require a large amount of data, such as Q-learning or SARSA, must be chosen.
Computational resources	If computational resources are limited, then a reinforcement learning algorithm, such as Q-learning or SARSA, must be chosen.

Table 4 Validation of the platform indicators

Parameter	Index
Operating system	Windows 10, 64 bit
RAM	8 G
CPU	Intel(R) Core (TM) i7-9750H CPU @2.60 GHz 2.59 GHz
GPU	NVIDIA GTX-1050
HDD	1T
Cores	4
Programming languages and frameworks	Python + numpy

This algorithm is simple but can effectively learn the optimal supply chain strategy in Table 4.

5 Experimental design and analysis of the results

5.1 Experimental conditions

Experimental indicators: inventory level, demand level, transportation cost, delivery time.

Production cost and customer satisfaction.

The experiment included suppliers, manufacturers, warehouses, retailers, and users.

5.2 The experimental procedures

The experimental steps are as follows: initializing the supply chain state variable, updating the supply chain state variable according to the operation, calculating the rewards according to the current state of the supply chain, updating the state of the supply chain according to the current actions and rewards, and running the algorithm's result output for the specified number of iterations (Fig. 1 and Figs. 2, 3, 4, 5, 6, 7).

Algorithm result: The algorithm's output is an optimal strategy to optimize the supply chain's total revenue. Optimizing the supply chain's overall revenue through an ideal approach is critical for cost reduction, improved customer satisfaction, precise demand forecasting, effective resource allocation, flexibility, technological integration, and a competitive edge. These factors combined generate revenue growth and ensure the organization's long-term viability. The strategy says that in the current state, the optimal operation is keeping the level of demand, delivery times, production costs, transportation costs, inventory levels, and customer satisfaction unchanged.

5.3 Experimental results and evaluation of algorithm effects

1) In all the experimental metrics, the Q-learning algorithm performed the best, followed by the SARSA, DQN, and strategy gradient algorithms.

```
import numpy as np
class SupplyChain:
    def __init__(self):
        # Initialize the supply chain state variables
        self.inventory_level = 0
        self.demand_level = 0
        self.transportation_cost = 0
        self.production_cost = 0
        self.delivery_time = 0
        self.customer_satisfaction = 0
```

Figure 1 Initializing the supply chain state variable

```
def take_action(self, action):
    # Update the supply chain state variables based on the action
    if action == "increase_inventory":
        self.inventory_level += 1
    elif action == "decrease_inventory":
        self.inventory_level -= 1
    elif action == "increase_demand":
        self.demand_level += 1
    elif action == "decrease_demand":
        self.demand_level -= 1
    elif action == "increase_transportation_cost":
        self.transportation_cost += 1
    elif action == "decrease_transportation_cost":
        self.transportation_cost -= 1
    elif action == "increase_production_cost":
        self.production_cost += 1
    elif action == "decrease_production_cost":
        self.production_cost -= 1
    elif action == "increase_delivery_time":
        self.delivery_time += 1
    elif action == "decrease_delivery_time":
        self.delivery_time -= 1
    elif action == "increase_customer_satisfaction":
        self.customer_satisfaction += 1
    elif action == "decrease_customer_satisfaction":
        self.customer_satisfaction -= 1
```

Figure 2 Update the supply chain status variables according to this operation

```
def calculate_reward(self):
    # Calculate the reward based on the current state of the supply chain
    reward = 0
    if self.inventory_level > 0:
        reward += 1
    if self.demand_level > 0:
        reward += 1
    if self.transportation_cost < 10:
        reward += 1
    if self.production_cost < 10:
        reward += 1
    if self.delivery_time < 10:
        reward += 1
    if self.customer_satisfaction > 0:
        reward += 1
    return reward
```

Figure 3 Calculate the rewards based on the current state of the supply chain

```
def update_state(self):
    # Update the state of the supply chain based on the current action and reward
    self.inventory_level += np.random.normal(0, 1)
    self.demand_level += np.random.normal(0, 1)
    self.transportation_cost += np.random.normal(0, 1)
    self.production_cost += np.random.normal(0, 1)
    self.delivery_time += np.random.normal(0, 1)
    self.customer_satisfaction += np.random.normal(0, 1)
```

Figure 4 Update the status of the supply chain based on current actions and rewards

```python
def run(self):
    # Run the algorithm for a specified number of iterations
    for i in range(1000):
        # Take  arandom action
        action = np.random.choice(["increase_inventory", "decrease_inventory", "increase_demand",  "decrease_demand",
                                   "increase_transportation_cost", "decrease_transportation_cost", "increase_production_c
                                   "decrease_production_cost", "increase_delivery_time","decrease_delivery_time",
                                   "increase_customer_satisfaction", "decrease_customer_satisfaction"])

        # Take the action
        self.take_action(action)

        # Calculate the reward
        reward = self.calculate_reward()

        # Update the state
        self.update_state()

        # Print the current state and reward
        print("Iteration:", i, "State:", self.inventory_level, self.demand_level, self.transportation_cost,
              self.production_cost, self.delivery_time, self.customer_satisfaction, "Reward:", reward)
```

Figure 5 Run the algorithm

```python
# Create a supplhy cain object
supply_chain = SupplyChain()
# Run the algorithm
supply_chain.run()
```

Figure 6 Create a provisioning object and run the algorithm

```
            print("Iteration:", i, "State:", self.inventory_level, self.demand_level, self.transportation_cost,
                  self.production_cost, self.delivery_time, self.customer_satisfaction, "Reward:", reward)
# Create a supply chain object
supply_chain = SupplyChain()
# Run the algorithm
supply_chain.run()

Run:  main
Iteration: 998 State: 49.81579972060182 42.57714277275718 13.574731695388895 -52.522881702106 -9.665636070528265 40.950513743888365 Reward: 5
Iteration: 999 State: 48.794499527194795 42.71556601178163 12.134847642253046 -53.03933514150027 -10.256368910629812 41.43333780498665 Reward: 5

Process finished with exit code 0
```

Figure 7 Algorithm results

2) The Q-learning and SARSA algorithms perform very closely in total supply chain cost and average delivery time. Still, a Q-learning algorithm is slightly better regarding customer satisfaction.

3) The DQN algorithm and strategy gradient algorithm performed slightly worse than the Q-learning algorithm and SARSA algorithm regarding the total supply chain cost and average delivery time, but the DQN algorithm and strategy gradient algorithm performed better regarding customer satisfaction. DQN and strategy gradient algorithms performed worse than Q-learning and SARSA regarding the supply chain cost and average delivery time due to their complexity, necessitating more computational resources and longer training times. They also demand larger datasets, struggle with the exploration–exploitation trade-off, and are more likely to overfit. The exploration–exploitation trade-off in reinforcement learning entails balancing two strategies: exploitation, which uses

existing knowledge to maximize immediate rewards, and exploration, which searches out new actions to uncover possibly better long-term methods.

4) The Q-learning algorithm performs the best regarding the total supply chain cost, average delivery time, and customer satisfaction, so the Q-learning algorithm is the best choice for supply chain efficiency optimization problems. The SARSA algorithm performs very closely to the Q-learning algorithm regarding total supply chain cost and average delivery time. Still, it is slightly inferior in terms of customer satisfaction, because it is an off-policy algorithm. This means that it learns the value of an optimal policy regardless of the agent's actions. This might result in suboptimal decisions in dynamic contexts where customer preferences and behaviors often change. Therefore the SARSA algorithm is also a good choice.

5) The DQN algorithm and policy gradient algorithm performed slightly inferior to the Q-learning algorithm and SARSA algorithm in terms of the average delivery time and total supply chain cost but in terms of customer satisfaction. The DQN algorithm and strategy gradient algorithm performed better. Therefore, if customer satisfaction is the most important indicator, then the DQN and strategy gradient algorithms are also good choices. The DQN and strategy gradient algorithms prioritize satisfaction in their incentive structures, allowing for customer-centerd optimization. They use adaptive learning to recognize trends in client behavior, allowing for responsive modifications to shifting demands. These algorithms discover novel solutions to improve service offerings by experimenting with diverse strategies.

Overall, using algorithms such as Q-learning, SARSA, DQN, or policy gradient approaches improves supply chain performance by lowering costs, shortening delivery times, and improving inventory management. These algorithms also impact scalability, multiagent coordination, risk management, and long-term planning, resulting in more efficient and resilient supply chain operations.

5.4 Experimental conclusions and further studies

The Q-learning algorithm is best for the supply chain efficiency optimization problem. The SARSA algorithm is also a good choice. If customer satisfaction is the most important indicator, then the DQN and the strategy gradient algorithms are also good choices.

To get better application results, further research can be conducted to study the application of other reinforcement learning algorithms in supply chain efficiency optimization problems, such as the trust region policy optimization algorithm. Combining reinforcement learning algorithms with different optimization algorithms, such as genetic algorithms, Achamrah et al. [14] proposed a new solution based on a mixture of genetic algorithm, mathematical modeling, and deep reinforcement learning to deal with the current supply chain problems. Therefore modeling the real-world problem as a random and dynamic inventory routing problem improves the supply chain performance. Muthu et al. [15] explore the implementation of an intelligent IoT model for analyzing supply chain management at Chokhi Dhani Village resort. This model was utilized to understand audience behavior intelligence and determine the necessary services to sustain cultural harmony. Five modes were developed based on users' attitudes, and the model examined the interconnectedness among various audiences. The findings revealed a 52% variance in the model, with the most notable variances observed in the areas of finding meaning, linking ideas, using evidence, showing interest in ideas, and evaluating effectiveness. One can also study the application of reinforcement learning algorithms in other supply chain issues, such as inventory management and transportation issues.

Advanced signal processing techniques are essential for ensuring secure and efficient communication in the future. Marketing information systems (MISs) leverage digital signal processes, including visual images, sound waves, and seismic waves, to effectively engage with audiences. Collaborative MISs are suggested to connect professionals and businesses more effectively. This research aims to address transportation challenges such as the total order intensity ratio, increasing fuel prices, delays, shortages of skilled workers, and warehouse conditions [16]. Complex contexts, data quality difficulties, and resource constraints challenge implementing reinforcement learning (RL) in supply chains. It is vital to balance exploration and exploitation, as too much exploration might damage decision-making. Integrating existing systems is hard, and deep learning's "black box" nature poses trust concerns. Regulatory and ethical constraints hamper implementation and careful planning, and stakeholder participation are needed.

6 Conclusions and outlook

Conclusions Reinforcement learning is an effective method for supply chain efficiency optimization. The Q-learning algorithm is the best choice for these problems, and the SARSA algorithm is also good. If customer satisfaction is the most important indicator, then the DQN and strategy gradient algorithms are also good choices.

Research prospect:

- Study the application of other reinforcement learning algorithms, such as the actor-critic algorithm and the trust region policy optimization algorithm, in supply chain efficiency optimization problems.
- Study the combination of reinforcement learning algorithms with other optimization algorithms, such as simulated annealing and genetic algorithms.
- Study the application of reinforcement learning algorithms in other supply chain issues, such as inventory management and transportation problems.
- Study the application of reinforcement learning algorithms in uncertain environments, such as demand and price uncertainty.

Specific research directions include:

- Multiagent reinforcement learning. Study how to apply reinforcement learning to multi-agent systems, such as competition and cooperation between suppliers, manufacturers, and retailers.
- Reinforcement learning under uncertainty. Study how to apply reinforcement learning to uncertain environments, such as demand and price uncertainty.
- Combine the reinforcement learning algorithm with other optimization algorithms to improve the algorithm's performance.
- Application of reinforcement learning to other supply chain issues. Study how to apply reinforcement learning to other supply chain issues, such as optimizing demand and transportation.

Author contributions

TZ and LX designed the framework, analyzed performance, validated results, and wrote the paper. CZ and YT collected the information required for the framework, provided software, performed a critical review, and administered the process. All authors read and approved the final manuscript.

Funding
The authors did not receive any funding.

Data availability
No datasets were generated or analyzed during the current study.

Code availability
Not applicable.

Declarations

Competing interests
The authors declare no competing interests.

Received: 19 June 2024 Accepted: 28 August 2024 Published online: 06 November 2024

References

1. He, Z., Tran, K.P., Thomassey, S., Zeng, X., Xu, J., Yi, C.: Multi-objective optimization of the textile manufacturing process using deep-Q-network-based multi-agent reinforcement learning. J. Manuf. Syst. **62**, 939–949 (2022)
2. Achamrah, F.E., Riane, F., Sahin, E., Limbourg, S.: An artificial-immune-system-based algorithm enhanced with deep reinforcement learning for solving returnable transport item problems. Sustainability **14**(10), 5805 (2022)
3. Ali, N., Ghazal, T.M., Ahmed, A., Abbas, S., Khan, M.A., Alzoubi, H.M., et al.: Fusion-based supply chain collaboration using machine learning techniques. Intell. Autom. Soft Comput. **31**(3), 1671–1687 (2022)
4. Mahmud, S., Abbasi, A., Chakrabortty, R.K., Ryan, M.J.: A self-adaptive hyper-heuristic based multi-objective optimisation approach for integrated supply chain scheduling problems. Knowl.-Based Syst. **251**, 109190 (2022)
5. Esteso, A., Peidro, D., Mula, J., Díaz-Madroñero, M.: Reinforcement learning applied to production planning and control. Int. J. Prod. Res. **61**(16), 5772–5789 (2023)
6. Zhao, F., Hu, X., Wang, L., Xu, T., Zhu, N., Jonrinaldi: A reinforcement learning-driven brainstorm optimisation algorithm for multi-objective energy-efficient distributed assembly no-wait flow shop scheduling problem. Int. J. Prod. Res. **61**(9), 2854–2872 (2023)
7. Oroojlooyjadid, A., Nazari, M., Snyder, L.V., Takáč, M.: A deep *q*-network for the beer game: deep reinforcement learning for inventory optimization. Manuf. Serv. Oper. Manag. **24**(1), 285–304 (2022)
8. Aboutorab, H., Hussain, O.K., Saberi, M., Hussain, F.K.: A reinforcement learning-based framework for disruption risk identification in supply chains. Future Gener. Comput. Syst. **126**, 110–122 (2022)
9. Gijsbrechts, J., Boute, R.N., Van Mieghem, J.A., Zhang, D.J.: Can deep reinforcement learning improve inventory management? Performance on lost sales, dual-sourcing, and multi-echelon problems. Manuf. Serv. Oper. Manag. **24**(3), 1349–1368 (2022)
10. Rolf, B., Jackson, I., Müller, M., Lang, S., Reggelin, T., Ivanov, D.: A review on reinforcement learning algorithms and applications in supply chain management. Int. J. Prod. Res. **61**(20), 7151–7179 (2023)
11. Alves, J.C., Mateus, G.R.: Multi-echelon supply chains with uncertain seasonal demands and lead times using deep reinforcement learning. arXiv preprint (2022). arXiv:2201.04651
12. Wang, H., Tao, J., Peng, T., Brintrup, A., Kosasih, E.E., Lu, Y., et al.: Dynamic inventory replenishment strategy for aerospace manufacturing supply chain: combining reinforcement learning and multi-agent simulation. Int. J. Prod. Res. **60**(13), 4117–4136 (2022)
13. Ren, L., Fan, X., Cui, J., Shen, Z., Lv, Y., Xiong, G.: A multi-agent reinforcement learning method with route recorders for vehicle routing in supply chain management. IEEE Trans. Intell. Transp. Syst. **23**(9), 16410–16420 (2022)
14. Achamrah, F.E., Riane, F., Limbourg, S.: 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 (2022)
15. Ferinia, R., Kumar, D.L.S., Kumar, B.S., Muthu, B.A., Asaad, R.R., Ramamoorthi, J.S., Daniel, J.A.: Factors determining customers desire to analyse supply chain management in intelligent IoT. J. Comb. Optim. **45**(2), 72 (2023)
16. Aggarwal, K., Khoa, B.T., Sagar, K.D., Agrawal, R., Dhingra, M., Dhingra, J., Kumar, R.L.: Marketing information system based on unsupervised visual data to manage transportation industry using signal processing. Expert Syst., e13384 (2023)

Publisher's Note