Ingénierie des Systèmes d'Information
Vol. 30, No. 4, April, 2025, pp. 835-844
Journal homepage: http://iieta.org/journals/isi

# Inventory Management System Using Reinforcement Learning: A Case Study

Sridhar Subramanian, Smita Mahajan*, Shrikrishna Kolhar

Symbiosis Institute of Technology, Symbiosis International (Deemed) University, Pune 412115, India

Corresponding Author Email: rajansmita@gmail.com



https://doi.org/10.18280/isi.300401

**Received:** 10 August 2024
**Revised:** 13 December 2025
**Accepted:** 24 February 2025
**Available online:** 30 April 2025

***Keywords:***
*reinforcement learning, deep Q leaning, deep reinforcement, inventory management, supply chain management*

**ABSTRACT**

This research intends to put RL methods related to SCM to use in the management of input stocks. Estimating the composition of a small retailer's inventory system, specifically to recharge Coke sales, the research aims to improve the forecast of merchandise, when they should be refilled, to fulfill client expectations. The deep Q network (DQN) algorithm is used to represent the objective of the study comparing the performance of the RL-based inventory control strategy with the classic static control method ((s, S) inventory control) in a numerical test. These financial parameters are determined along with other operational constraints, such as inventory capacity, lead time, and product order costs. The demand patterns between weekdays and weekends form the basis for the simulation of historical desire data to train DQN model. The comparison of RL-based methods in the retail industry supply chain is covered by this study monetarily. Consequentially, the study introduces RL-based methods as one of the techniques in the area of improvement of retail inventory management practical applications with real-life supply chain examples to complement and prove their success.

## 1. INTRODUCTION

Within a comprehensive framework of supply chain management, the intricacies of the heads of inventory control can be viewed as a hard nut to crack. Maintaining a sustainable equilibrium between the unpredictability of demand variabilities, considering optimum operational costs, and optimizing resources that usually involve business operations is a continuous struggle. By convention, those methods that work well to a certain extent could, however, be insufficient in coping with the ability for changes in units of supply and demand [1, 2].

The lane of reinforcement learning (RL), which is one of the most persuasive paradigms in machine learning technology, appears to be about to revolutionize inventory management. Using RL would allow organizations to develop the niche of self-regulatory decision-making since computer-based learning would consider dynamic data and feedback as the base. This approach also opens the door to the equally enticing possibility of improving inventory level maintenance, processes, and supply chain efficiency.

In particular, this paper will give a detailed insight into using RL in this field of study. Making use of a detailed critical review of the existing literature and cases from the field, our major objective is to find out whether or not the potential of supply chain management to use these algorithms in solving the multidimensional obstacles of inventory control is indeed there.

Through our investigation of the cornerstones, uses and orders, and consequences of RL in this realm, the aim is to bring illumination to the impact of this cutting-edge method on supply chain management, which can be a pass way for firms to overcome their inefficiency, resilience, and competitiveness problems.

The main focus of this work is to address the difficulty in predicting optimal inventory management and order timings in a dynamic environment. This can lead to inefficient stock replenishment. The main objective is to leverage the RL techniques, especially DQN, to optimize inventory management by cost minimization and stockout handling while adapting to changes in demand patterns.

## 2. LITERATURE REVIEW

As a result of a survey, among the deep reinforcement learning (DRL) algorithms compared, proximal policy optimization (PPO) turns out to be more adaptable to different topologies and configurations in the supply chain inventory management (SCIM) environment, therefore having constantly higher average profits. The major drawback is that, in the worst-case scenario (1P3W), PPO does not produce positive profits. The VPG rarely converges to a global maximum, and the local maxima are somewhat farther from PPO, as the number of warehouses increases, but results are not bad. The A3C (asynchronous advantage actor-critic) algorithm has repeatedly shown the highest speed among the methods under study [1, 3].

Reward shaping potential-based is a simple and effective way to change the incentive structure of an MDP without modifying its optimal policy. Our study has shown that potentiate reward shaping can improve the performance of the

DRL algorithms that handle inventories by transferring knowledge from the heuristic inventory regulations. By leveraging the stock- and BSP-low-EW policies as teachers, one can improve the efficiency of DRL and make the training process more stable through reward shaping. The preferred teacher's policy is often better than the unshaped DQN. In many cases, reward shaping outperforms its unshaped counterparts and sometimes even surpasses the teachers [2].

The core of Double DQN is two-fold: missed sales and multi-level inventory management. Enlarging the state space by including historical demand and inventory data results in better DDLS algorithm performance. Nevertheless, when the fixed cost is altered, the agent's ordering behavior dramatically differs. DDLS is ahead of most heuristics in limiting the losses in terms of revenue. The longer the time is given, the slower the convergence speed of the algorithm becomes [4, 5]. The paper focuses on a novel hybrid algorithm that couples reinforcement learning with Demand Driven Material Requirement Planning (DDMRP) to make better inventory control decisions. The environment is built for interaction by using the Markov Decision Process (MDP) to frame the inventory management problem and includes the elements of the DDMRP methodology. The RL algorithm, more precisely Q-learning, is applied to calculate the most appropriate time and amount of a purchase. The reward function is approached from three separate angles: inventory levels, the optimization based on the distance where inventory is relative to an optimal level of inventory, and the shaping function that considers inventory levels and distances that are relative to optimal levels of inventory [6, 7].

An investigation deals with the place of AI and ML in digital supply chain management and evolution inspired by intelligence and interpretation. [8, 9]. Deep reinforcement learning enabled by the proximal policy optimization method can act as a hub to regularize and direct the inbound and outbound flows and keep the business running in stochastic and non-stationary environments with end-to-end visibility. The proximal policy optimization method, which provides the DRL agent with the ability to remove hard-coded action space and significantly reduce the need for hyperparameter tuning, is used. The new technique is presented against the context of the base-stock strategy, which is widely employed in operations research and inventory control theory and is very popular [10].

Agents are responsible for the machine load scheduling and throughputs, improving the process for maximum returns. For example, when supplier agent gets a product delivery request from a customer, they assess the availability of their production scheduling queue and decide to accept or reject a request considering job price, delivery due date, accomplishment priorities, and expected future arrival of tasks by considering overall profit [11].

One of the RL advantages is the capability to provide near real-time responses based on current state data and integrated forecasting and optimization approaches. Nevertheless, there are issues such as processing complex multiple-agent systems and getting the best decisions in situations where there are multiple parties involved. The main research avenues are to overcome these challenges and improve the effectiveness of RL in logistics and SCM [12]. The RL approaches are endowed with the ability to handle dynamic and uncertain supply chain environments by learning and employing adaptive and situation-specific ordering techniques. The authors have explained different types of inventory problems, namely customer-driven replenishment and supplier-driven replenishment, which is responsible for determining the actions that suppliers have to follow. Furthermore, the classification of inventory-related publications is by viewpoints, instigators of players, and information exchange levels [13]. A paper that presents a theory of RL algorithms will include the fundamentals of Markov decision processes and the methods of doing RL, such as value-based, policy-based, and actor-critic methods. This paper deals with the interplay of classical control and RL. The compliments and shortcomings of applying RL in process industries will also be researched [14].

The architecture functions in a GPU-parallelized ecosystem that embraces a single warehouse and multiple stores. This system has been designed to decrease computation needs while still accommodating the complexities of the practical supply chain dynamics. The system expanded the state and action spaces of the warehouse agent into successfully finding a suitable policy, even when inventory is low. This gives the system the authority to take decisions in the context of the supply chain that has a dynamic nature. This system adopts MARL technique in a distributed fashion, with each of the supply chain agents acting based on the local data. This distributed approach is more flexible and adaptable to the change of environment, making a whole reformation is not necessary when the system needs a change [15].

In realizing and improving DRL algorithms for inventory management, major design decisions like NN architectures and hyperparameter tuning must be understood and optimized. Make sure that DRL algorithms realistically incorporate the complexity of everyday inventory systems using advanced training algorithms and enhancing training performance. Model DRL policies with interpretability by creating models that explain why certain actions are recommended and the reasons, making it an intuitive step for managers to understand and practice in the real world [16].

This mapping function is thus vital to the agent's learning process; it allows him to see the underlying patterns and relationships in the environment and, therefore, to arrive at optimal decisions related to the production, storage, and transport process. By implementing essential characteristics, the agent will be able to analyze the chain network and consume the data provided by the system so that the decision-making process will improve [17]. The major companies such as UPS and Amazon have successfully employed the RL algorithms on inventory management supplies chain. UPS, for instance, implemented an RL algorithm to design AI strategic plans and improve the efficacy of their supply chain operations. The aim is to gain AI strategic plans and increase the success rate of their supply chain operations [18].

This overview emphasizes the strengths and weaknesses of different RL algorithms applied to SCIM. PPO is powerful in different configurations of the supply chain; it faces some challenges in specific scenarios and can barely handle multi-level inventory systems. Efficiency improvement by knowledge transfer in DQN is realized, but shaping policies are required, and scalability is limited for larger systems. DDLS helps reduce losses in revenue; it suffers from shortcomings of slow convergence and sensitivity to fixed costs. When combined with DDMRP, this hybrid RL approach enhances the decision-making for inventory control, but delays happen in training as complexity increases and lacks real-world validation. For example, RL with supplier agents,

which allows for the dynamic scheduling and making of decisions, is beset by considerable limitations when the environment is highly uncertain. Multi-agent reinforcement learning can offer flexibility toward distributed supply chains but has issues with scalability and computational needs. Deep RL leverages neural networks that enhance performance and interpretability for managers, yet it is sensitive and requires careful design to handle fluctuations in unpredictable demand effectively.

## 3. REINFORCEMENT LEARNING AT A GLANCE

The typical scenario of RL indicating various components can be seen from Figure 1. The following components are the integral part of any RL application.

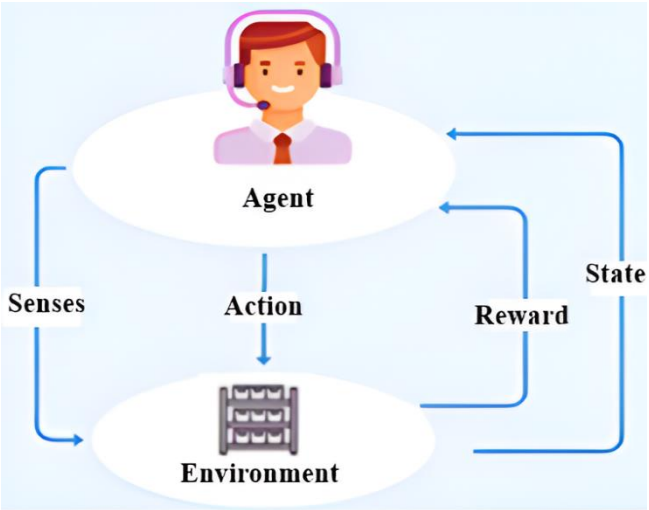


**Figure 1.** Agent, environment, and state in reinforcement learning

### 3.1 Elements of reinforcement learning

**State:** In RL, state is an objective status or internal state of the environment in which the agent works [19]. States are overarching abstractions that provide the agent with the dynamics of the environment and allow the agent to perform its functions. In the case of inventory management, states might consist of the following variables: current inventory levels, demand forecasts, lead times, and so on, producing factors that affect the managerial decision-making process. Through states, RL management agents can learn a good control policy by establishing an association between actions and states that results in desired outcomes.

**Agent:** The "agent" in RL is the entity where decision-making and action are taken inside the specific environment. In dealing with inventory management, the agent could be a software system or an algorithm that is expected to make decisions on optimal inventory level, order quantity, and other related variables to ensure maximum profitability or least cost. The agent responds to the environment by recognizing its current state, performing the actions given by the policy, and being rewarded or penalized by feedback [20]. By trial and error, the system is constantly being refined, and its strategy is adapting to achieve goals more effectively with every turn.

**Environment:** "Environment" in RL means the collection of goals, inputs, outputs, or whatever else makes up the outside environment or the problem domain with which the agent interacts. Inventory planning contains determinants like customer demand, vendor lead times, inventory limitations, and market factors, among other elements of the environment. The environment is a dynamic process that runs in time and depends on the state of the agent itself and other external factors. An agent of RL needs to manage the augmented surroundings when selecting buy-ins, which would entail lowering the costs of inventory stocks and giving customers the level of satisfaction, they deserve.

**Policy:** In RL, a "policy" refers to an agent's strategy based on the observed states of the environment and involves selecting actions. Policies can be deterministic or stochastic, where in the former, the actions are specified for a fixed state, while stochastic governs a probability distribution over the possible actions. RL aims to discover a policy that produces the highest summed rewards throughout the learning process [21]. In inventory management, this policy will provide the agent with the rules to decide when to order the new stock, how much to order, and how to distribute the current stock to meet the client's demand while minimizing expenses.

**Algorithms:** RL algorithms function as a basis of the computational environment, where the agents learn how to choose the optimal policies using interaction with the environment. These algorithms are cataloged by several classes, such as value-based methods (e.g., Q-learning), policy-related methods (e.g., policy gradients), and actor-critic methods (e.g., DQN). Each algorithm has pros and cons, so using them in different problems and conditions would help achieve the best results. RL algorithms help agents perform experiential learning, quickly adapt to changing conditions, and make sound inventory-related decisions to maximize the fulfillment of desired objectives.

### 3.2 Types of reinforcement learning

Figure 2 presents the taxonomy of reinforcement learning algorithms.

**(1) Model-free reinforcement learning**

In model-free reinforcement learning, the agents learn to obtain the maximum optimal policies by merely interacting with the environment without developing a model representing its dynamics. Such a method relies on its success on the circumstances in which the environment behaves in a complex way or is unpredictable. Regarding inventory management, model-free RL permits agents to acquire from experience via trial and error with the ultimate aim of developing the most suitable control strategies for performing this task successfully through exploitation and exploration. Unlike model-based RL agents that work using devised models, flexible model-free RL agents can update their policy based on current performance. Hence, they can respond to Unknown supply chain disruptions like demand changes and supply delay variations.

**(2) Model-based reinforcement learning**

Model-based reinforcement learning represents learning a model explicit to the system, which the agent employs in planning and decision-making. The difference between model-free RL, which solely works with the interaction data, and model-based RL, which incorporates the model of how the environment responds to actions as either empirical or predefined, is that the model-based RL incorporates the model. Model-based RL imbues the agents with the ability to visualize alternative situations, apply the what-if approach, and assess consequences, all in a virtual space, before executing any

action in the real world [22]. The AI agent can be trained to associate a model of inventory dynamics with the outcome of decision-making. In this way, agents can accurately predict the possible consequences and better use inventory policies, limiting the effects of uncertainty.

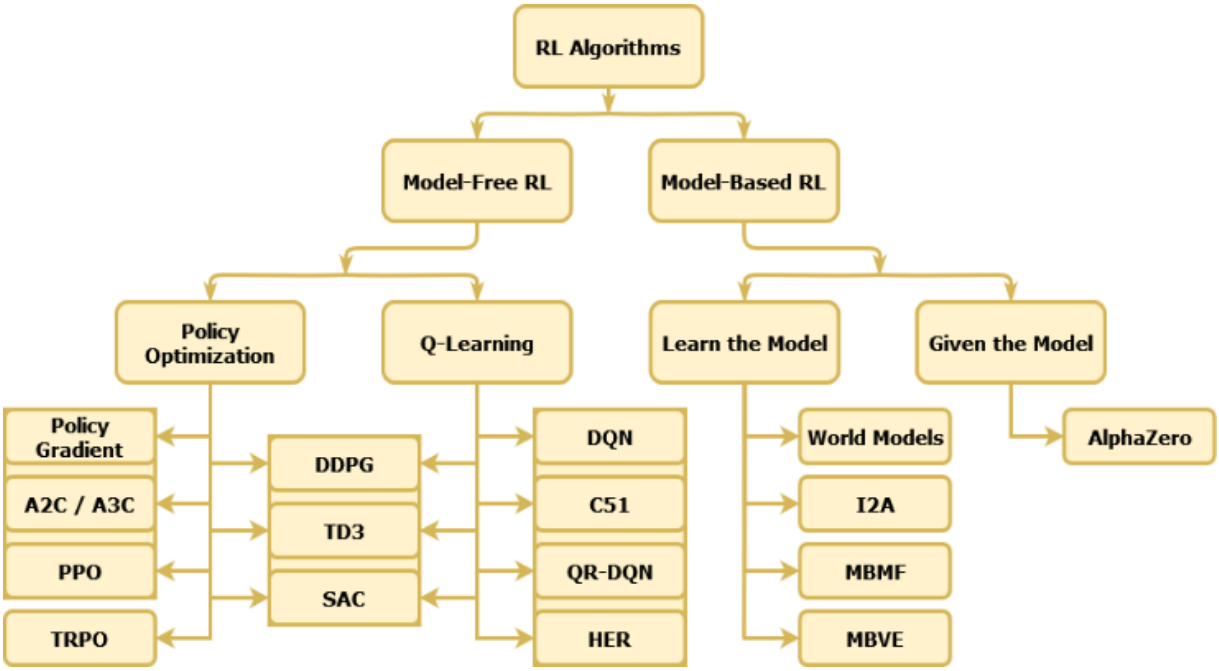


**Figure 2.** Taxonomy of reinforcement learning

**(3) On-policy reinforcement learning**

On-policy RL algorithms learn from the data while the policy is updated. This implies that the agent can learn from its actions and experiences, thereby making small adjustments to its decision-making to get better with experience [22]. Inventory management implementations reference on-policy RL agents that exploit the environment actively, collecting data on demand, lead times, and inventory levels to fine-tune their strategies. Besides, the policy adjustment of on-policy RL agents is the change of parameters according to the current benefits and feedback. In turn, on-policy RL agents can deal with changing environments, and they can make faster and more rational decisions in dynamic business lines.

**(4) Off-policy reinforcement learning**

The off-policy RL enables the algorithms to no longer join the policy for exploration with the policy for learning. In doing that, an agent behaves like a data-generating process created by different policies and uses it by sampling and exploring [23]. The role of this process can be highly efficient compared to other processes. Inventory management off-policy RL learns compared to the previous historical analysis and other exploration strategies, which helps to become better explorers. Agents of off-policy RL having the exploration/exploitation decoupling are more efficient; they keep the right balance between acquiring knowledge about the system and using it for policy execution. Thus, they enable the achievement of more robust and adaptive inventory control policies.

**(5) Deep Q network**

DQN is considered RL algorithm with a strong attribute of deep learning and Q-function approximation for the optimal action-value function. In a GM environment, DQN involves producing agents that try to learn optimal control policy only from raw data like inventory level and demand forecasts. By using neural networks to approximate Q-values, a DQN agent can handle complex optimum situations with large states and action spaces comprehensively; thus, this DQN framework can be applied to an inventory management situation that contains multifaceted input features.

**(6) Dyna Q architecture**

Dyna-Q is an interesting algorithm because it brings together the strong points of the model-based and model-free approaches of reinforcement learning. At the core of Dyna-Q is a model of the world that is used to simulate future scenarios and through which the agent can learn from both observation and experience. This feature makes Dyna-Q use its limited experience wisely; it modifies the worth function based on real and imagined observations. Dyna-Q now became able to systematically generalize the knowledge across the same states and actions, which resulted in better decision-making skills. In addition, Dyna-Q enables the multi-armed bandit approach by balancing the trade-off between exploration and exploitation to enable the agents to navigate complex and uncertain environments. The overall Dyna-Q is a novel research direction in RL with an attractive structure that handles plenty of real-life issues.

**(7) Actor-critic methods**

Actor-critic methods use both actor methods that are policy-based (actor) and critic methods that are value-based (critic). This is a good combination that allows the machine to explore and exploit. In this case, actor-critic algorithms need to learn a policy function (which acts according to the selected actions) and a value function (which evaluates the quality of selected actions). The simultaneous updating of both functions by the actor-critic agents achieves higher efficiency and better performance over other single-model approaches. This effectiveness of actor-critic methods arises from the fact that they are appropriate for inventory control tasks characterized by adaptive and dynamic decision-making in unpredictable environments.

**(8) Monte Carlo methods**

Monte Carlo methods are types of learning algorithms that guess value functions by using the average of sampled returns

from simulated paths. In managing stock, Monte Carlo methods help agents find the best rules by creating many rounds of play with the setting and figuring out returns from seen rewards [23]. By taking the average of returns from many paths, Monte Carlo agents can guess state and action values better, leading to smarter choices in changing and unsure supply chain situations.

**(9) Temporal difference (TD) learning**

TD learning is a key building block in reinforcement learning that allows an agent to learn from sequential inputs by stitching together rewards that are expected to come in the future [23]. TD learning algorithms that choose values based on the difference between the current and the estimated future rewards can record the experience and improve the policies as a learning agent. The fact that temporal difference updates introduce exploitation-exploration balance makes TD learning a technique that has been so frequently used in inventory control applications.

**(10) Proximal policy optimization**

PPO is an outstanding algorithmic method that makes learners from trial and error more sample-friendly with each training and, finally, sustainable regarding stock management. PPO algorithms refine the playing direction singularly or severally but don't exceed what a professional gamer would do to obtain as much reward while not getting the changes too big to keep the learning stable [24]. The policy of PPO aids this by fine-tuning the decisions with small steps up and simultaneously making sure that policy shifts are monitored and controlled. The main result is solid and firm plans for those helping in inventory management. This is an example because, in the same meaning, real-life supply networks must be very stable, thus leading to efficient working.

## 4. REAL-WORLD APPLICATIONS

The area of inventory management and other supply chain management (SCM) domains, among many, is covered by reinforcement learning, which offers novel solutions to the complicated problems of decision-making [25]. In the practice of stock management, RL algorithms have the potential to establish inventory control policies flexibly by updating the ordering, stocking, and fulfillment procedures in real time, which in turn helps to minimize costs while covering customers' demands. RL models, in turn, can deal with the changing demand patterns, the disruptions of the supply chain, and the limits of the inventory, leading businesses to become more dynamic and responsive. Beyond inventory management, RL holds promise in various SCM areas, including:

**(1) Supply chain optimization**

RL algorithms perform unit optimization in supply chain operations: dynamic resource allocation, resource management, production schedule optimization, and transportation logistics optimization. Real-time data and feedback are the basis upon which RL models can develop all supply chain strategies, which could then be applied to reduce costs, decrease lead time, or improve service levels [26].

**(2) Demand forecasting**

RL-based demand forecasting models can analyze the sales history, market trends, and environmental factors to achieve accurate demand predictions [27]. They can even adjust forecasts on a dynamic basis depending on the changing market conditions, which, in turn, improve inventory management, production planning, and distribution strategies.

**(3) Warehouse management**

RL techniques can be leveraged in warehouse management, which includes controlling the re-supply of stock and picking and packing operations. RL-based warehousing management systems could help to improve the efficiency of order realization and accuracy of tasks in the chain and provide optimal storage utilization, which could result in better order processing and delighted customers.

**(4) Supply chain risk management**

RL-based algorithms mitigate supply chain risks with the help of identification processes, which can predict potential disruptions and optimize risk mitigation strategies in the process [26]. Through learning from past mistakes and adjusting emerging risk factors, RL has the potential to upgrade supply chain resilience and lessen the disturbance of operations from disruptions.

**(5) Transportation management**

RL-based transportation management systems maximize route designing, dispatching, and fleet operating tasks to decrease transportation costs and delivery performance. The capabilities of RL models to learn from historical data as well as real-time traffic conditions can assist them in routing optimization, fuel saving and transportation efficiency enhancement.

## 5. CASE STUDY PROBLEM STATEMENT

### 5.1 Pre-work based on literature review

When addressing the competitive choice of algorithms for SCIM, which is particularly important, one has to consider strengths and limitations with respect to the particular characteristics of the problem. In general, SCIM involves discrete decision-making, for example, the amount of stock order, the frequency of customer demand, and the need for fast, effective actions in large state spaces. Among the different algorithms reviewed, DQN was chosen for this study because it seems uniquely appropriate for such situations. DQN is especially apt for SCIM since it is a reinforcement learning algorithm that optimizes in a discrete action and reward environment for decisions such as when or how much stock to order. Again, Q-value approximation enables the model to learn an effective inventory policy cost-efficiently.

In this case, DQN balanced the trade-off between inventory holding costs and stockout risks in fluctuating demand. The stabilizing of learning aided by using experience replay with target networks provides a robust algorithm applicable to environments requiring precise control over actions with immediate and long-term consequences, such as those in this paper. In contrast, PPO and A3C algorithms usually operate under continuous action spaces and stochastic environments. These make them less than ideal for SCIM, which is discrete in nature. Particularly, PPO, though efficient in generalization across various environments, converges poorly under discrete decision-making problems such as inventory control. The slower learning process, combined with the complexity of tuning hyperparameters, makes it less than ideal for small and medium retail settings, which need to make quick and effective decisions. Similarly, the main advantage of A3C is that it gives fast, on-policy updates that could receive huge advantages in certain environments. Still, regarding an inventory system, sensitivity to sparse rewards and large state spaces poses difficulties. The computational expense of the

algorithm makes it less practical for real-time applications in SCIM when the speed of decision-making is highly critical.

Additionally, vanilla policy gradient (VPG), though simple to implement, is not well-suited for the complexities of inventory management. Its high variance and sample inefficiency make it better suited for simpler environments than complex multi-agent systems like supply chains. Overall, DQN is the best algorithm choice for SCIM in this research work. In a nutshell, while dealing with large states and discrete actions, it efficiently learns the policy that needs to be decided in a dynamic inventory control system. The experimentation shows that DQN outperformed other RL algorithms because of the optimal determination of inventory policies, which can be chosen based on certain motivating factors relative to such a case study.

Experimenting with real historical sales, datasets showed higher effectiveness, consumer satisfaction, and revenue generation with the recommended technology method compared with the present ones. Particularly, the recommended strategy is to yield the legacy metrics of <5 and 5% more inventory turnover rate and earnings than the heuristic method, respectively [15].

Inventory management at the supply chain level is all about optimizing stock levels in order to meet customer demands effectively. Conventionally, static decision rules in inventory control policies like (R, Q), (T, S), and (s, S) are based on continuous demand patterns, as shown in Figure 3, that do not respond to dynamic conditions as well as they are predicted to do. This is because such policies can very often lead to suboptimal management of the inventory. Thus, the company may either lose some sales or incur additional inventory costs. To overcome this shortcoming, this case study focuses on the RL method that is DQN and it tries to create a dynamic inventory control policy that is suitable for a small retail shop that deals with Coke. Through RL application, this work aims to produce dynamic inventory arrangements to create greater profits than those of traditional static inventory policies.

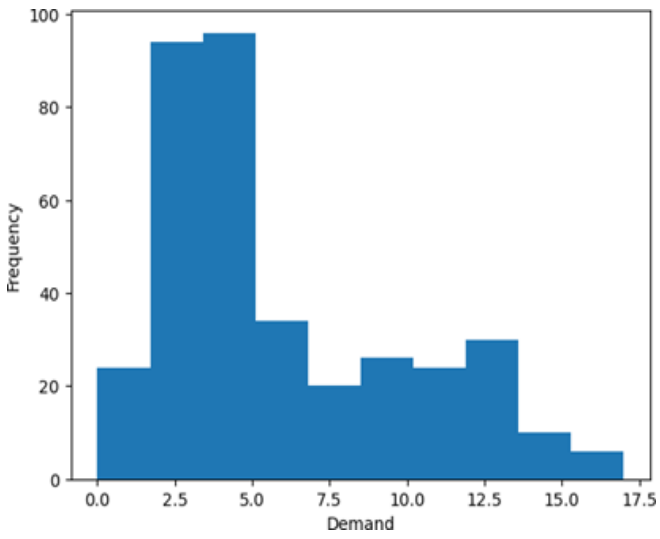


**Figure 3.** Environment for case-study

### 5.2 Classic inventory control policies

Traditional inventory control policies of (R, Q), (T, S), (S, s), and base stock have been the norm for inventory replenishment based on fixed decision rules. Although these policies are adequate for a limited demand category, they are not flexible enough to fit into the changing nature of the market condition. The shortcomings of such static practices point towards a proactive and dynamic approach to inventory administration.

### 5.3 Reinforcement learning for inventory optimization

The RL approach offers a promising concept for building dynamic modeling of inventory control policies that can adjust to fluctuations in demand. MDP, called Markov Decision Process (MDP), enables RL to learn the optimal decision-making strategies for agents within the environment through interaction. In the presented case study, the state, action, and reward which are the main components of the MDP as a means of modeling the inventory problem in the most effective way.

### 5.4 Solving the Markov decision process

Because model-free RL methods are based on reality and are therefore practical, the optimal solution of the designed MDP can be found by using these methods. In this regard, DQN, a variant of Q-learning, comes to the fore. Deep neural networks are used in DQN to approximate the Q-function, which gives rise to the efficient learning of the optimal control policies. Our main focus is the implementation of DQN. This learning tool is responsible for producing an adaptive inventory control policy, which computes and executes the optimum ordering activity based on real-time inventory position and future demand information, thus leading the retail store to profit maximization.

### 5.5 Case study experimenting

In a numerical let us introduce an experiment in which Classic control policies will be compared to policies learned by DQN for a hypothetical small retail store specializes in the sales of Coke. Focusing on this experiment, an assessment of the strategic decision-making process regarding inventory replenishment to meet the customer demand is focused. Inventory replenishment in a store involves ordering Coke cases, which is integer quantity and each contains 24 cans. Setting the context, the key financial parameters such as: a unit selling price of $30 a case, a holding cost of $3 a case a night, a fixed ordering cost of $50 an order, and a variable ordering cost of $10 a case are established. Moreover, some operational constraints such as inventory capacity of 50 cases, a maximum order quantity of 20 cases per order, an initial inventory of 25 cases at the end of Sunday, and 2 days for order fulfilment lead time are established.

In this experiment, the demand patterns based on a predefined structure: demand for Monday to Thursday will have a normal distribution N (3, 1.5), demand for Friday will have N (6, 1), and demand from Saturday to Sunday will have N (12, 2) are simulated. Through this, 52 samples of past data for 1 year to be used as a training dataset for the DQN model are obtained. As a yardstick, to tune the (s, S) inventory control policy, the same data collection as that used to train the DQN model is utilized. The test set scenario is the next step that involves DQN-learned policy and benchmark policy comparison. The goal of the analysis is to get knowledge on the comparative ability of the dynamic RL-based inventory control strategies and the traditional static inventory management, making it possible to give useful recommendations on supply chain optimization and

operational decisions in retailing operations.

### 5.6 Case study methodology

The given study applies DQN to designing a dynamic inventory control policy for a small retail store selling Coke. The implementation of DQN was necessary because static inventory control policies, such as (s, S), are rather ineffective in fluctuating demand conditions. The following methodology focuses on creating, through reinforcement learning, a flexible real-time decision-making model. The environment is modeled as a Markov Decision Process in which, at every given point in time, the state will be the current inventory, and based on that state, the agent chooses an action number of Coke cases to order earnings are considered by calculating rewards using profitability. Key financial factors determining profitability include holding costs, ordering costs, or revenue from sales. Its DQN uses a neural network structure with an input layer, two hidden layers of 64 neurons each, and ReLU as the activation function in the hidden layers. The network will be trained using the Adam optimizer with a learning rate of 0.001, while the discount factor $\gamma$ was chosen to be 0.95 to balance immediate and future rewards. Finally, $\varepsilon$-greedy exploration is used, where $\varepsilon$ starts at 1 and exponentially decays down to 0.1 to ensure enough exploration during training.

The data used is the simulated demand patterns of a 52-week period. The weekday demand is normally distributed by parameters N(3, 1.5), the Friday demand is N(6, 1), and the weekend demand is N(12, 2). The given financial parameters of the case study were: Unit selling price is $30; Holding cost per case per night is $ 3; Fixed ordering cost is $ 50; Variable ordering cost is $ 10/case. The inventory capacity is limited to 50 cases with a maximum order quantity of 20 cases. This is further put to the test in the performance against the classic, widely used (s, S) inventory policy across key metrics: inventory turnover rate, stockout frequency, holding costs, total revenue, and overall profitability. Quantitative analysis may be provided regarding comparative weekly costs, profit, and the number of stockout occurrences, supported by additional statistical tests confirming the significance of the differences between DQN and static control policies. This complete setup gives clarity on the RL algorithm that has to be implemented, the training data, and the metrics involved in the evaluation. Further, analysis with full details shows the influence of the DQN model in improving inventory management.

In the paper, we are supposed to use the DQN algorithm to optimize the inventory control decision. The DQN consists of two fully connected hidden layers of 128 neurons each, followed by a final layer with a number of output units corresponding to the action values, while the state space, with the representation of the inventory position and binary encoding of the day of the week, maps into Q-values over actions. The DQN is trained on a replay buffer of 500000 experiences, with a batch size of 128. The optimizer used is Adam, which has a learning rate of 0.0001 while performing a soft update with a factor of 0.001 to ensure gradual updates of the target network. The exploration-exploitation trade-off is managed by an epsilon-greedy policy decaying the epsilon value from 1.0 to 0.01 over episodes. The model updates every four steps, and we train it over 1,000 episodes, trying to maximize total rewards, reflecting profit from selling units while minimizing holding and ordering costs.

### 5.7 Experimenting with different cases

These cases thus include eight sets of variants in the demand distributions for different days of the week, either over 52 weeks (1 year) or over 104 weeks (2 years).

Case 1: The demand from Monday to Thursday is normally distributed with a mean of 3 and a standard deviation of 1.5. On Fridays, the demand distribution is normal, with a mean of 6 and a standard deviation of 1. In contrast, on weekends-i.e., Saturday and Sunday-the demand is normally distributed with a mean of 12 and a standard deviation of 2.

Case 2: The demands from Monday to Thursday are normally distributed with mean 2 and standard deviation 1. Each Friday, demand is normally distributed with a mean of 4 and a standard deviation of 2. The demand is normally distributed on Saturdays and Sundays with a means of 8 and a standard deviation of 4.

Case 3: Monday to Thursday, demand distribution is Normal with a mean of 0 and a standard deviation of 1. On Fridays, demand distribution is Normal, with a mean of 1 and a standard deviation of 2. Weekend demand distribution is Normal, with a mean of 3 and a standard deviation of 4.

Case 4: From Monday to Thursday, the demand is normally distributed with a mean of 1 and a very high standard deviation of 10, reflecting higher uncertainty. The demand distribution is normal on Fridays, with a mean of 3 and a standard deviation of 15. Lastly, for Saturday and Sunday, the demand is normally distributed with a mean of 7 and a high standard deviation of 20.

The analysis for each case is for two-time sets: one set for 52 weeks, which equals a year; another set for 104 weeks, or two years.

### 5.8 Discussions and findings

While it has become obvious through the case study that there are advantages in applying RL for optimizing inventory management, several practical considerations remain to be made when using these methods in realistic supply chains. In particular, data availability and quality: RL algorithms need data that is accurate and of high quality to train the algorithm, which may not always be accessible in realistic environments. Moreover, the computational burden of training RL models, especially DQN, often overwhelms small businesses due to the lack of advanced computing resources. Other limitations include scalability: this existing work considers a single retailer selling stock of one product in real-world applications where the demand is high and greatly variable for each product and location. Also, the integration of the RL models with currently established supply chain management systems tacitly requires real-time data exchange, which may introduce other operational problems.

These challenges notwithstanding, key takeaways from this study present RL-based inventory systems as those that can offer considerable improvements in performance compared to traditional static approaches. Our case also proves that the RL-driven approach can be effective in dynamic decision-making and demand forecasting, returning a profit of $17,202.08. Table 1 presents Experimental results for inventory management with varying demand patterns. Figures 4-11 represent episodic rewards for inventory management with varying demand patterns. These findings show that RL has the potential to improve inventory turnover rates, decrease stockouts, and increase profitability; thus, it is a promising

direction in supply chain optimization.

**Table 1.** Experimental results for inventory management with varying demand patterns

| Case | Weeks | Distribution | +ve Reward | Results |
|---|---|---|---|---|
| 1 | 52 | M-Th =N (3, 1.5)<br>F =N (6, 1)<br>S-Su =N (12, 2) | 300 Epi | $17,202.08 |
| 2 | 52 | M-Th =N (2, 1)<br>F =N (4, 2)<br>S-Su =N (8, 4) | 323 Epi | 10165.18 |
| 3 | 52 | M-Th =N (0, 1)<br>F =N (1, 2)<br>S-Su =N (3, 4) | 643 Epi | 407.08 |
| 4 | 52 | M-Th =N (1, 10)<br>F =N (3, 15)<br>S-Su =N (7, 20) | 214 Epi | 11866.61 |
| 5 | 104 | M-Th =N (3, 1.5)<br>F =N (6, 1)<br>S-Su =N (12, 2) | 123 Epi | 42416.72 |
| 6 | 104 | M-Th =N (2, 1)<br>F =N (4, 2)<br>S-Su =N (8, 4) | 300 Epi | 16205.09 |
| 7 | 104 | M-Th =N (0, 1)<br>F =N (1, 2)<br>S-Su =N (3, 4) | 840 Epi | -116.88 |
| 8 | 104 | M-Th =N (1, 10)<br>F =N (3, 15)<br>S-Su =N (7, 20) | 206 Epi | 24065.6 |

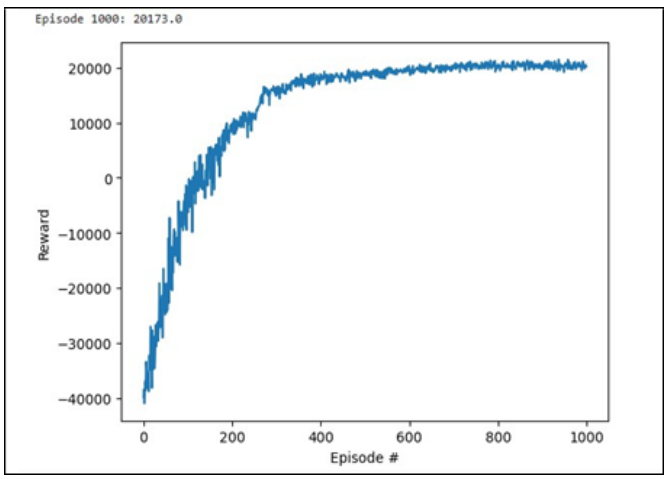


**Figure 4.** Episodic rewards for Case 1

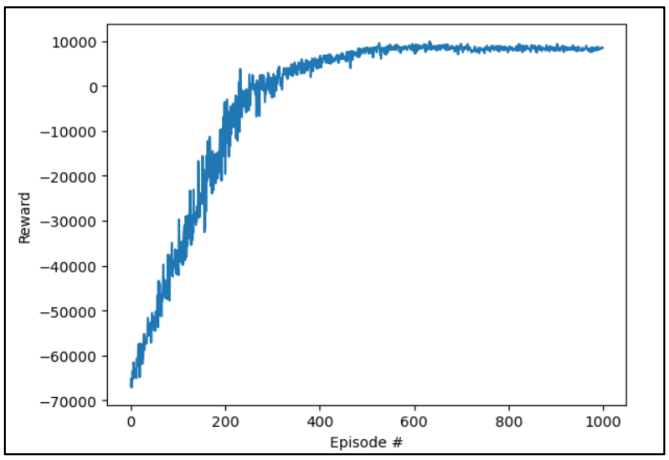


**Figure 5.** Episodic rewards for Case 2

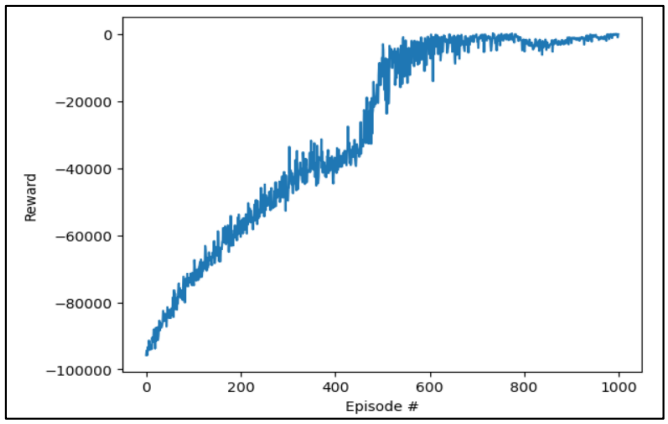


**Figure 6.** Episodic rewards for Case 3

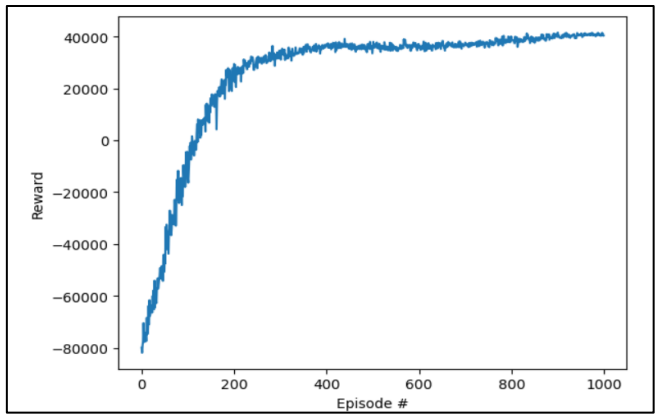


**Figure 7.** Episodic rewards for Case 4

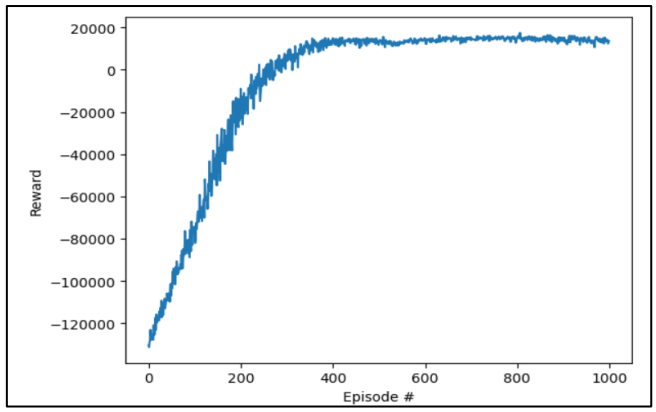


**Figure 8.** Episodic rewards for Case 5

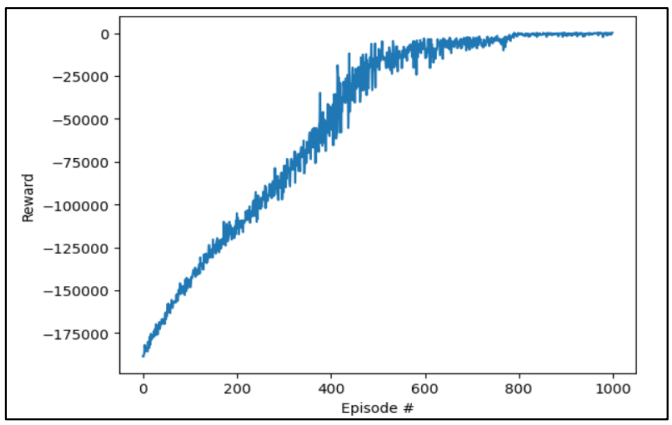


**Figure 9.** Episodic rewards for Case 6

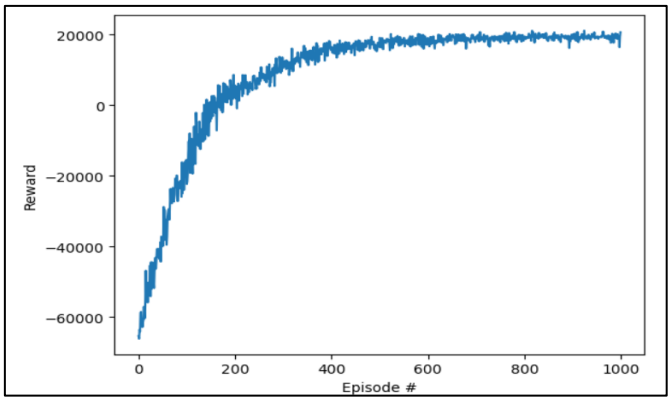


**Figure 10.** Episodic rewards for Case 7

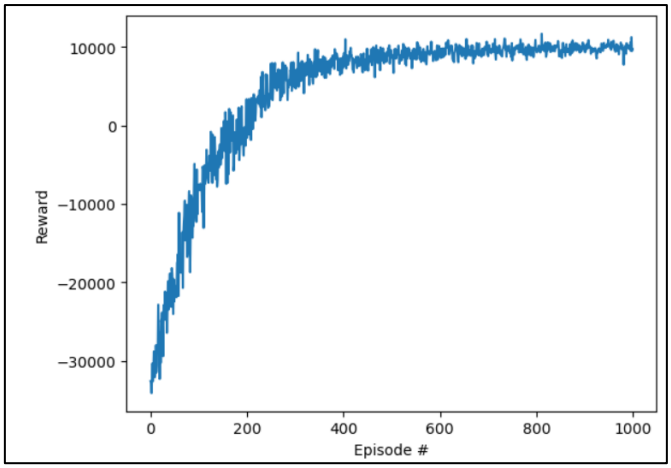


**Figure 11.** Episodic rewards for Case 8

Future research should be done to allow the scaling of RL models for multiple products and retailers, as well as testing with real-world data in live operation environments to prove their performance. The examination of other RL policies can also be performed, such as the base stock policy or hybrid approaches, which may yield further optimization on diverse retail settings. Improvements in model convergence speed and efficiency in learning will finally be done to seriously allow the big-time involvement of RL in real time for inventory management systems.

## 6. CONCLUSION AND FUTURE SCOPE

In the end, it is proved that inventory management through reinforcement learning is a profitable direction. Through the featuring of (s, S) policies, the supply chain performance level can be improved. The achieved income of $17,202.08 reaffirms the power of RL in ensuring economic inventory control policies. This success is the best evidence that RL is a very useful method to take into consideration the complexity and uncertainty that exist in supply chain management. Moving into the future, more research and development in inventory optimization using RL can potentially improve profitability, minimize cost, and improve all-around supply chain performance. Along with how fast technology is progressing, the inclusion of RL into supply chain systems would be a means of achieving greater efficiency and competitiveness in the marketplace. This case study focuses on a single retailer and one product in the inventory; this introduces a need to work with multiple products and inventory levels. Real-world data might also be utilized to verify DRL algorithms and determine whether they improve the performance of existing SCIM systems in use. It could have different policies like base stock policy and updated policies for convergence and learning Q-table.

## ACKNOWLEDGEMENTS

This work was supported by the Research Support Fund (RSF) of Symbiosis International (Deemed University), Pune, India.

## REFERENCES

[1] Stranieri, F., Stella, F. (2022). A deep reinforcement learning approach to supply chain inventory management. arXiv Preprint arXiv: 2204.09603.

[2] De Moor, B.J., Gijsbrechts, J., Boute, R.N. (2022). Reward shaping to improve the performance of deep reinforcement learning in perishable inventory management. European Journal of Operational Research, 301(2): 535-545. https://doi.org/10.1016/j.ejor.2021.10.045

[3] Babaeizadeh, M., Frosio, I., Tyree, S., Clemons, J., Kautz, J. (2016). Reinforcement learning through asynchronous advantage actor-critic on a GPU. arXiv Preprint arXiv: 1611.06256. https://doi.org/10.48550/arXiv.1611.06256

[4] Wang, Q., Peng, Y., Yang, Y. (2022). Solving inventory management problems through deep reinforcement learning. Journal of Systems Science and Systems Engineering, 31(6): 677-689. https://doi.org/10.1007/s11518-022-5544-6

[5] Hu, Y., Zhao, Y., Feng, Y., Ma, X. (2024). Double DQN method for botnet traffic detection system. Computers, Materials & Continua, 79(1): 509-530. https://doi.org/10.32604/cmc.2024.042216

[6] Cuartas, C., Aguilar, J. (2023). Hybrid algorithm based on reinforcement learning for smart inventory management. Journal of intelligent manufacturing, 34(1): 123-149. https://doi.org/10.1007/s10845-022-01982-5

[7] Cherednichenko, O., Vovk, M., Ivashchenko, O., Baggia, A., Stratiienko, N. (2021). Improving item searching on trading platform based on reinforcement learning approach. In COLINS, pp. 1444-1455.

[8] Rana, J., Daultani, Y. (2023). Mapping the role and impact of artificial intelligence and machine learning applications in supply chain digital transformation: A bibliometric analysis. Operations Management Research, 16(4): 1641-1666. https://doi.org/10.1007/s12063-022-00335-y

[9] Prudencio, R.F., Maximo, M.R., Colombini, E.L. (2023). A survey on offline reinforcement learning: Taxonomy, review, and open problems. IEEE Transactions on Neural Networks and Learning Systems, 35(8): 10237-10257. https://doi.org/10.1109/TNNLS.2023.3250269

[10] Kegenbekov, Z., Jackson, I. (2021). Adaptive supply chain: Demand-supply synchronization using deep reinforcement learning. Algorithms, 14(8): 240. https://doi.org/10.3390/a14080240

[11] Stockheim, T., Schwind, M., Koenig, W. (2003). A reinforcement learning approach for supply chain

management. In 1st European Workshop on Multi-Agent Systems, Oxford, UK.
[12] Yan, Y., Chow, A.H., Ho, C.P., Kuo, Y.H., Wu, Q., Ying, C. (2022). Reinforcement learning for logistics and supply chain management: Methodologies, state of the art, and future opportunities. Transportation Research Part E: Logistics and Transportation Review, 162: 102712. https://doi.org/10.1016/j.tre.2022.102712
[13] Rolf, B., Jackson, I., Müller, M., Lang, S., Reggelin, T., Ivanov, D. (2023). A review on reinforcement learning algorithms and applications in supply chain management. International Journal of Production Research, 61(20): 7151-7179. https://doi.org/10.1080/00207543.2022.2140221
[14] Dogru, O., Xie, J., Prakash, O., Chiplunkar, R., Soesanto, J., Chen, H., Velswamy, K., Ibrahim, F., Huang, B. (2024). Reinforcement learning in process industries: Review and perspective. IEEE/CAA Journal of Automatica Sinica, 11(2): 283-300. https://doi.org/10.1109/JAS.2024.124227
[15] Khirwar, M., Gurumoorthy, K.S., Jain, A.A., Manchenahally, S. (2023). Cooperative multi-agent reinforcement learning for inventory management. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Cham: Springer Nature Switzerland, pp. 619-634. https://doi.org/10.1007/978-3-031-43427-3_37
[16] Boute, R.N., Gijsbrechts, J., Van Jaarsveld, W., Vanvuchelen, N. (2022). Deep reinforcement learning for inventory control: A roadmap. European Journal of Operational Research, 298(2): 401-412. https://doi.org/10.1016/j.ejor.2021.07.016
[17] Kemmer, L., von Kleist, H., de Rochebouët, D., Tziortziotis, N., Read, J. (2018). Reinforcement learning for supply chain optimization. In European Workshop on Reinforcement Learning 14, pp. 1-9.
[18] D'Souza, S. (2021). Implementing reinforcement learning algorithms in retail supply chains with OpenAI Gym toolkit. arXiv Preprint arXiv: 2104.14398. https://doi.org/10.48550/arXiv.2104.14398
[19] Thrun, S., Schwartz, A. (1994). Finding structure in reinforcement learning. Advances in Neural Information Processing Systems, 7.
[20] Mousavi, S.S., Schukat, M., Howley, E. (2018). Deep reinforcement learning: An overview. In Proceedings of SAI Intelligent Systems Conference (IntelliSys), pp. 426-440. https://doi.org/10.1007/978-3-319-56991-8_32
[21] Demizu, T., Fukazawa, Y., Morita, H. (2023). Inventory management of new products in retailers using model-based deep reinforcement learning. Expert Systems with Applications, 229: 120256. https://doi.org/10.1016/j.eswa.2023.120256
[22] Moerland, T.M., Broekens, J., Plaat, A., Jonker, C.M. (2023). Model-based reinforcement learning: A survey. Foundations and Trends® in Machine Learning, 16(1): 1-118. http://doi.org/10.1561/2200000086
[23] Kaelbling, L.P., Littman, M.L., Moore, A.W. (1996). Reinforcement learning: A survey. Journal of Artificial Intelligence Research, 4: 237-285. https://doi.org/10.1613/jair.301
[24] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O. (2017). Proximal policy optimization algorithms. arXiv Preprint arXiv: 1707.06347. https://doi.org/10.48550/arXiv.1707.06347
[25] Giannoccaro, I., Pontrandolfo, P. (2002). Inventory management in supply chains: A reinforcement learning approach. International Journal of Production Economics, 78(2): 153-161. https://doi.org/10.1016/S0925-5273(00)00156-0
[26] Aamer, A., Eka Yani, L., Alan Priyatna, I. (2020). Data analytics in the supply chain management: Review of machine learning applications in demand forecasting. Operations and Supply Chain Management: An International Journal, 14(1): 1-13 http://doi.org/10.31387/oscm0440281
[27] Chopra, S., Meindl, P. (2022). Supply Chain Management: Strategy, Planning, and Operation (8th ed.). Pearson.