A HyAR Algorithm for Dynamic Order Approval in China Energy Railway Freight System

Meng Wang[1,2], Renjie Liu[1], Jinshan Pan[1,3,4], Shaoquan Ni[1,3,4] and Jingchun Geng[5,*]

[1] School of Transportation and Logistics, Southwest Jiaotong University, Chengdu, China

[2] China Energy Railway Equipment Co., Ltd., Beijing, China

[3] National and Local Joint Engineering Laboratory of Comprehensive Intelligent Transportation, Southwest Jiaotong University, Chengdu, China

[4] National Engineering Laboratory of Integrated Transportation Big Data Application Technology, Southwest Jiaotong University, Chengdu, China

[5] China Railway Economic and Planning Research Institute, Beijing, China

INFORMATION

Keywords:
Heavy-haul railway
order approval
client value
reinforcement learning
HyAR

DOI: 10.23967/j.rimni.2026.10.73703

SCIPEDIA Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería

Revista Internacional Métodos numéricos para cálculo y diseño en ingeniería RIMNI in cooperation with CIMNE

A HyAR Algorithm for Dynamic Order Approval in China Energy Railway Freight System

Meng Wang[1,2], Renjie Liu[1], Jinshan Pan[1,3,4], Shaoquan Ni[1,3,4] and Jingchun Geng[5,*]

[1]School of Transportation and Logistics, Southwest Jiaotong University, Chengdu, China

[2]China Energy Railway Equipment Co., Ltd., Beijing, China

[3]National and Local Joint Engineering Laboratory of Comprehensive Intelligent Transportation, Southwest Jiaotong University, Chengdu, China

[4]National Engineering Laboratory of Integrated Transportation Big Data Application Technology, Southwest Jiaotong University, Chengdu, China

[5]China Railway Economic and Planning Research Institute, Beijing, China

ABSTRACT

As a heavy-haul railway, China Energy Railway mainly transports coal and has the characteristics of stable supply and large freight volume. The order approval process follows a centralized and unified model. This approach suffers from prolonged approval cycles, extended intervals between order submission and actual transportation (with submissions required a year in advance), and vague planned delivery times, making it difficult to meet clients' demands for precise delivery timelines and the flexibility required for high-value scattered cargo orders. To stabilize long-term revenue for transport enterprises and enhance client satisfaction, this study introduces a client value coefficient and a delivery-time satisfaction function to evaluate order value. A dynamic order approval model for China Energy Railway freight services is constructed and solved using a deep reinforcement learning algorithm. Given the large volume of orders and the complexity of order requirements in China Energy Railway, which involve multiple auxiliary decision variables, some discrete decision variables are adjusted to continuous variables to accelerate model training. This adjustment is combined with the HyAR algorithm to enhance the training efficiency of intelligent agents. Finally, the model's performance is tested using freight data from China Energy Railway in March 2024. Under constrained capacity conditions, the dynamic order approval model achieves 3.1% improvement in comprehensive revenue compared to static approval methods.

OPEN ACCESS

Received: 23/09/2025

Accepted: 24/11/2025

DOI
10.23967/j.rimni.2026.10.73703

Keywords:
Heavy-haul railway
order approval
client value
reinforcement learning
HyAR

1 Introduction

The China Energy Railway (CER) implements an annual order approval model, requiring shippers to submit transportation orders for the following year by November of the preceding year. During the approval process, all orders are incorporated into the annual plan and allocated to specific months,

*Correspondence: Jingchun Geng (jingchungeng@163.com).

clarifying the execution month for each order. In the following year, upon entering the execution month, shippers and carriers negotiate the exact loading date. This static model of "centralized collection—unified approval" may lead to a mismatch between loading demand and transportation capacity, particularly failing to meet the delivery time requirements of the high-value-added, high-freight-rate "scattered cargo" market. It struggles to coordinate the dynamic demands of different types of clients, which is detrimental to attracting cargo flow in a competitive market.

In recent years, influenced by multiple factors such as adjustments in environmental policies, changes in the international energy trade landscape, and the transition in energy structure, the volatility in domestic coal import/export volumes and prices has significantly intensified. The drawbacks of the current static order approval model for freight transportation at CER have become increasingly apparent: 1. The extended time gap between order submission and actual transportation requires shippers to bear the market risks associated with coal price fluctuations during this period. 2. Due to competition from alternative energy sources and other factors, actual coal transportation volumes often experience unexpected significant fluctuations, leading to periodic shortages or idle capacity in short-term railway transportation resources.

Since its establishment, CER has long focused on coal transportation as its core business. In recent years, with the advancement of new line construction and the upgrading of existing routes, its transportation capacity has shown a surplus. Against this backdrop, CER is gradually transitioning from a single coal transportation model to an integrated logistics service model, aiming to enhance economic efficiency by expanding the variety of cargo types and increasing total transportation volume.

Orders for mass freight (such as ore, metal, and oil) have stable volume, large scale, and are insensitive to time. The low freight advantage of railway transportation can effectively attract such orders and has a certain adaptability to the current transportation capacity allocation system. For scattered cargo orders with high added value, high freight rate, and high profit margin, their characteristics are significantly different from those of mass freight: ① The arrival time limit is strict, and the transportation cycle usually needs to be shortened. ② The stability of the source of goods is poor, and the flow of goods is irregular. ③ The variety of goods is complex, and the traffic volume is difficult to predict accurately, which makes it difficult to reserve the transportation capacity. ④ Order submission time is flexible and time sensitive. The current static order approval mode adopted by CER makes it difficult to meet the dynamic demand of such orders due to the fixed approval cycle and rigid adjustment mechanism.

The static order approval model currently adopted by CER, characterized by fixed approval cycles and inflexible adjustment mechanisms, struggles to accommodate the dynamic demands of such orders, placing the company at a competitive disadvantage in the market.

Therefore, this study investigates a dynamic approval mechanism for freight orders on the CER. By incorporating client value coefficients and delivery-time satisfaction function, and leveraging technologies such as deep reinforcement learning (DRL) alongside real-time capacity resource data, we construct a dynamic order approval model for CER freight services. Under the premise of satisfying transportation safety constraints, this model dynamically evaluates the priority and expected cost-benefit of each order, significantly enhancing the timeliness and scientific rigor of order processing. It ensures that order decisions remain synchronized with market changes, effectively responding to dynamic transportation demands. Through quantitative analysis and optimization of multidimensional factors, the model mitigates capacity mismatches, meets client delivery requirements, and

improves corporate profitability. Ultimately, it enhances both the quality of freight services and the core competitiveness of CER in the logistics market.

1.1 Literature Review

Regarding the current state of research on freight order approval, Chen and Li point out that railway freight processes rely heavily on manual experience, suffering from "information silos" and rigid approval mechanisms, which lead to low client satisfaction due to issues such as insufficient timeliness and cumbersome procedures [1,2]. Wu and Wang argue that railways must promptly grasp market demands and leverage information technology to establish a comprehensive client relationship management system. This system would uniformly manage resources such as logistics service processes, client status, client satisfaction, and client costs, refine client classifications, and enable internal sharing to maximize efficiency in delivering satisfactory services to diverse client types [3]. Starting from the current state of railway freight transportation, Geng, based on an analysis of the railway freight market, proposes optimizing transportation organization forms by enriching product varieties, expanding product extensions, and improving transportation organization. This includes prioritizing capacity allocation, streamlining transportation processes, and simplifying procedures for approval, loading, billing, and settlement, thereby tailoring suitable transportation solutions for clients and meeting the diverse personalized needs of major clients [4]. Godwin et al. developed an analytical approximation method to estimate delivery times for freight transportation while determining the required capacity scale to meet expected service levels. The quality of this analytical approximation was validated using real numerical cases from the Indian railway system, one of the largest freight carriers globally [5,6]. Ghorpade et al., considering the possibility of actual demand deviating from the average, proposed an algorithm that adaptively adjusts planning cycles to accommodate demand fluctuations. Calculations demonstrate that this heuristic algorithm yields near-optimal solutions in terms of net profit and the number of vehicles required. When tested under dynamic demand conditions, results indicate that if the system exhibits low dynamism, the initial cyclic plan can be maintained with minor modifications. The algorithm was further tested on Indian freight rail data, showing improvements in rake utilization [7]. Ju comprehensively elaborated on and studied service quality management theory and the content of the "real cargo system" freight reform, identifying the management philosophy and methods for service quality management under this system [8]. He established an operational model for service quality management. Following this framework, starting from client requirements for railway freight service quality—including demand order acceptance, approval, etc.—the implementation process of railway freight service quality management was analyzed. Oyewole proposed a Hybrid Algorithm (HA) to solve the Fixed Charge Solid Location and Transportation problem (FCSLTP). The FCSLTP [9] considers the cost of facility location and route fixed costs during transportation planning or load consolidation. The HA integrates two heuristics into the Genetic Algorithm framework to solve the FCSLTP. Thus, the essence of railway freight order approval is a multi-constrained real-time decision-making problem (involving capacity, timeliness, and cost), necessitating the construction of a closed-loop system of "state perception–dynamic decision–feedback optimization." The online learning and policy optimization capabilities of DRL can serve as a key solution to this problem.

Breakthrough Applications of DRL Models in Complex Decision-Making. Chen et al. proposed a model-based hybrid soft actor-critic (MHSAC) algorithm that is developed based on the classic soft actor-critic (SAC) and model-based policy optimization (MBPO) framework. This algorithm can learn both continuous and discrete policies according to the current and predictive state of the patient's physiological information with high data efficiency. Results reveal that our proposed model significantly

outperforms the baseline models, achieving superior efficiency and high accuracy in the OpenAI Gym simulation environment [10]. In studies on the universality of the Soft Actor-Critic (SAC) algorithm. Hui et al. developed a revolutionary context-based deep meta-reinforcement learning (CB-DMRL) algorithm. The proposed CB-DMRL algorithm combines Bayesian optimization (BO) with deep reinforcement learning (DRL), allowing the general agent to adapt to new tasks quickly and efficiently [11]. Li et al. introduced Portfolio Zero, a novel model to address these problems. PortfolioZero utilizes three connected deep neural networks combined with a Monte Carlo Tree to discover patterns of financial assets. In the representation network, a Transformer-based model is used to embed financial price data to capture temporal dynamics and potential correlations, providing richer feature representations; the prediction network and Monte Carlo Tree Search are redesigned to handle the continuous action space [12]. Anoop et al. addressed multi-objective optimization of a freight allocation problem and presented the case of a food grain organization in India (FOI). The inventory and warehouse parameters that are relevant in the regional level allocation of food grains (using freight trains) are represented using three penalty factors, namely rake penalty factor, capacity utilization penalty factor, and weekly penalty factor [13]. In dynamic environment research, Fan et al. introduced a DRL learning-based framework, DRLAttack, capable of launching both white-box and black-box data poisoning attacks [14]. Brezulianu et al. developed an AI/ML software module, the Ferodata AI Engine, which integrates a supervised random forest model [15]. This model automatically assigns train crews to freight orders based on data, including locomotive drivers' fatigue scores, their distance to the order departure point, driver availability, and cycle history. Wang et al. proposed a three-stage reinforcement learning training framework (Offline Meta Policy Training, Online Adaptation, Fine-tuning) to avoid the risk of real systems caused by agents collecting data directly online [16].

In the research on DRL algorithms. Liu et al. developed a data-driven agent-based environment to update multi-modal transit data and stranded passenger information in real time. Two coordination strategies are introduced: (1) an independent strategy using a decentralized training and distributed execution algorithm, and (2) a collaborative strategy using a hybrid centralized training and distributed execution algorithm [17]. Jiang et al. developed a time series prediction model for big data on an improved bionic population intelligence algorithm [18]. Yu et al. introduced NSGA II for configuration optimization of a cold chain logistic network with fully shared facilities and equipment [19]. Coskun et al. studied a Deep Deterministic Policy Gradient (DDPG) algorithm, introducing a multi-objective reward function and proposing a hierarchical control strategy suitable for connected hybrid electric vehicles (HEVs) [20]. Zheng et al. proposed a novel DRL framework for urban traffic signal control in Vehicle-to-Everything (V2X) ecosystems [21]. Park et al. investigated DRL algorithms capable of policy learning in high-dimensional environments characterized by complex state-action interactions [22]. Hu and Wang introduced a new DRL method, with extensive experiments on real road networks demonstrating that it significantly outperforms four representative baseline methods [23].

In summary, from the perspective of research coverage (Table 1), systematic studies on freight order approval mechanisms, process optimization, and dynamic response to freight demand remain limited. Existing literature predominantly focuses on the execution level of transportation planning, with insufficient exploration of core issues such as decision-making logic and efficiency improvement in the order acceptance phase. Regarding order approval decision-making models, current mechanisms generally exhibit static characteristics. Most approval processes rely on predefined rules or historical data to establish fixed standards, failing to adequately incorporate variables such as real-time cargo flow fluctuations, dynamic changes in transportation capacity, and unexpected events. This results in discrepancies between approval outcomes and actual transportation scenarios, highlighting

the scarcity of dynamic research. From a technological application perspective, the research and application of DRL in the railway sector face significant limitations. Existing studies primarily concentrate on operational aspects such as autonomous train control, dynamic route planning, and marshaling yard scheduling, where the core objective is to optimize the efficiency of individual processes through intelligent algorithms. However, this technology has not yet been extended to the critical decision-making phase of freight order approval, nor has it facilitated the construction of a dynamic approval model based on real-time data. The vast data resources accumulated by railway freight transportation platforms are currently underutilized, often limited to simple statistical analysis or post-hoc reviews. They have not been structured into standardized training datasets, nor leveraged through DRL algorithms to uncover deeper insights such as demand prediction patterns or capacity matching models. Consequently, order approval decisions lack data-driven intelligent support, making it challenging to adapt to the dynamic changes in freight demand under market conditions.

Table 1: Research focus of articles on railway goods

Refs.	Study subject	Objective	Uncertainties	Path planning
[5]	Freight rail networks	Estimate order delivery times and fleet capacity via simulation	No	No
[6]	Freight rail networks	Estimate order delivery times and fleet capacity via analytic approximation	No	No
[7]	Freight railways	Alternative routing strategy using order-first-split-second heuristic	No	Yes
[9]	Location and transportation	Solve the fixed charge solid location and transportation problem	No	No
[10]	Medical ventilation	Optimize ventilator settings via hybrid soft actor-critic DRL	No	No
[11]	High-speed railway	Active pantograph control via deep meta reinforcement learning	No	No
[12]	Stock portfolio	Portfolio model based on deep reinforcement learning	No	No
[13]	Food grain supply chain	Multi-objective optimization for freight allocation	No	Yes
[14]	Data security	Data poisoning attack on collaborative filtering via DRL	No	No
[15]	Rail freight driver assignment	Optimize locomotive driver assignment using AI	No	No
[16]	Urban transit system	Online multi-modal evacuation during passenger flow outburst via multi-agent RL	Yes	Yes
[7]	Freight railways	Alternative routing strategy using order-first-split-second heuristic	No	Yes
[9]	Location and transportation	Solve fixed charge solid location and transportation problem	No	No

(Continued)

Table 1 (continued)

Refs.	Study subject	Objective	Uncertainties	Path planning
[10]	Medical ventilation	Optimize ventilator settings via hybrid soft actor-critic DRL	No	No
[17]	Time series prediction	Big data time series prediction via improved bionic population algorithm	No	No
[18]	Cold chain logistics	Configuration optimization of logistic network via NSGA II	No	Yes
[19]	Connected and automated HEVs	Energy management via multi-objective hierarchical DRL	No	No
[20]	Urban traffic control	Traffic control for the V2X ecosystem via DRL	No	No
[21]	Urban autonomous driving	Comparative study of DRL algorithms for autonomous driving	Yes	Yes
[22]	Cold chain distribution	Route optimization for multi-compartment distribution via DRL	No	Yes

Therefore, this paper proposes the construction of a dynamic freight order approval model for CER based on DRL. Grounded in the current freight transportation needs of CER, the model aims to achieve dynamic order approval decisions, enhance transportation efficiency, and improve client satisfaction. It embodies core values such as dynamic adaptability, multi-objective optimization, and systemic intelligence, holding significant theoretical and practical importance.

1.2 Contributions

1. This paper proposes a dynamic order approval model that adopts an order-activated approval process to reduce demand backlog and client waiting time. Within the context of the complete transportation process, it analyzes capacity bottlenecks involved in order decision-making, formulates demand management methods, decision processes, and evaluation criteria for decision effectiveness, thereby improving the alignment between freight demand and transportation capacity. The model refines the temporal granularity of the decision-making process, controlling the planned delivery time at the daily level.
2. The dynamic order approval model comprehensively considers order revenue, client satisfaction, and long-term revenue stability, incorporating client value coefficients and on-time delivery satisfaction into the optimization objectives. The on-time delivery satisfaction metric quantifies the sensitivity of different cargo types to delivery time deviations, integrating rigid delivery deadlines and flexible client expectations into a unified function.
3. Treating order decision-making as a multi-stage optimization problem aimed at maximizing the cumulative revenue of all decisions, this study employs reinforcement learning algorithms suitable for sequential problems. Given the complexity of network data and coupled constraints

in the decision process, deep neural networks are used for feature extraction and model simplification. Since the approval process involves multiple discrete actions and the reinforcement learning agent operates in a large action space, some discrete actions are transformed into continuous actions, and the HyAR algorithm is introduced to accelerate model convergence. Finally, real freight data from CER in March 2024 is used as experimental data to compare the performance of the dynamic order decision model against traditional static models and sliding window models. The order volume is expanded based on real data to test the model's performance under varying capacity conditions.

2 Dynamic Approval Model

2.1 Problem Description

CER handles both mass freight (coal, chemical products, ore) and scattered cargo orders. These two categories differ significantly in volume, time-sensitivity, and supply stability. An integrated evaluation must therefore incorporate delivery deadline, cargo type, and order value. By rigorously assessing each order's current and potential worth, together with the negative externalities of missing the client's expected delivery time (EDT), the operator can safeguard delivery deadlines, improve the alignment between transport capacity and order attributes, and maximize aggregate system revenue.

In addition, because it is difficult for clients to accurately predict their coal transportation demand in the next year during the annual order submission process, the reported data often fluctuates widely from the actual demand. The annual examination and approval method and the lack of a clear delivery schedule further weaken the reliability of transportation planning. In addition, because it is difficult for clients to accurately predict the coal transportation demand in the next year during the annual order submission process, the reported data may often fluctuate greatly from the actual demand. The annual examination and approval method and the lack of a clear delivery schedule further weaken the reliability of transportation planning.

2.2 Approval Model Construction

2.2.1 Model Assumptions

Assumptions 1: Shipment volume is declared in wagon units at the time of order submission.

Assumptions 2: Empty-wagon supply is sufficient: the required number and type of empty wagons are always available at the loading station.

2.2.2 Variables and Parameters Description

Related variables and parameters involved in the model (Table 2).

Table 2: Parameter and variable description

Set	
S	Station set, $\|S\|$ total number of stations
D	Date set, $D = \{1, 2, \ldots, \|D\|\}$, $\|D\|$ total number of days in the planning horizon
L	Section set, $L = \{s_1s_2, s_2s_3, \ldots\}$, a section between two adjacent stations
R	Order set, $R = \{1, 2, \ldots, \|R\|\}$, $\|R\|$ total number of orders accepted orders
C	Client set, $C = \{1, 2, \ldots, \|C\|\}$, all clients in the system

(Continued)

Table 2 (continued)

Index	
i	Station index, $S_i \in S$
j	Order index, $R_j \in R$
m	Client index, $C_m \in C$
d	Date index, $E_d \in E$
Parameter	
S_i^f	Station-S_i Capacity of daily loading
$S_{i,d}^f$	On the d day, the loading capacity occupied by the approved order at the S_i station
S_i^{f2}	Station-S_i Daily unloading capacity
S_{id}^{f2u}	On the d day, the unloading capacity occupied by the approved order at the S_i station
S_i^{f3}	Station S_i Daily combination and decomposition ability
$S_{i,d}^{fu}$	On the d day, the combination and decomposition capacity occupied by the approved orders at the S_i station
$L_{i,i}^f + 1$	Section between S_i station and S_{i+1} station, one-day capacity
$L_{i,i}^l + 1$	Section length between station S_i and station S_{i+1}
t^1	The time required for fundamental processes in freight transport, including the loading and unloading of goods
t^2	Train running time between stations
t^3	Composition and Decomposition duration (Consolidation Waiting Time)
t_j^4	Estimated transit time for order j
V^v	Train operating speed
V^f	Train full-load car count
β	Train full-load ratio
λ_1	Goods planning is earlier than EDT, time sensitivity degree.
λ_2	Goods planning is later than EDT, time sensitivity degree.
R_j^p	Order price
R_j^n	Cargo category
R_j^f	Order car count
R_j^{d1}	Client expected cargo arrival time
R_j^{d2}	Delivery time limit
R_j^{so}	Order Origin station, $R^o \in S,\ R^o \neq R^d$
R_j^{sd}	Order Destination station, $R^d \in S,\ R^d \neq R^o$
Decision variables	
x_o	Accept or reject orders, {0, 1}
x_j^d	Scheduled loading time of goods, $\{0, 1, 2, \ldots, R_j^{d1}, \ldots, R_j^{d2}\}$
x_j^{s1}	The order may be combined with other orders to form a train. $x_j^{s1} \in \{0, R_j^{so}+1, R_j^{so}+2, \ldots, R_j^{sd-1}\}$. 0 express that no combination operation will be performed.
x_j^{s2}	The train carrying the order may undergo disassembly operations. If disassembly operations are carried out, the station is at $x_j^{s2}.x_j^{s2} \in \{0, 1, 2, \ldots, R_j^{sd-1}\}$. 0 express that no decomposition operation will be performed.

2.2.3 *Dynamic Order Approval Model*

Building on the foregoing outline of the current order-approval process in CER, this study proposes a dynamic order-approval model. A new order instantiates the approval workflow upon arrival (Fig. 1).

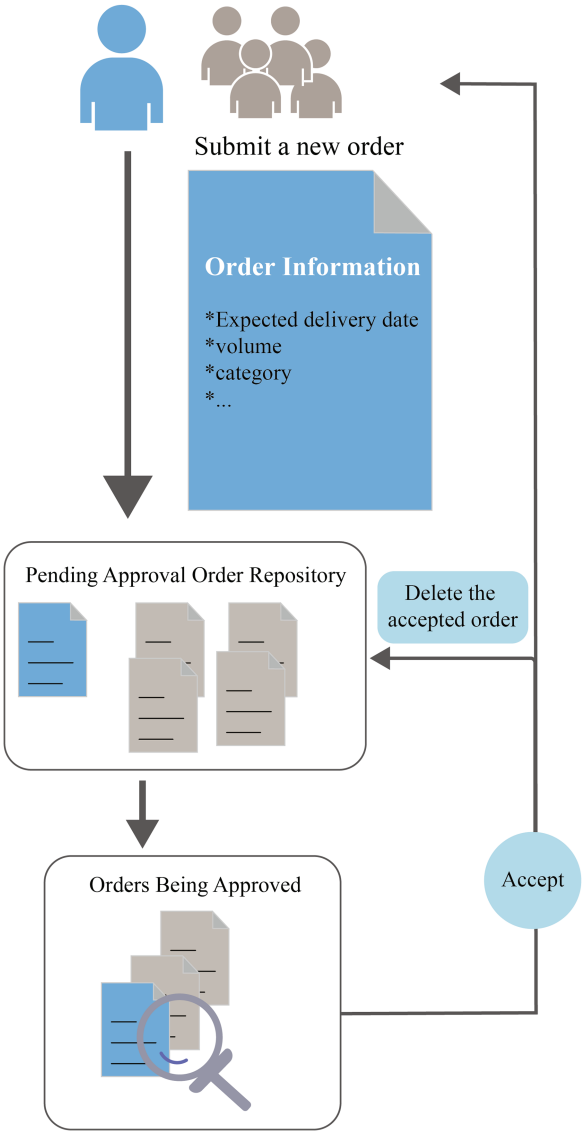

Figure 1: Schematic diagram of the dynamic order approval process

Clients can submit new orders at any time, which serve as the trigger to initiate the approval process. Each order includes the desired delivery time, delivery deadline, type of goods, and number of railcars required. New orders enter a pending approval queue, where a comprehensive decision is made on whether to accept them immediately based on factors such as order attributes, compatibility with other orders, and remaining network capacity. If an order is accepted, it is removed from the pending approval pool, and the remaining network capacity is updated accordingly.

2.2.4 *Objective Function*

Order approval constitutes the primary interface between the client and the railway enterprise; consequently, any admissible decision needs to trade off client requirements against corporate profitability. By disaggregating the delivery-time quotation and aligning it closely with the client's preferred arrival window, the proposed model enhances client satisfaction. To operationalize this objective, we formulate the first optimization objective, Z^1, centered on the time-satisfaction degree (TSD). Simultaneously, to safeguard the carrier's long-term profitability, we introduce a second object, Z^2, that explicitly incorporates client lifetime value.

$$Z_j^1 = tsd \cdot R_j^1 \tag{1}$$

$$Z_j^2 = c_j \cdot \left(R_j^p - prime\ cost\right) \tag{2}$$

Primary cost denotes the primary expense incurred by the carrier for transporting the order's freight, comprising a fixed order-specific cost (CO_1) and a variable cost (CO_2 as mileage coefficient) [24].

$$prime\ cost = CO_1 + CO_2 \times R_j^f \times \sum\nolimits_{i=R_j^{od}}^{R_j^{sd}-1} L_{i,i+1}^l \tag{3}$$

Delivery-time satisfaction function

The prevailing logistics market is buyer-dominated; hence, carriers need to accommodate shippers' requirements as fully as possible to enhance service quality. From the client's perspective, we innovatively propose a delivery-time satisfaction function. Traditional delivery-window constraints treat any arrival within the deadline as equally acceptable, implicitly assuming uniform client impact. In contrast, the satisfaction function assigns a time-sensitive utility value to every instant inside the window, conditional on the order-specific sensitivity [25]. By exploiting heterogeneous time sensitivities across commodities, the model allocates capacity along the temporal dimension: when local capacity conflicts arise, shipments with lower sensitivity are re-timed first. Because advancing a shipment is less detrimental than delaying it, the client time-satisfaction function is designed as follows, the schematic diagram of the function is shown in Fig. 2.

$$tsd \begin{cases} \dfrac{1}{\sqrt{2\pi}\lambda_2} \cdot e^{\wedge}\left(-\dfrac{\Delta d^2}{2\lambda_1^2}\right) \Delta d\ < 0 \\ \dfrac{1}{\sqrt{2\pi}\lambda_2} \cdot e^{\wedge}\left(-\dfrac{\Delta d^2}{2\lambda_2^2}\right) \Delta d \geq 0 \end{cases} \tag{4}$$

Let denote Δ_d the deviation between the scheduled delivery date and the client's EDT. The planned x_j^d loading (departure) date is treated as an auxiliary variable; together with the en-route transit time, it determines the projected unloading (delivery) time. The travel time (t^4) is simplified as the sum of a base transit time (t^1), Section running time $\left(\dfrac{\sum_{i=R_j^{so}}^{R_j^{sd}-1} L_{i,i+1}^l}{V^v}\right)$, and train dwell time ($t^2$). Clients perceive early and late deliveries asymmetrically: in general, a tardy arrival relative to the desired date exerts a markedly stronger negative effect than an early arrival. The parameter accordingly $\dfrac{\lambda_1}{\lambda_2}$ weights the relative impact of earliness vs. lateness on overall system revenue. Coupling, uncoupling, and

waiting time occur only for non-through trains; they comprise the time consumed during wagon reformation and accumulation at classification yards. It is assumed that the working time of a train combination/decomposition is fixed. Non-direct trains are combined or decomposed k times, and the duration of all combination and decomposition operations of one train is t^3. The number of cycles is determined by the diversity of loading and unloading points embedded in the train set:

$$k = \left|\left\{R_j^{so} \times x_1, R_j^{sd} \times x_2\right\}\right| \quad \text{where } R_j^{so}, R_j^{sd} \in S \tag{5}$$

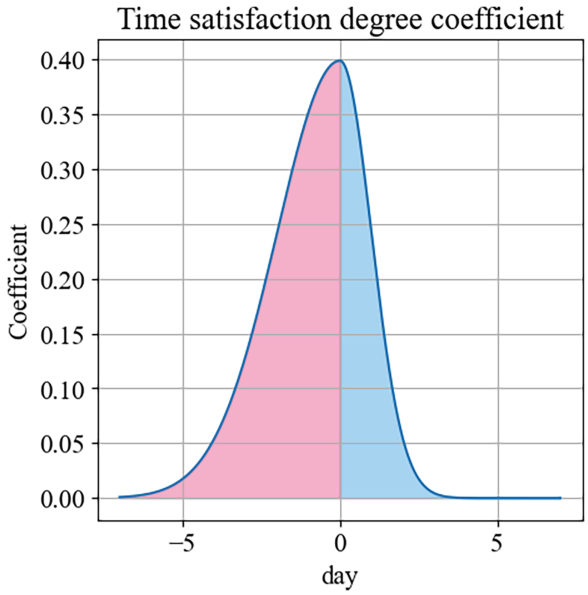

Figure 2: Delivery-time satisfaction function image. The delivery time limit is designed to be 3 days. Arriving goods six days in advance or three days late will result in almost no profit

Travel time is expressed as: $t_4 = t_1 + t_2 + k \cdot t_3$.

Client value function

Carriers ration limited capacity by prioritizing high-yield orders since an overzealous focus on customer happiness invariably drives up operating expenses. Even though the clearance decision is made instantly, its effects could spread over a considerably longer period of time [26].

A comprehensive valuation of an order must therefore subsume not only its immediate surplus (the difference between the quoted price and the attendant transport cost) but also its expected impact on future earnings. Adopting a client-centric perspective, we assess the long-term profit implications of accepting a single order by grounding the analysis in the value that the client generates for the firm. Drawing on client lifetime-value theory, we construct a composite index from five observable dimensions—annual tonnage, gross margin, cargo class, and two additional firm-specific indicators—to quantify client value (Table 3). To mitigate distortions arising from clients with sparse transaction records during the observation window, the client-value coefficient is truncated to the interval [0.01, 0.50].

One of the main functions of dynamic order approval is to serve the future market order increase. This paper chooses the TOPSIS method to calculate client value, and the calculation method is Algorithm 1 [27].

Table 3: Factors of client value consideration

Symbol	Name	Describe
c_m^1	Freight volume	The average number of vehicles per order per unit time.
c_m^2	Value of cargo	The higher the value of the consignment, the higher the additional profit.
c_m^3	Distance	Average distance of client orders.
c_m^4	Unit price	The average unit price of all orders of the client.
c_m^5	Transaction frequency	Number of transactions per unit time. The higher the transaction frequency, the stronger the client dependence.

Algorithm 1: Client value calculation process

Input: data of five indicators of all clients within the statistical time.

1. The historical data of all clients can be represented by an original data matrix $\boldsymbol{A}$. m for client number, which is used to distinguish clients.

$$\boldsymbol{A} = \left[c_m^i\right]$$

Different indicators have different meanings and units of measurement, and their orders of magnitude are different. They are transformed into dimensionless relative values through standardization, and all values are in the same order of magnitude, ranging from *0* to *1*. The normalized matrix is represented by $\boldsymbol{A}'$.

The larger the value, the better the index: $\left(c_m^2, c_m^4, c_m^5\right)$

$$c_m^{i'} = \frac{c_m^i - \min\left(c_m^i\right)}{\max\left(c_m^i\right) - \min\left(c_m^i\right)} i = 1, 2, 3, .., nj = 1, 2, 3, ..m$$

The smaller the value, the better the index: $\left(c_m^1, c_m^4\right)$.

$$c_m^{i'} = \frac{\max\left(c_m^i\right) - c_m^i}{\max\left(c_m^i\right) - \min\left(c_m^i\right)}$$

Where refers to $c_{ij}^{'}$ the i-th row and m-th column element of matrix $\boldsymbol{A}'$, and represents the quantized value of the i-th index of client m.

3. Entropy value of indicator i:

$$entropy_i = -\frac{1}{\ln\left(\max\left(i\right)\right)} \sum_i \left(\frac{c_m^{'i}}{\sum_i c_m^{'i}} \times \ln\left(\frac{c_m^{'i}}{\sum_i c_m^{'i}}\right)\right)$$

4. Entropy weight of index i:

$$w_i = \frac{1 - entropy_i}{\sum_{i=1}\left(1 - entropy_i\right)}$$

5. Weighted standardized matrix of evaluation index

The weight of w^i each evaluation index is calculated by the entropy method. Weighting the index value to $c_j^{i'}$ obtain a matrix:

$$\overline{\mathrm{A}} = \begin{bmatrix} w^1 c_1^{1'} & w^2 c_1^{2'} & \ldots & w^5 c_1^{5'} \\ w^1 c_{21'} & w_2 c_2^{2'} & \ldots & w^5 c_{25'} \\ \ldots & \ldots & \ldots & \ldots \\ w^1 c_m^{1'} & w_2 c_m^{2'} & \ldots & w^5 c_m^{5'} \end{bmatrix}$$

6. Determine the positive and negative ideal solutions

Positive Ideal Solution:

(Continued)

Algorithm 1 (continued)

$\overline{A}^{+} = \max\left\{\left(w^1 c_m^{1'}\right)^{+}, \left(w^2 c_m^{2'}\right)^{+}, \ldots, \left(w^5 c_m^{5'}\right)^{+}\right\}$

Positive Ideal Solution: $\overline{A}^{-} = \min\left\{\left(w^1 c_m^{1'}\right)^{-}, \left(w^2 c_m^{2'}\right)^{-}, \ldots, \left(w^5 c_m^{5'}\right)^{-}\right\}$

The positive ideal solution is the set of optimal values for each criterion, while the negative ideal solution is the set of worst values.

7. Calculate the distance of client i from the ideal solution

Let the Euclidean distances from the sample to the positive and negative ideal solutions be denoted as B^{+} and B^{-}, respectively; then:

$$\begin{cases} B_i^{+} = \sqrt{\sum_{j=1}^{n}\left(w_i c_m^i - \bar{A}^{+}\right)^2} \\ B_i^{-} = \sqrt{\sum_{j=1}^{n}\left(w^i c_m^i - \bar{A}^{-}\right)^2} \end{cases}$$

8. Client value calculation, used to indicate the importance of a client.

$$c_i^p = \frac{B_i^{-}}{B_i^{+} - B_i^{-}}$$

2.2.5 Network Capacity Constraints

A graph made up of nodes and arcs can be used to represent the railway transportation network, where all components work together to complete transport operations. The amount of transport work that can be finished in a single day is limited because of capacity constraints placed on this network by infrastructure, machinery, and human resource management [28]. Cargoes are unlikely to arrive within the specified delivery window if the accepted order volume over this limit. A very detailed estimate of network capacity is not required at the order-approval stage since many variables, including the actual train-deployment plan and the following train graph development, are still unknown. To lessen the computational load, capacity limits are thus treated in a purposefully ambiguous manner.

Constraints on station loading capacity:

$$S_i^{f1} \geq S_{i,d}^{f1u} + \sum\nolimits_j x_j \times R_j^f \quad d = x_j^d \tag{6}$$

Constraints on the unloading capacity of stations:

Upon arrival at the destination station, trains are unloaded on the same day. Under the stated assumption, the unloading time is equal to the loading date of the freight plus the transit time route.

Constraints on decomposition and composition capacity of stations:

Except for the through train, the other trains need to be combined and decomposed at the station.

$$S_i^{f3} = S_{i,d}^{bu} + \sum\nolimits_j R_j^f$$

$$R_j^f \in \{\textit{Vehicles (belongingorders) that are planned to be disassembled at Station } i.\} \tag{7}$$

Section passing capacity constraint:

$$b_{ia} > L_{iat} > z, x \in R_f, \quad L \in \left\{R_j^{os}, R_j^{os+1}, R_j^{os+2}, R_j^{od}\right\} \tag{8}$$

Full axis rate constraint:

In this paper, the "full-axle ratio" is calculated on the basis of the number of wagons, defined as the ratio of the actual wagons assembled in a train to the maximum permissible train length expressed in wagons. When capacity is scarce, trains are customarily operated at full axle to maximize total throughput. Conversely, when capacity is abundant, the full-axle ratio is deliberately reduced to enhance operational flexibility and to accommodate dynamic fluctuations in market demand.

$$\beta \leq \frac{\sum_j x_j R_j^f}{\nabla f} \leq 1 \tag{9}$$

Delivery deadline constraint:

The latest delivery time of the cargo ordered.

$$R_j^{d2} > x_j^d + \left\lceil t_j^4 \right\rceil \tag{10}$$

2.3 Mathematical Expression of the Dynamic Approval Model

The description of DOA mentioned above can be transformed into the description in Table 4.

Table 4: Mathematical expression of the DOA

Core decision variables:
$x_0, x_0 \in \{0, 1\}$
Auxiliary decision variable
$x_j^d, x_j^d \in \left\{0, 1, 2, \ldots, R_j^{d1}, \ldots, R_j^{d2}\right\}$
$x_j^{s1}, x_j^{s1} \in \left\{0, R_j^{so}+1, R_j^{so}+2, \ldots, R_j^{sd-1}\right\}$
$x_j^{s2}, x_j^{s2} \in \left\{0, x_j^{s1}, x_j^{s1+1}, \ldots, R_j^{sd-1}\right\}$
Objective functions:
$\max Z_j^1 = tsd \cdot R_j^p$
where *tsd* is time-satisfaction degree, the calculation follows formula (2).
$\max Z_j^2 = c_j \cdot \left(R_j^p - \text{prime cost}\right)$
where "*prime cost*" refers to the main cost of transportation, calculated according to formula (3).
Constraints

3 Deep Reinforcement Learning Algorithm

3.1 Agent and Environment Construction

Reinforcement learning can be viewed as the process in which an agent learns the optimal policy function through interaction with its environment. In accordance with the decision variables, constraints, and data characteristics arising during the order-approval process, we construct both the agent and the virtual environment.

3.1.1 Deep-Neural-Network-Based Agent

Although the only decision variable in DOA is binary (accept/reject), auxiliary variables—loading time and decomposition yard—are required to evaluate the decision's effectiveness.

In the preceding mathematical model, all three decision variables (acceptance, loading date, and decomposition yard) were treated as discrete. For a neural network, discrete variables impede stable gradient computation. Interpreting the loading date as a continuous temporal variable allows the model to perform domain search more naturally and enables the network to discover temporal patterns more easily.

Geographical position, transport capacity, and connectivity differ markedly among stations. For example, in a complex network, two yards may be geographically close yet lack a direct link. Consequently, stations are represented by one-hot encoding.

The agent's observation consists of two parts: network information (station and segment attributes) and order information. One-day single station information and single order information are rearranged into a one-dimensional format, which is convenient for deep network reading (Fig. 3). Network information comprises adjacent stations, remaining loading/unloading capacity, decomposition capacity, residual segment capacity, and segment length. Order information comprises origin station, destination station, number of wagons, client value, order-submission time, desired delivery time, and delivery deadline.

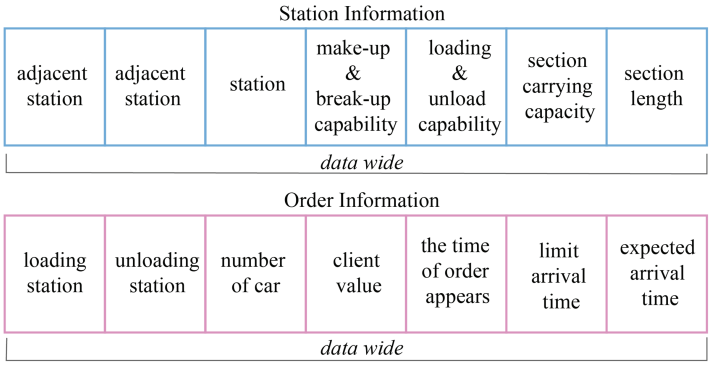

Figure 3: Data storage format of station information and order information

Due to the large amount of input information, a Convolutional Neural Network (CNN) is chosen for feature extraction. The common convolution kernel size of CNN's convolutional network is a square of 3 or 5. This results in mixing between data from different rows, making the language information unclear. Combining the semantics and format of the data, the convolution kernel of the first layer is designed to be 1 $\times$ data wide (data wide = the data length of a single station information or the data length of a single order information). The convolution kernel only moves along the station dimension, maintaining the original feature extraction to understand the semantic information of the data.

The design schematic of the model is as Fig. 4, with the main body consisting of a CNN. Road network information and order information are extracted through two independent one-dimensional layered convolutions, concatenated by channel, and then enter the Resnet module for information fusion. Finally, the output is generated through FCN.

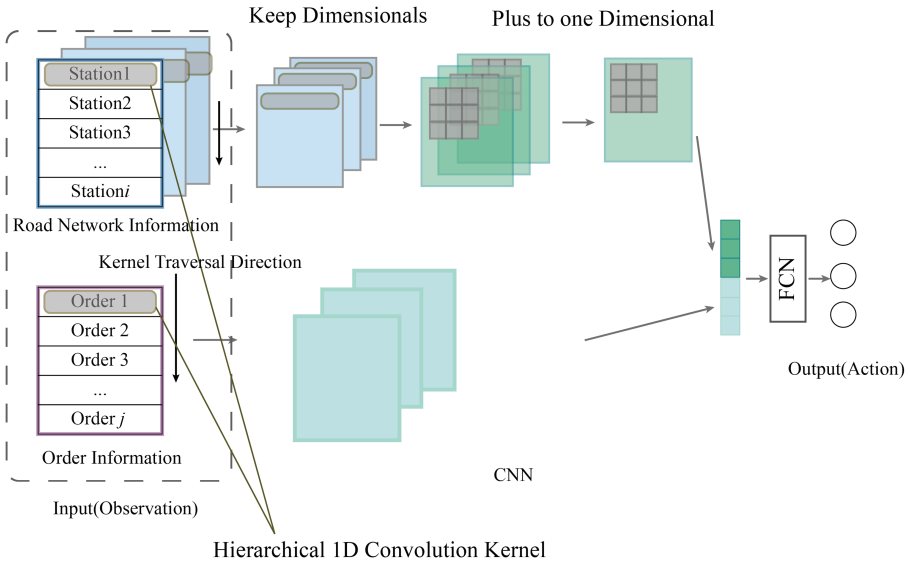

Figure 4: Deep neural network structure of the decision agent net

3.1.2 Construction of the Virtual Environment

The agent interacts with the virtual environment, which simulates the real world. After absorbing the agent's action and carrying out internal calculations, it sends back a reward signal. The environment in this study keeps two persistent repositories: a network-state database and a pending-order pool. The entire historical record of every order is kept in an auxiliary order archive. Real or synthetic order instances can be found in the order data set, which is a list. In actuality, if no approving response is given within a fair amount of time, an order may be withdrawn. As a result, the agent's action serves as the catalyst for the state transition. The environment first updates the pending-order pool after accepting an action by executing the approved/rejected decision in accordance with the mathematical model outlined in Section 2 and deleting expired orders based on the amount of time that has passed. The pending-order pool is then filled with a single new order that has been sampled from the order data set. Lastly, the environment releases a new observation, which includes the updated pending-order pool and the current network status, together with the instant reward. Fig. 5 illustrates the agent-environment interaction process.

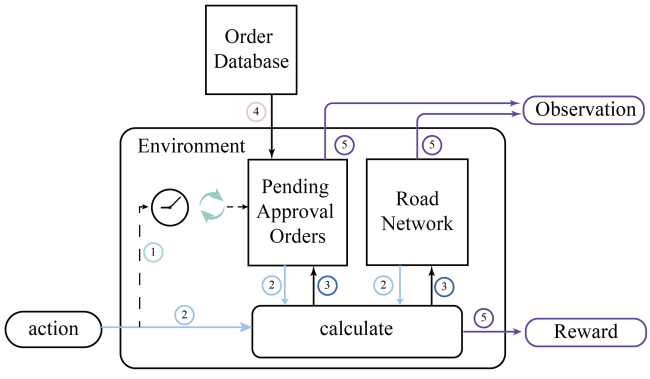

Figure 5: The process of an agent interacting with a virtual environment online and collecting data

Depending on the agent's action and the current resource state, action–state pairs fall into three cases: ① The order is accepted, and sufficient capacity is available; remaining capacity decreases. ② The order is accepted, but the capacity is insufficient; the remaining capacity remains unchanged. ③ The order is rejected; remaining capacity remains unchanged. Cases ① and ③ are valid, whereas case ② constitutes an erroneous action. In the ideal policy, the agent maximizes the objective function while avoiding erroneous decisions. Consequently, the reward function is defined as follows:

$$reward = \begin{cases} \sum_{j=0}^{C} \mu_1 \cdot Z_j^1 + \mu_2 \cdot Z_j^2 & ① \\ \sum_{j=0} -\mu_3 \times R_j^p & ② \\ 0 & ③ \end{cases} \tag{11}$$

The order price values exhibit significant fluctuations. Using the raw values to compute the reward can lead to large gradient oscillations and slow convergence. Numerical stability is achieved through hyperparameters μ_1, μ_2 and μ_3, where the ratio of μ_1 to μ_2 represents the weighting between the two objective functions.

3.2 The SAC+HyAR Algorithm

Current mainstream reinforcement learning algorithms are primarily applied to either discrete or continuous action spaces [28–30]. The dynamic order-approval task, however, entails both discrete decisions (accept/reject) and continuous variables (loading time, decomposition yard), so these standard algorithms cannot be applied directly.

The HyAR framework regards discrete and continuous actions as heterogeneous modalities that nevertheless jointly drive state transitions [31]. Its key idea is to map both types of actions into a shared continuous latent space, after which the problem can be treated as a standard continuous-control task. By representing mixed actions in a unified latent space, no additional bridging mechanism is required. Because the latent actions must ultimately be mapped back to the original hybrid space for environment interaction, the overall architecture is invertible. Discrete variables are projected using an embedding table, while continuous variables are projected using a conditional VAE. In the Hyar framework, the process of interaction between the agent and the environment is shown in Fig. 6. This process can be summarized as follows:

$$latent\ space\ action = \pi\ (observation) \tag{12}$$

$$e_{\text{discrete}} = arg\ min_k |a_{ltn} - E_k| \tag{13}$$

$$a_{original} = e_{cont} = E^{r2}\ (e_{disc}, a_{ltn}) \tag{14}$$

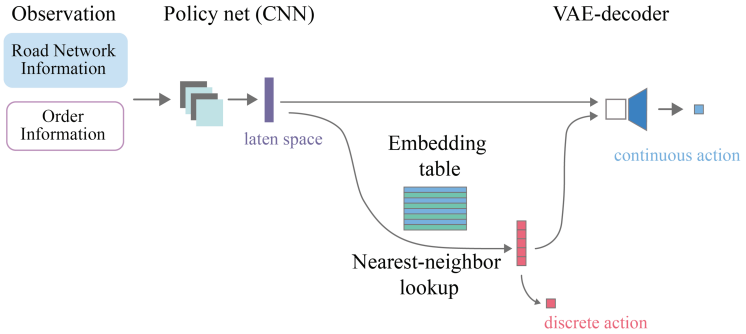

Figure 6: The data flow of the Hyar framework during the inference process

Query the embedding table to obtain discrete original actions.

Where $\pi(\cdot)$ is policy net, E^1 is embedding table, E^{r2} is VAE decoder part.

Li et al. conducted performance tests on Hyar in several scenes of Gym, and the results show that Hyar can effectively accelerate the model [32]. Hao et al. used the optimized Hyar algorithm to select tasks at the UAV task point, and the experiment shows that the method can accelerate the model convergence [33].

We build upon the Soft Actor-Critic (SAC) algorithm by replacing the actor-training module with HyAR [33]. The action-generation pipeline and training procedure are illustrated in Fig. 7. In the resulting SAC+HyAR algorithm, the process that converts discrete actions into latent vectors together with the CVAE encoder is termed the HyAR encoder conversely, the mapping from latent vectors back to discrete actions and the CVAE decoder are collectively called the HyAR decoder. Notation and variable names are re-defined accordingly (Table 5).

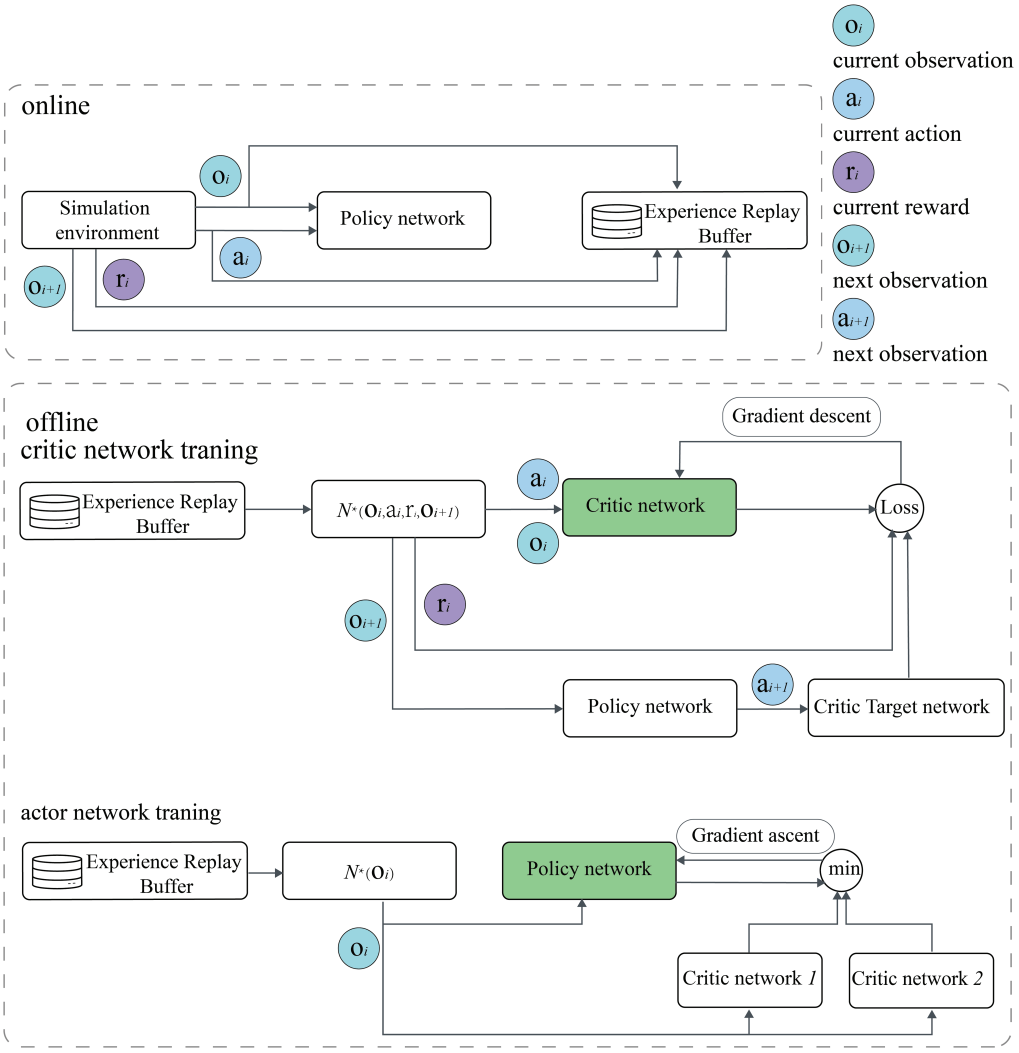

Figure 7: The flow chart of online data collection and offline training of agents

Table 5: SAC&HyAR algorithm parameters

Initialize		
Hyperparameter initialize		
discount rate	γ	0.95
Soft update coefficient of network	τ	0.0001
Initial temperature parameter	α_{init}	1
Target entropy	H	
Experience playback buffer capacity	B	5000
Training batch size	D	500
Policy network learning rate	η_1	0.002
Critic network learning rate	η_2	0.0025
VAE net learning rate	η_3	0.002
Temperature update rate	η_α	
Network parameter initialization		
Policy network	$\pi_{\varphi 1}$	
Critic network	$Q^1{}_{\varphi 2,}\ Q^2{}_{\varphi 3}$	
Target critic network	$Q^3{}_{\varphi 4}, Q^4{}_{\varphi 5}$	
HyAR encoder	$E^p = E^1_{\varphi_6} + E^{p2}_{\varphi_7}$	
(VAE encoder network + Embedding network)		
HyAR decoder	$E^r = E^1_{\varphi_6} + E^{r2}_{\varphi_8}$	
(VAE decoder network, Embedding network (reversal project))		
HyAR network traning		
Initialize embedding network parameters		
Traning VAE network parameters		
Freeze embedding network parameters and VAE network parameters		
Training cycle		
Latent act:		
If random sampling:		
The laten act of randomly sampling from latent space.		
Else:		
Policy network generation letan space act. *laten sapce act* $= \pi\ (o_i)$		
Latent space generation origin action:		
$originalact = E^r$ (latent space act)		
Input the action into the environment		
return to the next observation(o_i+1) and reward(r_i)		
Store interactive data into $\{O_i, \text{act}_i, latenact_i, r_i, O_{i+1}\}$ buffer		
Network traning:		
Random sampling from experience playback buffer		
$\{O_i, \text{act}_i, latenact_i, r_i, O_{i+1}\}^D$		
For each $i = 1$ to D:		
Calculate Q_value:		
$Q_{\text{target}} = r_i + \gamma \left[\min\left(Q^1\left(O_i, \text{latent act}_i\right), Q^2\left(O_i, \text{latent act}_i\right)\right) - \exp\left(\log\alpha\right)\log\pi\left(a_i^{'}|O_i\right)\right.$		
$\mathcal{L}_{Q^1} = \frac{1}{D}\sum_{i=1}^{D}\left(Q^1\left(O_i, a_i^{\text{latent}}\right) - Q_{\text{target},i}\right)^2$		

(Continued)

Table 5 (continued)

$$\mathcal{L}_{Q^2}(\varphi_3) = \frac{1}{D}\sum_{i=1}^{D}\left[Q^2(O_i, E^r(o_i)) - Q_{\text{target},i}\right]^2$$

Update critic network parameters:

$$\varphi_2 \leftarrow \varphi_2 - \eta_2 \nabla_{\theta_1} \mathcal{L}_{Q^1}(\varphi_2)$$

$\varphi_3 \leftarrow \varphi_3 - \eta_2 \nabla_{\theta_1} \mathcal{L}_{Q^2}(\varphi_3)$

Update policy network parameters:

$$\overline{a_i} \sim \pi_\phi(O_i)$$

$$\mathcal{L}_\pi(\phi) = \frac{1}{D}\sum_{i=1}^{D}\left[\alpha \log\pi\left(\tilde{a}_i|O_i\right) - min\left(Q_{\theta_1}\left(s_i, E^p\left(\tilde{a}_i\right)\right), Q_{\theta_2}\left(s_i, E^p\left(\tilde{a}_i\right)\right)\right)\right]$$

$$\phi \leftarrow \phi - \eta_1 \nabla_{\phi \mathcal{L}_\pi}(\phi)$$

Update temperature parameters:

$$\mathcal{L}_\alpha = \frac{1}{D}\sum_{i=1}^{D} -\log\alpha \cdot (\log\pi(\widetilde{a}_i|O_i) + \mathcal{H})$$

$$\log\alpha \leftarrow \log\alpha - \eta_4 \nabla_{\log\alpha} \mathcal{L}_\alpha$$

$\alpha \leftarrow \exp(\log\alpha)$

Update target parameters:

$$\varphi_3^{'} \leftarrow \tau\theta_1 + (1-\tau)\varphi_3^{'}$$
$$\varphi_4^{'} \leftarrow \tau\theta_1 + (1-\tau)\varphi_4^{'}$$

4 Simulation Experiments and Analysis

The simulation network consists of nine stations (Hanjiacun, Aobaogou, Daliuta South, Shenmu North, Shenchi South, Dingzhou West, Litianmu, and Huanghua Port) along the core operating corridors of CER—Baoshen, Shenshuo, and Shuohuang lines—forming a 986.5 km route, as shown in Fig. 8. Multiple power plants are clustered around S6 and S7, while S9 is a seaport; the dominant freight flow runs from S1 to S9.

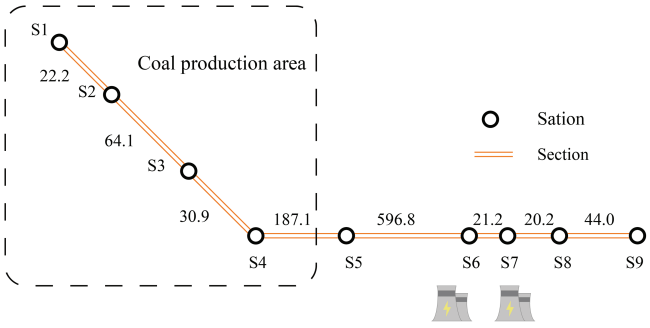

Figure 8: Schematic diagram of simulated road network

We use March 2024 waybill data (Data Set I) as the baseline dataset. Records from 01–24 March serve as the training set, and those from 24–31 March (a complete calendar week) constitute the test set. Against the backdrop of the "Comprehensive Logistics" (The development direction of CER in order to increase revenue) initiative, market-generated orders are expected to rise steadily. Therefore, we augment the March data with additional market orders to create Data Set II. Because coal-order

volumes are also highly volatile, we further compile Data Set III to evaluate model performance under a substantial surge in demand.

4.1 Data Pre-Processing

Additional orders in Data Set II are generated by perturbing two key attributes of the original records: EDT and shipment volume. The calculation method is Algorithm 2. The statistical information of data set I, data set II and data set III are shown in Tables 6 and 7. Table 6 is the training data information, and Table 7 is the test data information.

Algorithm 2: A virtual new order information making process

1. Count the departure time set {*date*} of the original order and the traffic volume set {*number of compartment*}^{n} by category.
2. Randomly draw an original order.
3. Randomly select a date (d) from {*date*}. The category {*number of compartment*} to which the new order belongs randomly selects a number as the traffic volume.
4. EDT = d + 1.5 (section distance and/or train speed). The coefficient of 1.5 is the statistics of historical orders, the average time consumption and (distance/ideal train speed).
5. Replace the EDT and order volume to generate a new order.

Table 6: Model training data information

Number of data set	Total orders	Number of compartments on all orders	Coal	Other mass freight	Scattered cargo
I	1025	96,579 (100%)	1007	15	3
II	1187	117,557 (121%)	1007	150	30
III	1519	137,713 (142%)	1309	150	60

Table 7: Model test data information

Number of data set	Total orders	Number of compartments on all orders	Coal	Other mass freight	Scattered cargo
I	279	24,341	275	4	0
II	326	27,143	275	40	11
III	419	34,419	357	40	22

4.2 Hyper-Parameter Settings

Some parameters mentioned above were not assigned fixed values; their descriptions are provided in Table 8. Because the reward function in reinforcement learning directly reflects real-world requirements, we adopt the following design principle: the weight placed on long-term revenue equals that on immediate revenue, and the weight assigned to client time-sensitivity equals that assigned to overall revenue.

Consequently, the mean of positive rewards is scaled to 3, whereas negative rewards are clamped at –50.

Table 8: Values of superparameters and partial variables

Parameter name	Symbol	Value
Number of vehicles with full axle		55
Train speed	V^v	80 km/h
Minimum full-axis coefficient	β	0.7
Minimum time consumption of single combination decomposition		1 h
Fixed cost per train		23000 yuan
Fixed cost per order		10000 yuan
Cost per vehicle per kilometer		17 yuan per vehicle per km
Unit price of coal transportation		20 yuan per vehicle per km
Unit price of other mass freight transportation		35 yuan per vehicle per km
Unit price of Scattered cargo transportation		40 yuan per vehicle per km
Reward adjustment parameter	μ_1, μ_2, μ_3	$1.37e^{-5}$, $2.21e^{-5}$, $7.3e^{-5}$
Order time sensitivity	λ_1, λ_2	0.6, 0.3 (Scattered cargo) 2, 1 (mass freight)

4.3 Experimental Results and Analysis

4.3.1 Algorithmic Convergence Speed

To accelerate model convergence, the auxiliary variable "loading time" was converted from a discrete to a continuous representation when designing the agent's action space. Consequently, the agent's actions contain both continuous and discrete variables, motivating the introduction of the HyAR algorithm to project both types into a shared latent space for optimization. To demonstrate the acceleration effect of the SAC + Hyar hybrid algorithm, we include SAC and Double DQN as baselines [34]. Since SAC is inherently designed for continuous control, the original continuous dimensions were retained via kernel-density estimation. Detailed Actor-network hyper-parameters are listed in Table 9; the pending-order buffer size was fixed at 20. The architecture comprises two independent hierarchical convolutional branches and a ResNet module built on ResNet-18, a total of 11 328 741 parameters for SAC + Hyar.

Table 9: Actor network

Layer	Layer number	Parameter	
CNN		Kernel size(c × h × w)	
		Railway network information branch (depthwise convolution)	Order information branch
	1	7 × 1 × 9	16 × 1 × 7
	2	7 × 1 × 9	32 × 1 × 7

(Continued)

Table 9 (continued)

Layer	Layer number	Parameter	
	3	$7 \times 1 \times 9$	$16 \times 1 \times 7$
	4	$16 \times 1 \times 1$	
AdaptiveAvgPool2d	5	7×7	7×7
feature fusion		Concatenation with channel	
Resnet	6–24	Resnet18	
FCN	25	Input: 512 output: 256	
	26	Input: 256 output: 16	
	27	Input: 16 output: 3	

The three methods employ similar network architectures (differing only in the final output layer). Suitable parameters were selected through grid search. During training on Dataset I, the convergence behavior of the models is shown in the Fig. 9. The SAC + Hyar algorithm converged around episode 2500, while the standalone SAC algorithm converged around episode 2900. After convergence, the SAC + Hyar algorithm exhibited smaller fluctuations in reward. After the SAC + Hyar method converges, the reward value fluctuates slightly.

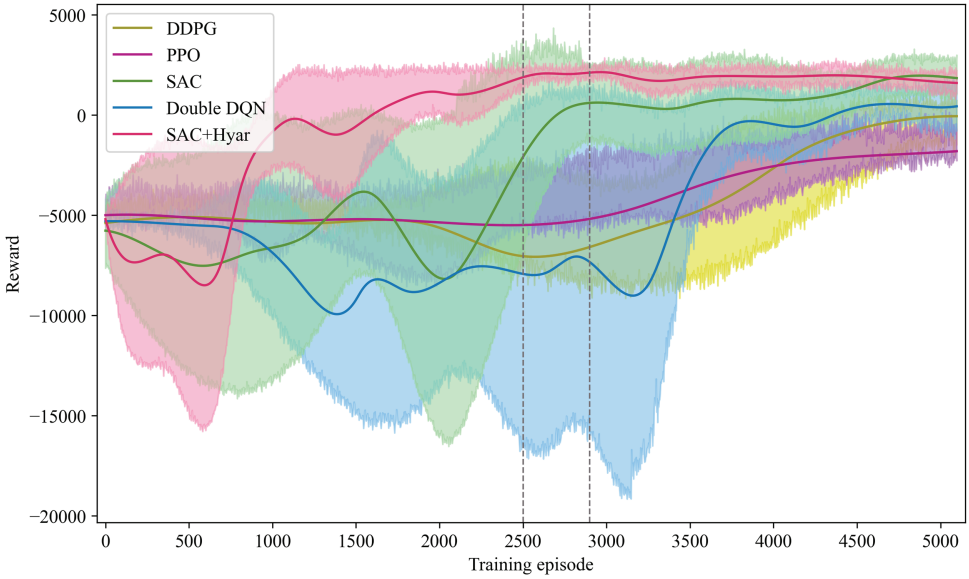

Figure 9: HyAR training convergence curve

4.3.2 Comprehensive Revenue Comparison

We benchmark the proposed framework against three widely-adopted approval strategies. Program A First-Come-First-Served (FCFS): orders are examined in their arrival sequence and accepted immediately if residual transport capacity is sufficient.

Program B: Rolling-Time-Window + Ant Colony Optimization (ACO), the system accumulates orders for a fixed time interval and then solves the resulting batch with an ACO meta-heuristic [35]. Program C: Static Approval (currently deployed by CER): the algorithm waits until the order-collection window closes, after which all pending requests are optimized in a single ACO run. The key parameter settings of the ACO algorithm are summarized in Table 10.

Table 10: Parameter setting of ant colony algorithm

Parameter name	Value
Number of ants	50
Maximum iterations	100
Pheromone weight	1
Heuristic weight	2
Pheromone evaporation rate	0.1
Pheromone deposition coefficient	100
Initial pheromone level	0.1

The single-solution time consumption of the SAC+HyAR method, Solution B, and Solution C is shown in the Table 11. The deep learning model can reduce the time required for a single solution and improve response speed. The computation time of the SAC+HyAR method is related to the capacity of the pending approval pool. In this experiment, the capacity was consistently fixed at 20, resulting in a constant solution time. (The capacity of the pending order pool was determined through multiple experiments, where it was observed that the number of orders stored in the pool never exceeded 20). In contrast, Solutions B and C employ a centralized approval approach, so their solution time gradually increases as the number of orders grows.

Table 11: The per-solve time consumption of the three optimization algorithms(s)

	SAC+HyAR	B	C
I	0.027	0.159	1762.873
II	0.027	0.207	2607.484
III	0.027	0.269	4541.297

The four programs were solved on the three datasets with the objective of maximizing the total integrated profit; the resulting profits are reported in Table 12. Because the solution process contains stochastic elements, each reported value is the average of five independent runs.

Table 12: Integrated profit of four programs

	SAC+HyAR	A	B	C
I	46,817,810	46,817,810	46,817,810	44,523,737
II	53,138,214	47,754,166	51,031,412	51,967,769
III	58,850,244	48,868,056	52,290,583	57,075,905

In Dataset I, the number of orders is small, and the available capacity is sufficient to cover all requested transport; consequently, Programs A, B, and C all accept every order. Program C, however, postpones every decision to the last day, so a few orders time-out and are lost, which makes its final profit slightly lower than that of the other program.

In Datasets II and III total capacity is scarce, so only a subset of orders can be accepted. The Fig. 10 illustrates the acceptance process for Dataset III, each circle represents one order, and its colours indicate how long the order has already waited. In program B, the order approval time interval is fixed, and no orders are lost due to timing issues. However, before transportation capacity becomes tight, the model behaves almost like a greedy algorithm, accepting all orders that can be accommodated. The reinforcement-learning agent, trained on historical data, learns to estimate long-term profit. It therefore selects orders more judiciously, avoiding the early acceptance of low-value requests that would leave no capacity for high-value requests that arrive later. As observed in Fig. 10, both Solution A and Solution B exhibit a significant decline in order acceptance after the seventh day, while the daily volume of new orders remains similar. SAC+HyAR, by contrast, accepts very few low-value orders during the first three days; those low-value orders are visibly deferred while high-value orders are given priority. Using the profitability of Program C as the baseline, the relative profits of the other programs are compared in the Table 13. When capacity is abundant, the approval strategy has little influence on profit; the remaining differences are mainly caused by order loss. As the order volume grows and capacity becomes tight, programs with a global optimisation capability—SAC+HyAR and the static centralised program—achieve significantly higher profits.

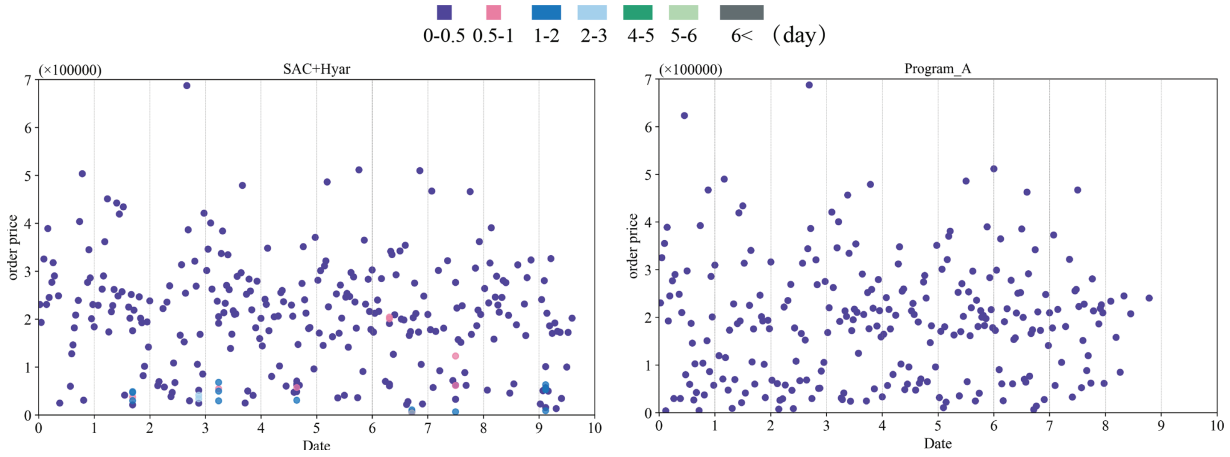

Figure 10: (Continued)

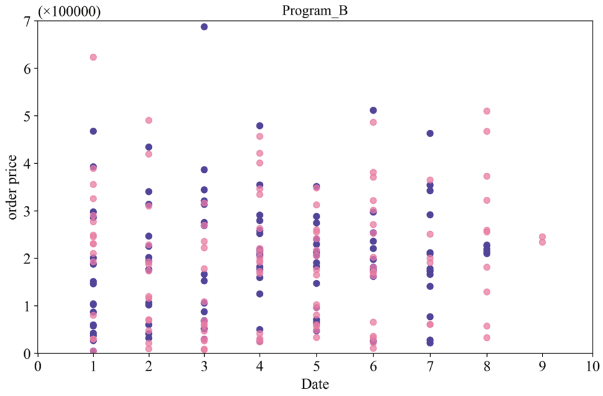

Figure 10: The time distribution of order acceptance in different programs

Table 13: Profit difference across programs

	SAC+HyAR		A		B	
	Different value	**Ratio**	**Different value**	**Ratio**	**Different value**	**Ratio**
I	2,294,073	+5.2%	2,294,073	+5.2%	2,294,073	+5.2%
II	1,170,445	+2.2%	−4,213,603	−8.1%	−936,357	−1.8%
III	1,774,339	+3.1%	−8,207,849	−14.3%	−4,785,322	−8.3%

Both Program B (which involves fixed-time approval) and DOA can effectively reduce the time customers wait for approval. Table 14 shows the waiting time for approval for orders with different price levels (from low to high: the top 20%, 20%–50%, and after 50%). In the DOA, the higher the value of the order, the faster the approval speed, reducing the risk of loss. Plan B adopts a daily approval schedule, with an average waiting time of 0.5 days. The fixed time interval approach prevents individual orders from experiencing excessively long waiting times. However, regardless of the value of the order, the probability of loss remains the same.

Table 14: Waiting time for orders with different values

		Order Price		
		<20%	20%–50%	<50%
Waiting time	SAC + Hyar	0.78	0.21	0.04
	Program B	0.50	0.50	0.50

4.3.3 Performance Disparities under Single-Objective Conditions

The composite benefit re-evaluates order value by incorporating delivery-time satisfaction and client value as weighting coefficients, thereby embedding long-term returns into immediate decisions. This section quantifies how the two constituent objectives of composite benefit influence short-term revenue. In small-scale instances, sufficient capacity allows every order to be accepted, so objective

choice does not alter the final decision; consequently, the experiments employ the SAC+HyAR algorithm together with Program C on medium- and large-scale instances. Three separate objectives are adopted: (i) aggregate order value, (ii) order value weighted by the delivery-time satisfaction coefficient, and (iii) order value weighted by the client coefficient. Table 15 shows the order receipt and total order value under different objectives.

Table 15: Profit difference of different programs

	Program	Coal	Other mass freight	Scattered cargo	Target function value	Aggregate order value
Integrated profit	SAC + HyAR	247	34	17	58,850,244	37,613,597
	Program C	250	33	16	56,952,593	36,386,496
Std. value	SAC + HyAR	251	34	10	31,318,507	36,979,149
	Program C	253	31	9	30,503,515	35,716,569
Client value	SAC + HyAR	245	33	20	28,648,778	38,595,065
	Program C	246	32	20	27,666,785	37,797,156

Focusing exclusively on client value—and thereby disregarding delivery-time satisfaction—renders shipment duration neutral to revenue as long as the contractual transit-time limit is respected. Consequently, the carrier can arbitrarily reschedule deliveries within this window and accommodate additional high-value spot orders, raising aggregate revenue. However, the mean deviation of actual delivery from the client's most preferred time widens from 0.12 to 0.71 days. The present study confines the planning horizon to seven days and restricts mass freight to a five-day transit deadline; in real-world approval cycles, longer planning horizons and more lenient deadlines further magnify delivery-time uncertainty, inflating client' inventory-holding costs and eroding satisfaction. Moreover, an objective that prioritises client value skews order acceptance toward historically high-volume, high-margin clients, impeding the acquisition of new clients. When the objective is switched to delivery-time satisfaction, Scattered cargo orders—being time-sensitive—lose their revenue advantage once the time-penalty coefficient is applied. Hence, under a time-satisfaction regime and in the presence of a composite-revenue benchmark, the average direct revenue declines by approximately 1.6% as the number of accepted Scattered cargo order shipments falls.

Under circumstances of constrained transport capacity, different optimization objectives primarily influence the decision-making process when selecting between two or more conflicting orders. From an overall revenue perspective, the values under different objectives show a relatively small difference (approximately 3.2%). However, significant disparities emerge when examined from the perspective of order types. For instance, when considering only delivery time satisfaction vs. only client value, the average number of mass freight orders differs by 52.5%. Therefore, in practice, CER can adjust the proportion of specific order types by controlling hyperparameters based on their preference for different categories of orders.

4.3.4 Expand the Transportation Capacity of the Road Network

In the case of dataset II, the transportation capacity has already reached a bottleneck. After adding orders in dataset III, due to the constraints of transportation capacity, the increase in revenue is not significant. Therefore, this section observes the decision-making effects of different schemes by increasing the transportation capacity of the road network. The Table 16 shows the order receiving situation after increasing the road network transportation capacity by 10% and 20% for program A, C, and DOA.

Table 16: Order accepting status under different transportation capacities

		Total number of orders	Number of coal order	Number of other mass freight	Number of scattered cargo	Aggregate order amount
Original	SAC + Hyar	313	263	31	19	58,850,244
	Program_A	317	273	29	15	48,868,056
	Program_C	297	255	35	7	57,075,905
Increase by 10%	SAC + Hyar	337	286	32	19	62,557,809
	Program_A	350	301	32	17	53,526,513
	Program_C	327	277	35	15	61,208,201
Increase by 20%	SAC + Hyar	366	312	35	19	65,906,529
	Program_A	386	332	35	19	59,217,725
	Program_C	361	309	35	17	64,609,924

With sufficient orders, transportation capacity can be increased, leading to an increase in both the number of received orders and overall revenue. However, the proportion of revenue increase is slightly lower than the growth in transportation capacity. This is because, whether it is static optimization or dynamic optimization, high-value orders are prioritized when transportation capacity is limited. Therefore, the new orders added due to increased transportation capacity have lower prices compared to previous orders. Consequently, the revenue growth is less than the growth in order quantity and transportation capacity (The change in average order price is shown in Fig. 11). Program C can see all orders, so the main orders are high-value coal orders. The DOA can only select orders from the pending approval pool, which is equivalent to local optimization. Therefore, the average order price is lower than that of Program C. The order accepted under Program A is unrelated to the order, so its order price remains stable around the average value.

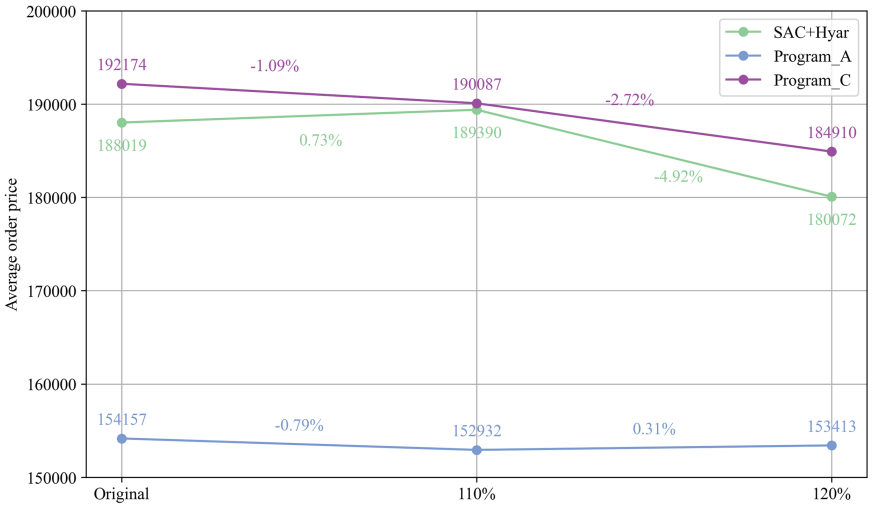

Figure 11: The change of average order price when the transportation capacity increases by 10% and 20%

5 Conclusion and Future Work

This study formulates decision-making objectives from the perspectives of improving client satisfaction and stabilizing the long-term revenue of transportation enterprises. It proposes the "on-time delivery satisfaction" metric to enhance the clarity of delivery timeframes, while introducing client lifetime value to ensure stable long-term corporate earnings. Given the sequential nature of order decisions, a DRL algorithm with sequential decision-making capabilities is employed for modeling.

This article formulates decision-making objectives from the perspective of enhancing client satisfaction and stabilizing the long-term earnings of the carrier enterprise. It proposes the "delivery-time satisfaction" metric to make delivery times clearer, and introduces client value to ensure the stability of the enterprise's long-term earnings. Given the sequential nature of order decision-making, this article employs a deep reinforcement learning algorithm with sequential decision-making capabilities for modeling and solving.

Dynamic order approval is a scheme with an unfixed approval interval, which prioritizes accepting high-value orders while reducing order loss. The model was tested using actual operational data from Guoneng Railway in March 2024, and the results showed that the SAC+Hyar algorithm had a shorter calculation time compared to the traditional ant colony algorithm. The dynamic approval process can effectively reduce order loss and increase total revenue. Given that the transportation capacity was not saturated in March, further construction of medium and large-scale simulation datasets to simulate tight transportation capacity scenarios showed that the dynamic approval strategy improved revenue by 2.0% and 3.3% compared to the static mode.

The original decision variable of the model is the received outcome, but to check the validity of the decision, multiple discrete auxiliary decision variables are introduced. This expands the decision space of the agent. To enhance the convergence efficiency of the model, some discrete auxiliary variables are relaxed to continuous variables, transforming some discrete actions into continuous actions, thereby reducing the output dimension of the deep neural network. By combining the Hyar algorithm to accelerate the training of the deep model, approximately 13.7% of training time is saved. Although

the reinforcement learning model requires more training time compared to the heuristic algorithm, the inference phase takes about 16.9% of the time required by the heuristic algorithm.

The purpose of introducing the customer value function into dynamic order approval is to increase the long-term earnings of enterprises. Limited by the amount of data (one month's data), it is impossible to make a quantitative assessment of the long-term impact of the enterprise. Subsequent research can assess the potential impact of different schemes from the perspective of demand forecasting, customer survey, system robustness, etc. In order to verify the effectiveness of the dynamic approval method in the simulation experiment, the number of selected stations is smaller, which reduces the difficulty of the solution. When the solution space increases, the training and solution time of a single agent will increase exponentially. Transform the single agent framework into the multi-agent framework, reduce the action space of a single agent, realize a parallel solution, and reduce the solution time. At the same time, multi-agent systems can introduce more complex strategies to meet more complex requirements.

In order to guarantee the time limit of goods arrival, the order approval stage needs to query the stations, sections, planned lines, etc., which are all related to the road network information. Further research can focus on the storage structure and representation of the railway. Whether it is an ant colony algorithm or reinforcement learning, the model needs to repeatedly query the road network information. An efficient road network storage structure can reduce the time cost of reading data. The order decision needs to plan the freight transportation route, and the actual railway network topology is complex. The graph neural network is used to represent the road network, so that the reinforcement learning agent can understand the road network, reduce the scale of the neural network, and accelerate model training.

Acknowledgement: Not applicable.

Funding Statement: This research was supported by the National Natural Science Foundation of China (Project No. 52172321), China Energy Shuohuang Railway Co., Ltd. Technology Innovation Program (Project No. KYL202507-0143), Sichuan Province Science and Technology Innovation Talent Project (2024JDRC0020). Sichuan Science and Technology Program(2025YFHZ0328), China Railway Shanghai Bureau Group Co., Ltd. Science and Technology Program (2025037).

Author Contributions: The authors confirm contribution to the paper as follows: conceptualization: Meng Wang, Renjie Liu and Jingchun Geng; data collection: Meng Wang and Jinshan Pan; code writing: Renjie Liu, Meng Wang and Jinshan Pan; methodology: Meng Wang, Shaoquan Ni and Jingchun Geng; supervision: Shaoquan Ni and Jingchun Geng; funding acquisition: Jingchun Geng and Meng Wang. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding author, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cheng JY. Business process reengineering of railway logistics system based on customer demand analysis [master's thesis]. Chengdu, China: Southwest Jiaotong University; 2018. (In Chinese).
2. Li YT. Study on the evaluation system of railway freight service quality under real goods [master's thesis]. Chengdu, China: Southwest Jiaotong University; 2018. (In Chinese).
3. Wu X, Wang L. Research on railway freight marketing strategy based on modern logistics concept. Railw Freight. 2010;28(3):4. (In Chinese). doi:10.2495/itie20130541.
4. Geng Z. Analysis of railway freight products and marketing strategies. Railw Transp Econ. 2010;32:4. (In Chinese). doi:10.3969/j.issn.1003-1421.2010.05.011.
5. Godwin T, Gopalan R, Narendran TT. Estimating order delivery times and fleet capacity in freight rail networks: part I—simulation modelling. Int J Oper Res. 2015;24(3):329. doi:10.1504/ijor.2015.072232.
6. Godwin T, Gopalan R, Narendran TT. Estimating order delivery times and fleet capacity in freight rail networks: part II—analytic approximation. Int J Oper Res. 2015;24(4):369. doi:10.1504/ijor.2015.072722.
7. Ghorpade T, Rangaraj N. Order first split second heuristic for alternative routing strategy for freight railways. Transp Policy. 2022;124:139–48. doi:10.1016/j.tranpol.2019.10.010.
8. Ju HR. Research of the railway freight service quality management based on "real-freight system" [master's thesis]. Chengdu, China: Southwest Jiaotong University; 2016. (In Chinese).
9. John Oyewole G, Adetunji O. A hybrid algorithm to solve the fixed charge solid location and transportation problem. Eng Herit J. 2021;5(1):1–11. doi:10.26480/gwk.01.2021.01.11.
10. Chen S, Qiu X, Tan X, Fang Z, Jin Y. A model-based hybrid soft actor-critic deep reinforcement learning algorithm for optimal ventilator settings. Inf Sci. 2022;611:47–64. doi:10.1016/j.ins.2022.08.028.
11. Wang H, Liu Z, Han Z, Wu Y, Liu D. Rapid adaptation for active pantograph control in high-speed railway via deep meta reinforcement learning. IEEE Trans Cybern. 2024;54(5):2811–23. doi:10.1109/TCYB.2023.3271900.
12. Li H, Hai M. PortfolioZero: a stock portfolio model based on deep reinforcement learning. Appl Soft Comput. 2025;183:113578. doi:10.1016/j.asoc.2025.113578.
13. Anoop KP, Panicker VV, Siby J, Aryadutt CS. Multi-objective optimization algorithms for freight allocation in a food grain supply chain. Appl Soft Comput. 2025;171(12):112729. doi:10.1016/j.asoc.2025.112729.
14. Fan J, Li M, Sun Y, Chen P. DRLAttack: a deep reinforcement learning-based framework for data poisoning attack on collaborative filtering algorithms. Appl Sci. 2025;15(10):5461. doi:10.3390/app15105461.
15. Brezulianu A, Geman O, Popa IV. Artificial intelligence component of the FERODATA AI engine to optimize the assignment of rail freight locomotive drivers. Appl Sci. 2023;13(20):11516. doi:10.3390/app132011516.
16. Wang H, Liu Z, Hu G, Wang X, Han Z. Offline meta-reinforcement learning for active pantograph control in high-speed railways. IEEE Trans Ind Inform. 2024;20(8):10669–79. doi:10.1109/TII.2024.3394554.
17. Liu E, Zhan S, Zhu Y, Lin Z, Wang D. Online multi-modal evacuation during passenger flow outburst in urban transit system: a heterogeneous multi-agent reinforcement learning framework. Transp Res Part E Logist Transp Rev. 2025;204(2):104411. doi:10.1016/j.tre.2025.104411.
18. Jiang L, Shen Z, Liu R, Wang X, Pan J. Two-stage heuristic algorithm for dynamic train operation plan of high-speed express freight train. J Inf Hiding Multimed Signal Process. 2024;9(1):536–50.
19. Yu Q, An L, Yu Q, Liu X, Pu F, Ni S. NSGA II for configuration optimization of cold chain logistic network with fully shared facilities and equipments. J Netw Intell. 2023;8(4):1497–516.
20. Coskun S, Yazar O, Zhang F, Li L, Huang C, Karimi HR. A multi-objective hierarchical deep reinforcement learning algorithm for connected and automated HEVs energy management. Control Eng Pract. 2024;153(2):106104. doi:10.1016/j.conengprac.2024.106104.
21. Zheng L, Chen H, Zou Y. A deep reinforcement learning-based urban traffic control model for vehicle-to-everything ecosystem. J Adv Transp. 2025;2025(1):5579549. doi:10.1155/atr/5579549.

22. Park Y, Jun W, Lee S. A comparative study of deep reinforcement learning algorithms for urban autonomous driving: addressing the geographic and regulatory challenges in CARLA. Appl Sci. 2025;15(12):6838. doi:10.3390/app15126838.
23. Hu J, Wang C. A deep reinforcement-learning-based route optimization model for multi-compartment cold chain distribution. Mathematics. 2025;13(13):2039. doi:10.3390/math13132039.
24. Brucker P, Hurink J, Rolfes T. Routing of railway carriages. J Glob Optim. 2003;27(2):313–32. doi:10.1023/A:1024843208074.
25. Krüger NA, Vierth I. Precautionary and operational costs of freight train delays: a case study of a Swedish grocery company. Eur Transp Res Rev. 2015;7(1):6. doi:10.1007/s12544-015-0155-7.
26. Stahl HK, Matzler K, Hinterhuber HH. Linking customer lifetime value with shareholder value. Ind Mark Manag. 2003;32(4):267–79. doi:10.1016/S0019-8501(02)00188-8.
27. Jia QX. Comprehensive evaluation of railway freight customers and optimal allocation of resources [master's thesis]. Lanzhou, China: Lanzhou Jiaotong University; 2024. (In Chinese).
28. Feng F, Yang L, Lan D. Order-parameter model for synergetic theory-based railway freight system and evolution in China. Promet Traffic Transp. 2013;25(3):195–207. doi:10.7307/ptt.v25i3.307.
29. Haarnoja T, Zhou A, Hartikainen K, Tucker G, Ha S, Tan J, et al. Soft actor-critic algorithms and applications. arXiv:1812.05905. 2018.
30. Haarnoja T, Zhou A, Abbeel P, Levine S. Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor. PMLR. 2018;80:1861–70. doi:10.1109/icra48891.2023.10161395.
31. Chandak Y, Theocharous G, Kostas J, Jordan S, Thomas P. Learning action representations for reinforcement learning. Int Conf Mach Learn PMLR. 2019;97:941–50.
32. Li B, Tang H, Zheng Y, Hao J, Li P, Wang Z, et al. HyAR: addressing discrete-continuous action reinforcement learning via hybrid action representation. arXiv:2109.05490. 2022.
33. Hao H, Xu C, Zhang W, Yang S, Muntean GM. Joint task offloading, resource allocation, and trajectory design for multi-UAV cooperative edge computing with task priority. IEEE Trans Mob Comput. 2024;23(9):8649–63. doi:10.1109/tmc.2024.3350078.
34. Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-learning. Proc AAAI Conf Artif Intell. 2016;30(1):2094–100. doi:10.1609/aaai.v30i1.10295.
35. Zhou H, Zhou L, Guo B, Bai Z, Wang Z, Yang L. A scheduling approach for the combination scheme and train timetable of a heavy-haul railway. Mathematics. 2021;9(23):3068. doi:10.3390/math9233068.