Proceedings of the 4th International Conference on Management Research and Economic Development
DOI: 10.54254/2754-1169/2026.32833

Construction of a Dynamic Scheduling and Prediction Model for Multi-Node Cargo Turnover Data Feature Fitting in Distributed Intelligent Warehousing

Yu Huang

Master of business, Monash University, Melbourne, Australia
yhua0386@student.momash.edu

Abstract. Compared with conventional algorithms that model spatiotemporal coupling features of cargo turnover data, for example, between fitting and scheduling, lack of multi-node collaboration, and difficulty in porting algorithms to edge computing, this study developed a dynamic scheduling and prediction model based on cargo turnover information feature fitting. This study designed an explicit model for multi-node cargo turnover in which data are collected and preprocessed, mine spatial-temporal features are mined, and multi-dimensional constraint modeling is performed. The study employs an attention-based spatiotepitrograph convolutional network and a deep reinforcement learning scheduling module based on near-end policy optimization to achieve end-to-end closed-loop joint feature fitting and dynamic scheduling, and employs a cloud-edge dual-layer collaborative deployment framework to reduce model computational cost. Simulation results show that compared to mainstream algorithms, this model reduces the fitting error, order fulfillment rate, and turnover efficiency, meets the industrial real-time needs of single-round decision delay, and has high model stability and scalability. The model solves the problem of disconnect between distributed warehouse scheduling and data fitting, and provides excellent advantages for multi-node collaborative scheduling and edge deployment, with reliable technical support for distributed intelligent warehousing.

Keywords: distributed intelligent warehousing, multi-node collaboration, cargo turnover data, feature fitting, and dynamic scheduling prediction

1. Introduction

With the recent development of Industry 4.0 and intelligent manufacturing, distributed multi-node warehousing has become an essential component of modern logistics systems. Due to a hierarchical layout of central warehouses and forward warehouses, it can efficiently storage and expedite delivery of goods and thus considerably reduces the logistics cost and the order fulfillment time. According to the "2025 China Intelligent Warehousing Industry Development Report" by the China Warehousing & Distribution Association, in 2025, the penetration rate of distributed multi nodes in three core areas of e-commerce, automobile manufacturing, and fast-moving consumer goods is 78.3%, 65.7%, and 61.2%, up 31.5%, 27.8%, and 24.3% compared to 2022, respectively. Distributed

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2754-1169/2026.32833

warehousing is the mainstream layout model in the logistics industry. The inventory turnover data of distributed nodes show a strong spatiotemporal coupling, non-stationarity, heterogeneity, and multi-nodes correlation [1]. In the temporal dimension, turnover data are impacted by order cycles, holidays, major promotional events, and high trends, periodicity, and sudden changes [2]. In the spatial dimension, the turnover volume of different nodes has high spatial autocorrelation; replenishment and restocking of neighboring nodes directly affect the total network turnover efficiency, and fluctuations in the turnover of a single node can propagate to the entire system through network coupling relationships. Current mainstream warehouse scheduling algorithms are generic to single-node scenarios and suffer from poor adaptability to distributed multi-node scenarios [3]. According to the "25 Industrial Logistics Automation Implementation White Paper," the actual industrial deployment rate of existing intelligent scheduling algorithms in small and medium-sized warehouses in China is less than 22%. Key limitations include high overhead (i.e., they cannot be used in the limited computing power of edge warehouse nodes, local node congestion, and network-wide resource mismatch), lack of multi-network collaboration to multi-node coordination, where prediction results only serve as the reference input and are not included in the closed-loop optimization of scheduling decisions for real-time turnover variations [4].

Distributed warehouse scheduling and turnover prediction have been extensively studied in both the United States and worldwide. In the United Kingdom, in 2023, a distributed dynamic scheduling based on deep reinforcement learning for warehouses by dynamically scheduling orders among multiple nodes was proposed by MIT. However, it did not consider the spatiotemporal coupling features of the cargo turnover data, and the prediction accuracy was inadequate, and the scheduling approach was weak. In 2024, a multi-node cargo turnover prediction model based on time series decomposition was proposed by the Fraunhofer Institute for Logistics in Germany, and the long-term turnover volumes were precisely fitted [5]. The model only worked for the prediction step; it did not consider scheduling decisions; thus, it did not realize an optimal warehouse operation. In 2025, in IEEE Transactions on Industrial Informatics, a multi-node warehouse turnover prediction method based on graph neural networks and mining the spatial correlation features of multiple nodes was proposed in China. The model is highly complicated and requires more than 1 s to process a decision in real time. A team from Southeast University in 2024 proposed a distributed warehouse multi-AGV collaborative scheduling algorithm that optimized the operational efficiency within a single node but did not include cross-node CSC and only obtained limited gains in network resource utilization [6]. In 2024 a team from East Southeast University proposed a warehouse cargo turnover predictive model based upon LSTM and obtained high-Precision fitting of single node turnover data, but ignored spatial correlation properties of multiple node which caused a significant increase in fitting error in distributed situations [7]. In 2025, a distributed warehouse optimization based on computer-integrated manufacturing systems was developed in the manner of prediction and scheduling. Based on a serial architecture, the prediction and scheduler modules did not achieve end-to-end joint optimization, and their fit and scheduling were not equivalent, which resulted in a gap between fitting and scheduling [8]. Currently, there are three main issues with distributed warehouse scheduling and turnover prediction. Most studies split prediction and scheduling into two independent steps, and the fitting results are not implemented in real-time state updates of scheduling decisions [9,10]. Consequently, no complete clustering of the spatiotemporal coupling features of the entire process is considered.

To address these problems, this study presents the following core research: (1) a standard modeling for multi-node cargo turnover data in distributed intelligent warehousing, including data collection and preprocessing, spatiotemporal feature analysis, and modeling fitting goals and

DOI: 10.54254/2754-1169/2026.32833

constraints to solve the standardization problem of heterogeneous data across multiple nodes; (2) an end-to-end prediction model with spatiotemporal feature fitting and dynamic scheduling; (3) an attention-based spatiotemporal graph convolutional fitting module and a deep reinforcement learning scheduling decision module to jointly optimize fitting and scheduling; (4) construction of a distributed multi-nanoscale warehousing simulation experimental environment and selection of mainstream algorithms for testing; and (5) verification of the effectiveness and industrial value of the proposed model based on fitting accuracy, scheduling performance, cost, and stability.

2. Distributed intelligent warehousing multi-node cargo turnover data modeling

2.1. Multi-node cargo turnover data collection and preprocessing

To facilitate the collection and standardization of heterogeneous data from multiple nodes in distributed warehousing, this study proposed four primary data collection dimensional dimensions to account for all elements of the behavior of the entire network turnover. spatial dimension: node location, maximum node capacity, safety stock threshold, distance between nodes, node route time, and coverage; temporal dimension: cargo inbound time, outbound time, storage time, turnover frequency, and order fulfillment cycle; business dimension: volume and weight, order priority, replenishment cycle, and cross-node cargo transfer record; status dimension: real-time load of node equipment, aisle congestion, sorting time, equipment failure rate, and transportation resource availability. missing values, outliers, and heterogeneous processing: This study designed a three-level preprocessing algorithm. Missing value processing: This study adopted a more spatiotemporal K nearest neighbor interpolation algorithm and the historical time series of the target node as a priori time series for adjacent nodes to fill missing values (to avoid accumulating any errors of a single interpolation). Outlier denoising: This study used a wavelet threshold denoising algorithm to remove abnormal fluctuation noise caused by large-scale promotions and equipment failures, while retaining the main trends and periodic features of the turnover data. Normalization processing: Min-Max normalization was used to map the heterogeneous original data to [0,1] with no difference in dimension regarding the fit accuracy. The normalization formula is shown in (1):

$$x_{i,j}^{*} = \frac{x_{i,j} - \min(x_j)}{\max(x_j) - \min(x_j)} \tag{1}$$

$x_{i,j}^{*}$ is the default value of i sample in j feature dimension; $x_{i,j}$ is its initial value; $\max(x_j)$, $\min(x_j)$ are max and minimum value of all samples in j features dimension, respectively. With the above preprocessing, The paper build a standard freight turnover dataset of all nodes, full cycle and all elements, which is used for feature analysis and fitting modeling.

2.2. Feature analysis of multi-node freight turnover data

From the two main dimensions of time and space, this study considers the features of distributed multi-node cargo turnover data and designs a feature index system [11].In time dimension, cargo turnover may have three main features. Trend: Due to increase in enterprise business size and price changes, the volume may grow or decrease; Periodicity: Due from working day/rest days, weekly replenishment cycle, monthly sales plan, the turnover may stay stable day/weekly and periodically fluctuate; suddenness: due to e-commerce promotion, emergency orders, sudden events and the turnover quantity may show short-term large fluctuations and is non-stationary. Cargo turnover may have two core features.

DOI: 10.54254/2754-1169/2026.32833

Spatial autocorrelation. The turnover volume of adjacent nodes has the same directional variation (that is, if a node has a larger turnover volume, the turnover volume becomes larger for adjacent nodes in the radiation range). The global spatial autocorrelation index can be used to quantify this feature, and the calculation formula is given in (2).

$$I_t = \frac{n\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,j}(y_{i,t}-\bar{y}_t)(y_{j,t}-\bar{y}_t)}{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,j}\sum_{i=1}^{n}(y_{i,t}-\bar{y}_t)^2} \tag{2}$$

In the formula: I_t is the global spatial autocorrelation index of the turnover of multiple nodes at time t, with a value in the range of [-1, 1], and a value greater than 0 indicates the existence of positive spatial autocorrelation; n is the total number of nodes in the distributed warehousing network; $w_{i,j}$ is the element of the spatial weight matrix between node i and node j, with the value being the reciprocal of the transportation distance between nodes, and the closer the distance, the greater the weight; $y_{i,t}$, $y_{j,t}$ are the cargo turnover of node i and node j at time t, respectively; $\bar{y}_t$ is the average turnover of all nodes in the network at time t.

Node heterogeneity: Different nodes have different storage capacities, operational capacities, radiation ranges, and order structures; therefore, node heterogeneity exists in turnover efficiency and fluctuation patterns [12]. Based on the feature analysis above, the paper builds a feature system including four main indicators and 18 secondary indicators, time series features, spatial correlation features, business attribute features, and node status features, for feature fitting.

2.3. Data feature fitting objectives and constraints

With the objective function and constraints of the fitting model, to obtain a smaller fitting error of the multi-node cargo turnover data as well as the actual industrial constraints of distributed warehousing and transportation, the objective function or constraints of this fitting model are defined [13]. The objective is to minimize the fitting error for the turnover volume over the entire network and lifecycle. The objective function is described by (3).

$$\min F = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n}\left|\hat{y}_{i,t} - y_{i,t}\right| \tag{3}$$

In the formula: F is the objective function of fitting error; T is the length of the fitting time window; $\hat{y}_{i,t}$ is the fitted value of the turnover of goods at node i at time t. Based on the actual needs of warehouse operation, four categories of constraints are set:

Time delay limitation: In industrial settings, fits and scheduling decisions must be made in real time, and the calculation time of a round must be longer than the maximum possible delay. The constraint formula is in the form of Equation (4):

$$t_{\text{compute}} \le t_{\max} \tag{4}$$

In the formula: t_{compute} is the total computation time for a single round of fitting and scheduling decision; $t_{\max}$ is the maximum decision delay allowed in the industrial scenario, which is set to 500 ms in this paper.

Node capacity constraint. The capacity turnover of the fitted data must meet the safety stock or maximum capacity of the node to avoid stock-ups or overstocking. Constraint formula (5)

$$L_{i,\min} \le s_{i,t} + \hat{y}_{i,t}^{in} - \hat{y}_{i,t}^{out} \le L_{i,\max} \tag{5}$$

DOI: 10.54254/2754-1169/2026.32833

In the formula: $s_{i,t}$ is the current inventory of node i at time t; $\hat{y}_{i,t}^{in}$, $\hat{y}_{i,t}^{\text{out}}$ are the fitted inbound and outbound turnover of node i at time t, respectively; $L_{i,min}$, $L_{i,\max}$ are the lower limit of safety stock and the upper limit of maximum storage capacity of node i, respectively.

Cross-node collaborative constraints: All cross-node transfers in the entire network must be balanced, and there should be no increase or decrease in goods. The constraint formula is given by Equation (6).

$$\sum_{i=1}^{n} \hat{y}_{i,t}^{\text{transfer}} = 0 \tag{6}$$

In the formula: $\hat{y}_{i,t}^{\text{transfer}}$ " is the cross-node transfer turnover of node i at time t, with positive values for goods transferred in and negative values for goods transferred out.

Computational cost: The fitting model must be limited to the finite computing power of the edge green storage nodes. The computational resource used for a round must be larger than the maximum available value of the node. The constraint formula (7) is expressed as in (8).

$$C_{\text{model}} \leq C_{\text{edge}} \tag{7}$$

In the formula: C_{model} represents the floating-point operation cost (FLOPs) of a single round of model computation; C_{edge} represents the maximum available computing resources of the edge node in the warehouse.

3. Construction of a distributed dynamic scheduling prediction model integrating feature fitting

To solve the problem of the fit versus scheduling disconnect, insufficient multi-node collaboration, and poor algorithm performance from previous work, an end-to-end distributed dynamic scheduling prediction model [14] is proposed. It combines the cargo turnover data feature fitting module with the dynamic scheduling decision module to jointly optimize the fitting accuracy and scheduling performance. In addition, an edge-cloud collaborative distributed deployment model is proposed to optimize the adaptive performance for industrial applications.

3.1. Overall model architecture

The proposed model utilizes a three-step approach–"feature fitting − decision generation − collaborative deployment": the cloud performs global model training, hyperparameter optimization, and full network state monitoring; lightweight fitting and scheduling sub-models are deployed on the edge nodes, where local node data are collected and collected for real-time fit and scheduling decisions; node cooperation occurs between nodes through point-to-point communications [15]. The key feature of the model is embedding the output of the feature fitting module into the state input of the scheduling decision module in real time, updating the input information of the fitting module at the execution of the schedule decision, where the fitting and schedule decision has a closed-loop linkage, achieving end-to-end joint training, which simultaneously leads to fit accuracy and scheduling performance, solving the disconnection problem of traditional serial architectures, and designing a lightweight model and a distributed collaborative mechanism to adapt to the limited computing power of edge nodes and achieve a better industrial deployment rate.

DOI: 10.54254/2754-1169/2026.32833

3.2. Multi-spatiotemporal dimension cargo turnover data feature fitting module

To treat the spatiotemporal coupling feature of multi-node turnover data, the present study introduces a spatiotemporal graph convolutional network (ASTGCN) feature fitting module through an attention mechanism and extracts temporal series features and spatial correlation features of turnover data for more accurate fitting. The proposed module consists of three parts: a spatial convolution layer, a temporal convolution feature, and an attention mechanism. Spatial convolution layer: Graph convolution network (GCN)-based features are automatically extracted from the graph structure data of storage nodes by mining spatial correlation features between nodes. The spatial weight matrix in Equation (2) is the adjacency matrix of the graph to automatically extract multisugar spatial coupling features. Temporal convolution network-based features and nodes are extracted by a causal convolutionally network. The temporal conversional network is applied to the time series data of a single node to extract trend, period, and suddenness features of the turnover data. Long-term time series features are extracts automatically by multi-layer causal convolve. attention mechanism: Our spatiotemic attention module is proposed to assign dynamic weights to features of different time steps and different nodes by considering features and the nodes that contribute to turnover efficiency, eliminate noise due to irrelevant noise, and improve the fitting performance. The input of the fitting module is the preprocessed standardized multi-next-turnover feature data, and the output is the fitted value of the cargo turnover of each node in the entire network in the future time window for accurate input of the scheduling decision module.

3.3. Dynamic scheduling decision module

Using the turnover fitting value output of the feature fitting module as the input, the present study designs a dynamic scheduling decision module based on deep reinforcement learning to facilitate the end-to-end scheduling performance of distributed multi-node warehouses. The present study adopts the proximal policy optimization (PPO) algorithm as the scheduling decision algorithm. Compared to the deep Q-network (DQN) and actor-critic (A2C) based reinforcement learning algorithms, the PPO algorithm has high training stability, high hyperparameter robustness, and is capable of scheduling a distributed multi node warehouse in a systematic manner. Specifically, The paper define the key elements of reinforcement learning:

State Space S : future turnover fitting value of each node output by feature fitting, current inventory of each individual node, equipment load, transportation resource, order queue, and others. state vector $s_t \in S$ state of all network at t. Action Space A : inbound allocation strategy, outbound order fulfillment strategy, node transfer strategy, and transportation resource scheduling strategy of each given node. action vector $a_t \in A$ state of the whole network at time t. Reward Function R : This paper design a reward function of the different types, considering turnover efficiency, fulfillment delay, and transport cost, to achieve global optimal scheduling. The reward function (8)

$$R_t = \omega_1 \eta_t - \omega_2 \tau_t - \omega_3 c_t \tag{8}$$

In the formula: R_t is the immediate reward value of the scheduling decision at time t; η_t is the order fulfillment rate of the entire network at time t; τ_t is the average fulfillment delay of fulfilled orders at time t; c_t is the total transportation cost of cross-node cargo transfer at time t; $\omega_1, \omega_2, \omega_3$ are the weight coefficients of each indicator, all of which are positive and satisfy $\omega_1 + \omega_2 + \omega_3 = 1$, and the weight allocation can be adjusted according to the actual needs of warehouse operations.

DOI: 10.54254/2754-1169/2026.32833

The scheduling decision module receives the turnover fitted value output by the fitting module and the real-time status of the entire network, and the output is the globally optimal multi-node collaborative scheduling strategy. The execution of the scheduling policy is fed back into the data collection module in real time to update the input data of the fitting policy, thereby achieving closed-loop fitting and scheduling optimization.

3.4. Distributed collaborative deployment mechanism

To minimize the edge node deployment rate of existing algorithms, we design a two-layer distributed collaborative deployment scheme with the following two schemes. cloud deployment. The cloud server gathers global model history across the entire network, trains and optimizes global model hyperparameters, and monitors and issues early warnings of network failures. On the same day, the trained and optimized global model parameters are shared with each edge node to automatically update the global model. Edge deployment. Lightweight sub-models are deployed on edge devices at each warehouse node. Model pruning and quantization are performed to constrain the model size and computation time. The sub-model records local node data, preprocessing, and real-time feature fitting, as well as scheduling decisions. Local operational data and model running status are uploaded in the network using local real-time decisions and global collaborative optimization. Inter-node collaboration. Near-edge edge nodes share local inventory and turnover fitting values as well as transportation resource status, which enables cross-node, near-node sharing and collaborative scheduling without the delay of cloud communication and real time performance of scheduling decisions.

3.5. End-to-end model training optimization

To eliminate the gap between the fitting and scheduling of traditional serial architectures, in this study, an end-to-end joint training is designed, which jointly optimizes the fitting accuracy of the feature fitting module and the scheduling performance of the scheduling decision module. Our overall loss function is as following:

$$L_{\text{total}} = \alpha L_{\text{fit}} + \beta L_{\text{schedule}} \tag{9}$$

In the formula: L_{total} is the end-to-end total loss function of the model; L_{fit} is the mean squared error loss of the feature fitting module, used to measure the error between the fitted value and the true value; L_{schedule} is the negative cumulative reward loss of the scheduling decision module, used to measure the performance of the scheduling strategy; α, β are the weight coefficients of the loss function, satisfying $\alpha + \beta = 1$, used to balance the optimization objective of fitting accuracy and scheduling performance.

Modeling is divided into two tasks. Pre-training: The feature fitting module is pre-trained based on past turnover data to achieve high fit and reliable input for scheduling decision module; Joint training: The pre-fitting module and scheduling decision modules are jointly trained in the end-to-end and the total loss function L_{total} is used. Adam optimizer updates the model parameters and improve the fit accuracy and scheduling performance by fitting and scheduling.

DOI: 10.54254/2754-1169/2026.32833

4. Experimental simulation design

4.1. Experimental environment setup

The hardware environment comprised a cloud server and edge nodes. The cloud server had an Intel Xeon 8375C CPU (32 cores), an NVIDIA A100 40G GPU, and 128GB of DDR4 memory. The edge nodes were simulated using an NVIDIA Jetson Xavier NX development board (6-core ARM CPU, 384-core Volta GPU, and 8GB of memory). The execution space was equal to the actual load of warehouse edge nodes in the simulation environment. The experimental environment was Ubuntu 22.04 LTS, the programming language was Python 3.10, the deep learning framework was PyTorch 2.2.0, and the reinforcement learning framework was Gymnasium 0.29.0. The inter-node transportation environment is Simulation with SUMO 1.18.0 traffic simulation software, and the data processing uses Pandas 2. 2.0 and NumPy 1.26.0 libraries. In the experiment of a distributed warehousing scenario, a distributed warehouse network consisting of one central warehouse and eight local forward warehouses with a radius of 300 km was constructed. The amount of transport between nodes was 10–150 km, the maximum storage of each node was 5000–20000 SKUs, and the maximum processing capacity was 1000–5000 orders per day. The routing between nodes utilized box trucks with a speed of 60 80 km/h, which matches the actual distributed warehouse layouts of domestic e-commerce and FMCG industries.

4.2. Experimental design

4.2.1. Experimental dataset

Our experimental dataset comprises of two parts. Public Dataset (2023-2024 distributed warehouse operation dataset provided by JD Logistics: 10 warehouse nodes, 12 months cargo turnover, order data, relocation data, and equipment status data (1.2M valid samples); Simulation Datasets (Based on the statistical nature of the public dataset, The paper simulate a real-world scenario data set, a major promotion event, a multi-node equipment failure, and an extreme weather transportation disruption data set (300 K valid samples). Dataset preprocessing implements the three-level preprocessing proposed in Section 2.2, which splits the dataset into training, validation, and test sets in an 8:1:1 ratio.

4.2.2. Algorithm comparison

To assess the performance of the model presented in this study, four state-of-the-art distributed warehouse scheduling and prediction algorithms were selected:

Algorithm 1: Traditional genetic algorithm (GA) and classical heuristic scheduling algorithm used in warehouse scheduling.

Algorithm 2: Deep reinforcement learning scheduling (DQN) based on DQN only based on the real-time state with no feature fitting module.

Algorithm 3: STGCN prediction + rule scheduling based on spatiotemporal graph convolutional network, prediction results, and manual rules in a serial architecture.

Algorithm 4: GAT + PPO algorithm based on a graph attention network (inference to spatial features) and PPO reinforcement learning scheduling based on a current standard end-to-end scheduling algorithm.

DOI: 10.54254/2754-1169/2026.32833

4.2.3. Evaluation indicators

An evaluation indicator system is built from four dimensions to perform the following.

Fitting accuracy: Mean absolute error (MAE), mean absolute square error (RMSE), and mean absolute percentage error (MAPE). The lower The paper get; the better fitting accuracy will be.

Scheduling performance. Order fulfillment rate, average order turnaround time, and total cross-node relocation costs: High fulfillment rates, shorter turnaround times, and higher relocation costs indicate better scheduling performance.

Computational cost estimates: mean delay per round of decisions, FLOPs, and model memory. Lower values imply a low computational cost, and the model is used for edge use.

Stability measurements: Scheduling performance volatility at different node loads. Lower values indicate better model stability and robustness.

4.3. Experimental results and analysis

Table 1 presents the performance index comparison results of the proposed model and four alternative algorithms for the core evaluation metrics in four dimensions. The proposed model outperformed all alternative algorithms in terms of fitting accuracy. Compared with the existing mainstream GAT+PPO algorithm, the proposed solution reduces MAE, RMSE, and MAPE by 32.73 %, 33.11%, and 28.47 %, respectively, and compared with the current traditional STGCN prediction algorithm, which minimizes MAPE and MAPEs by 41.78%. This is because the proposed methodology introduces a spatiotemporal attention mechanism that can extract spatiotemporal coupling properties of multimode turnover data from a spatiotemporal basis. Through end-to-end joint training, the scheduling goal is passed back to the fitting module to match the fitting accuracy with scheduling requirements.

Table 1. Comparison of comprehensive performance indicators between the proposed model and comparative algorithms

Evaluation Dimensions	Specific indicators	GA	DQN	STGCN+ Rule-based scheduling	GAT+PPO	This article's model
Fitting accuracy	MAE (Single/Daily)	-	128.37	87.62	72.45	48.73
	RMSE (Single/Day)	-	167.52	112.34	94.61	63.28
	MAPE(%)	-	14.82	8.76	7.13	5.10
Scheduling performance	Order fulfillment rate (%)	87.52	92.23	94.16	96.38	98.72
	Average turnaround time (h)	14.37	11.82	10.25	9.16	7.83
	Total cost of goods transfer (ten thousand yuan/month)	128.73	106.25	94.62	87.34	71.28
Computational overhead	Single-round decision delay (ms)	128.36	287.45	412.73	622.18	328.42
	Single-wheel FLOPs(G)	0.32	2.17	3.46	5.82	2.03

DOI: 10.54254/2754-1169/2026.32833

Table 1. (continued)

Stability	Performance volatility (%)	12.73	8.62	6.34	4.27	2.10

Figure. 1 shows the MAPE variation curves of each algorithm under different fitting time windows of 1–24 h. The trend is that the time window increases, but the MAPE of the proposed model is below 8%, which is far lower than that of the comparison algorithms. For 24 h, the MAPE is below 7.82%, and the MAPE for STGCN is above 16.35%, indicating that the proposed methodology fits long-term time series and provides stable and accurate inputs to long-cycle scheduling.

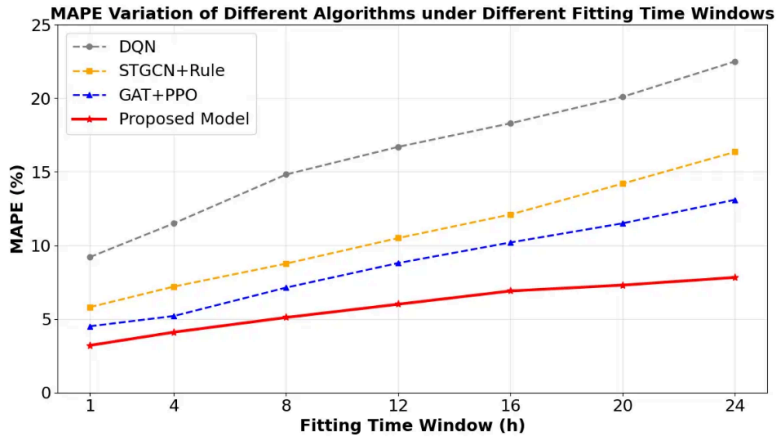


Figure 1. MAPE variation curves of different algorithms under different fitting time windows from 1h to 24h

From a scheduling perspective, scheduling performance of the proposed model is highly superior to that obtained from the GAT+PPO performance. Compared with the GA-based GAT+, the proposed algorithm achieves a 2.34% order fulfillment rate, a 14.52% average turnaround time, and a 12.39% total cost of cross-node inventory transfer. Compared with GA, the proposed method achieves a 11.2% order completion rate, an average turnaround rate, an 18.51% cost of inventory transfer, and a total cost at cross-locations of 44.63%. The main reason for this is that the proposed approach achieves closed-loop optimal fitting and scheduling, early scheduling optimization based on the correct turnaround volume fitting results, and optimal utilization of network resources through a multi-node collaborative approach.

Figure 2 shows the order fulfillment rate variations for different algorithms for different node load rates ranging from 30% to 100%. From the curves, for low node load rates (<60%), all algorithms satisfy less than 90%, and for high load rates (~80%), the fulfillment rates of the compared algorithms all drop; however, the fulfillment rate of the proposed model is still below 95%, even at 100% full load, and the fulfillment time is higher than 96.27%, indicating that the proposed model is robust in high-load cases.

DOI: 10.54254/2754-1169/2026.32833

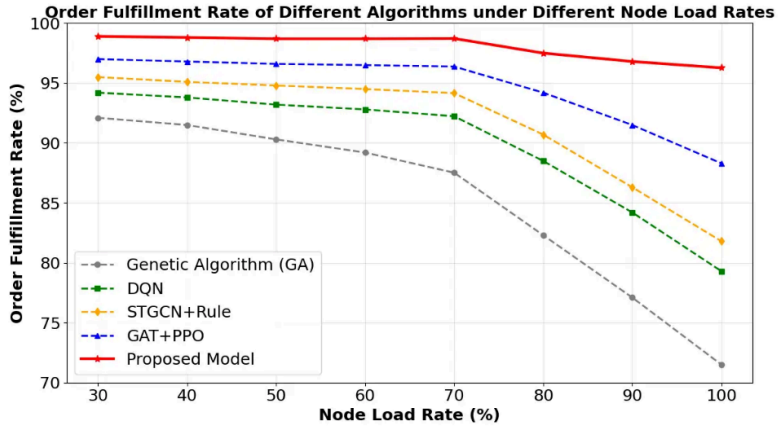


Figure 2. Order fulfillment rate variation of different algorithms under different node load rates of 30%-100%

Figure 3 shows the single-round decision latency variation curves of different algorithms under different numbers of nodes from 2 to 20. As observed from the curves, the decision latency of the two methods increases with the number of nodes. When the number of nodes increases to 20, the latency of GAT + PPO grows to 1200 ms, whereas the growth in the proposed model is slow. Even when the number of nodes increases to 20, the delay is still 487.36ms, which is below 500 ms, suggesting that the proposed method is a good multi-node scalable.

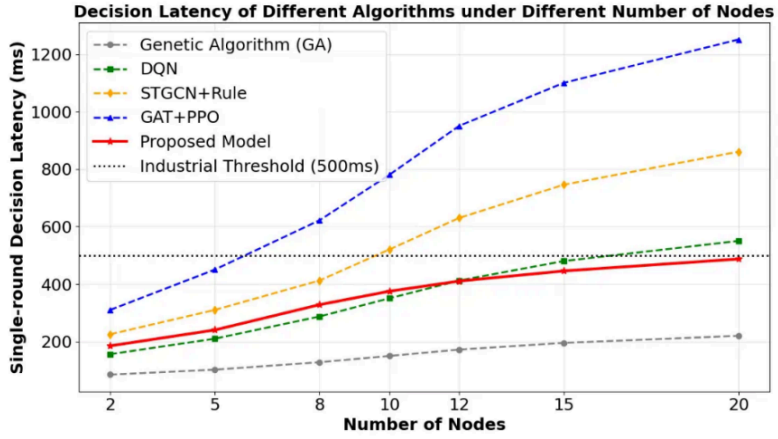


Figure 3. Changes in single-round decision latency for different algorithms with 2-20 different numbers of nodes

5. Conclusion

The paper developed a generalized modeling system for multi-node cargo turnover data in distributed intelligent warehousing, proposed an end-to-end prediction model based on spatiotemporal feature fitting and dynamic scheduling, achieved accurate turnover data fitting via attention spatiotemporal graph convolutional networks, performed multi-node collaborative scheduling decisions based on deep reinforcement learning, and improved the industrial application of the model with a cloud-edge collaborative architecture. Experiments demonstrated that the model outperformed state-of-the-art algorithms in terms of fitting accuracy, scheduling, computational overhead, and stability, and could cope with the multitasking of the multi-node cooperative operation in distributed warehousing. However, there are still gaps. The model is unable to adapt to

DOI: 10.54254/2754-1169/2026.32833

extreme cases, such as multi-neural concurrent failures and transportation link interruptions. In addition to e-commerce and FMCG fields, it does not generalize well to manufacturing, cold chain, and other scenarios. Furthermore, collaborative scheduling performance must be optimized in large-scale warehousing networks with over 100 nodes. In the future, the study will develop few-shot and transfer learning to improve the model's adaptability to extreme scenarios, extend multi-industry warehousing datasets for generalization, unify model compression and quantization to further reduce algorithm weight, and include digital twin technology to build a real-time warehousing mapping system for closed-loop iteration and engineering applications of the approach in more cases.

References

[1] Zakaria, I. H., Lyn, J., Azizul, P. M., Mazlan, F. A. A., & Bakri, A. (2024). Smart warehousing solutions: enhancing efficiency and sustainability in logistics and transportation. Journal on Technical and Vocational Education, 9(2), 409-420.
[2] Aksentijević, S., Tijan, E., Kapidani, N., & Žgaljić, D. (2022). Dynamic smart numbering of modular cargo containers. Sustainability, 14(14), 8548.
[3] Bai, K., Rosli, R. B., Duraipandi, O., Salman, S. A., & Ali, D. A. (2025). Smart Transformation of Zhengzhou's Logistics Industry in the Context of Digital Economy: Present Situation, Dilemmas and Advancement Paths. Infrastructure Reports, 1(3), 1-12.
[4] Oleksandr, V. (2025). Digital Transformation of Warehouse Logistics: From Manual Records to Cloud Platforms. Холодная наука, (18), 18-30.
[5] Ge, X., & He, Q. (2025). Research on the Application of AGV in the Smart Logistics Field. Economics & Business Management, 1(3), 75-81.
[6] Li, T., & Liu, S. (2026). Digital twin-based intelligent warehouse thermal environment simulation and automated control optimization. Thermal Science, 30(1 Part A), 41-49.
[7] Yi, W., & Hayashi, K. (2026). Overseas Warehouse Strategy and Information System Advancement in Chinese Cross-border E-commerce. International Information Institute (Tokyo). Information, 29(1), 63-70.
[8] Bai, K., Rosli, R. B., Duraipandi, O., AhmedSalman, S., & Ali, D. A. (2025). Quantifying Smart Logistics Evolution: An Entropy-TOPSIS Assessment of Zhengzhou's 10-Year Transformation (2014-2023). Finance and Trade Dynamics, 1(3), 1-11.
[9] Li, K. (2023). Optimizing warehouse logistics scheduling strategy using soft computing and advanced machine learning techniques. Soft Computing, 27(23), 18077-18092.
[10] Kembro, J., & Norrman, A. (2022). The transformation from manual to smart warehousing: an exploratory study with Swedish retailers. The International Journal of Logistics Management, 33(5), 107-135.
[11] Moskvichenko, I., Stadnik, V., & Kushnir, L. (2024). Improvement of the quality management system in the transport and logistics sector. Baltic Journal of Economic Studies, 10(4), 301-309.
[12] Qian, C. (2025). Assessing the Impact of Forecasting Inaccuracies on Air Cargo Logistics Performance: The Mediating Role of Smart Contracts in China. Sustainable Marketing Practices, 1(2), 1-13.
[13] Ellithy, K., Salah, M., Fahim, I. S., & Shalaby, R. (2024). AGV and Industry 4.0 in warehouses: a comprehensive analysis of existing literature and an innovative framework for flexible automation. The International Journal of Advanced Manufacturing Technology, 134(1), 15-38.
[14] Mambetalin, D., Koshekov, A., Orazbayeva, B., Moldabekov, A., & Keribayeva, T. (2026). Application of machine learning to storage allocation decision making system within air cargo terminals. Acta Logistica, 13(1), 52-65.
[15] Balkan, D., & Akyuz, G. A. (2023). Logistics sector turnover: Forecasting for Turkey, EU27 and EA19 under effects of COVID-19. Logistics, 7(2), 23.