HiLL PUBLISHING

Engineering Advances, 2026, 6(1), 22-26
https://www.hillpublisher.com/journals/ea/
ISSN Online: 2768-7961

Research on Warehouse Capacity Optimization Methods Based on Predictive Modeling

Yiting Hong

Forecasting & Purchasing & QC Department, The Antigua Group, Peoria, AZ 85382, USA.

How to cite this paper: Yiting Hong. (2026) Research on Warehouse Capacity Optimization Methods Based on Predictive Modeling. *Engineering Advances*, 6(1), 22-26.
DOI: 10.26855/ea.2026.03.005

Received: November 30, 2025
Accepted: December 28, 2025
Published: January 29, 2026

***Corresponding author:** Yiting Hong, Forecasting & Purchasing & QC Department, The Antigua Group, Peoria, AZ 85382, USA.

Abstract

The study investigates warehouse capacity optimization under dynamic demand conditions. By introducing predictive modeling, historical data on order volumes, inbound frequency, and inventory fluctuations are analyzed to achieve accurate forecasting and dynamic capacity allocation. A hybrid approach combining time-series analysis and machine learning algorithms is employed to construct the prediction model, while a multi-objective optimization method is used to balance space utilization and operational cost. The research further integrates real-time feedback and adaptive recalibration mechanisms to ensure model robustness under continuous data variation. The results demonstrate that the proposed approach effectively enhances capacity utilization and forecasting accuracy, reduces resource waste, and provides reliable technical support for intelligent warehouse management and sustainable logistics operations, achieving an average 15% improvement in storage utilization and a 12% reduction in energy consumption compared with conventional static allocation strategies.

Keywords

Predictive modeling; Warehouse capacity optimization; Time-series analysis; Machine learning; Multi-objective optimization

Introduction

With the rapid development of global logistics and e-commerce, warehouse systems have become a critical component of supply chains for efficient resource allocation. Capacity planning, as an essential part of warehouse management, directly influences space utilization and operational cost. In environments with frequent demand fluctuations and accelerated product turnover, traditional planning methods based on static parameters or human experience fail to support real-time decision-making. Predictive modeling enables proactive and adaptive capacity allocation through the analysis and extrapolation of historical data, becoming a key technique for modern warehouse optimization. By integrating data-driven forecasting mechanisms with multi-objective optimization algorithms, warehouse resources can be dynamically allocated and operational strategies refined, thereby improving system efficiency and advancing intelligent management capabilities. However, existing studies often treat prediction and optimization separately, limiting their effectiveness in dynamic environments. This research bridges that gap by proposing an integrated predictive-model-based framework that continuously aligns forecasting results with capacity adjustment. Furthermore, the convergence of intelligent prediction and algorithmic optimization establishes a scalable technical foundation for autonomous decision-making and long-term digital transformation within global supply networks.

1. Theoretical Analysis and Modeling Requirements for Warehouse Capacity Optimization

1.1 Characteristics and Constraints of Warehouse Capacity Optimization

Warehouse capacity optimization is a dynamic multi-objective resource allocation problem that requires balancing limited space, fluctuating demand, and operational cost [1]. Variations in inbound and outbound flow create cyclical fluctuations in capacity utilization, while product volume density, retention period, and dispatch frequency determine the spatial structure of storage. Physical constraints such as rack load capacity, aisle width, and stacking height limit usable space, and equipment capability and energy consumption further influence operational efficiency. High-load conditions may cause congestion, whereas low-load conditions lead to idle capacity and unnecessary energy usage. Effective planning requires temporal and spatial coordination. Optimization models typically consider space utilization, turnover time, energy efficiency, and response rate as key indicators. Through weighted parameterization, these objectives are balanced to ensure operational feasibility and stability across different load conditions.

1.2 Predictive Modeling Logic in Capacity Planning

Predictive modeling plays a central role in warehouse capacity optimization by providing forward-looking estimations based on historical data. The model identifies temporal trends and seasonal patterns in order quantities, inbound frequency, and inventory turnover, providing quantitative inputs for dynamic scheduling. Data are smoothed, normalized, and cleansed of anomalies to improve stability. The ARIMA model is suitable for periodic sequences, the LSTM network captures nonlinear and long-term dependencies, and the XGBoost model achieves high generalization performance with high-dimensional data. Model selection depends on data characteristics and real-time requirements, while hybrid strategies can enhance prediction accuracy. The forecasting results serve as key inputs for optimization algorithms, enabling dynamic coordination of spatial allocation, inventory control, and scheduling strategies to improve responsiveness and resource utilization [2].

2. Design of the Warehouse Capacity Optimization Method Based on Predictive Modeling

2.1 Construction and Training Mechanism of the Predictive Model

The predictive model is established based on time-series data to quantitatively forecast future capacity demand through nonlinear mapping. Let the input sequence be X = {x1, x2, ⋯, xt} and the output yt+k; the prediction function is defined as:

$$\hat{Y}_{t+k} = f(W \cdot X_t + b) \tag{1}$$

where W represents the weight matrix, b the bias term, and f (·) the nonlinear activation function. The model is trained by minimizing the mean squared error (MSE) expressed as:

$$L = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

Model parameters are updated using gradient descent. Recurrent units are employed to capture temporal dependencies in capacity fluctuations, and hidden state transitions enable long-term memory of multidimensional features. During training, a sliding-window structure is used to maintain temporal continuity, while normalization eliminates discrepancies in feature scales [3].

For example, in a North American e-commerce logistics center, historical warehouse data from the past 24 hours—including inbound and outbound volumes, order quantities, and inventory turnover rates—were selected as input features to construct an LSTM prediction model. The dataset was divided into training, validation, and testing subsets in a 70:20:10 ratio. The Adam optimizer was used to adjust the learning rate adaptively, with a batch size of 64. During training, the loss curve was monitored in real time to dynamically adjust iteration steps and weight update strategies. After normalization and time-step unfolding, the model generated a four-hour-ahead capacity demand sequence, providing accurate predictive input for subsequent optimization calculations.

2.2 Capacity Optimization Algorithm Design and Objective Function Construction

The capacity optimization algorithm uses the predicted capacity demand sequence as its core input and achieves dynamic balance among space utilization, energy consumption, and operational cost through a multi-objective function. The objective function is formulated as:

$$min\, F = \alpha_1(1 - U) + \alpha_2 C + \alpha_3 E \tag{3}$$

where U represents space utilization, C denotes operational cost, and E indicates energy consumption. The coefficients α_1, α_2, α_3 are normalized weighting factors that regulate the priority of each objective. The optimization is subject to constraints including total storage capacity, rack load-bearing limits, and operation time windows, ensuring that the feasible solution space meets both physical and operational requirements. An improved Particle Swarm Optimization (PSO) algorithm is adopted for computation, using the predicted data as the search reference. Particle positions and velocities are iteratively updated to obtain the optimal capacity allocation. The inertia weight and learning factors are dynamically adjusted during iterations to maintain a balance between global exploration and local convergence. The process terminates when the fitness variation meets the convergence threshold, outputting a capacity scheduling matrix and an energy distribution table [4].

In a Northern European automated warehouse experiment, the system utilized the predicted 24-hour capacity demand curve as the input for the optimization module. The particle swarm was initialized with 80 particles and a maximum of 200 iterations, with the inertia weight linearly decreasing from 0.9 to 0.4. Each particle represented a distinct combination of capacity allocation and energy scheduling. The fitness function, defined by the objective function, evaluated each particle's overall score. During iterations, the algorithm continuously refined the space allocation ratio, energy distribution, and operation time slots until the convergence criteria were satisfied. The final output included a capacity allocation matrix and an energy configuration dataset, providing executable scheduling guidance for the warehouse system. The optimization and prediction modules interacted through real-time data interfaces, forming an integrated dynamic decision framework for coordinated capacity planning.

2.3 Prediction-Optimization Collaborative Integration Mechanism

The collaborative mechanism between prediction and optimization establishes a bidirectional feedback architecture, forming a closed-loop control between capacity forecasting and optimization computation. The system uses the predicted capacity demand vector as the initial input, while the optimization algorithm generates corresponding allocation schemes that are fed back to the prediction module for model calibration. The mechanism operates through dynamic parameter updating and adaptive model adjustment. When prediction deviation exceeds a predefined threshold, the system automatically triggers retraining to modify model weights and time-window lengths, maintaining the forecasting accuracy. In parallel, the optimization module continuously adjusts constraint parameters based on real-time operational data, enabling rolling planning across multiple scheduling periods. Both modules operate within a unified data interface to synchronize information flow and parameter updates.

In a Western European intelligent warehouse system, the prediction module produces short-term capacity demand sequences every two hours, and the optimization module performs iterative updates using the latest forecasts. Message queues manage communication between forecasting outputs and optimization parameters, while asynchronous computation enables both modules to run concurrently. Optimization results are used to correct residual errors in the predictive model and refine input features for the next forecasting cycle. Parameter exchange through shared memory ensures consistency in timing and computation. This integrated mechanism connects forecasting and optimization into a continuous adaptive loop, providing the operational foundation for the performance validation described in the following section [5].

3. Model Validation and Performance Evaluation Analysis

3.1 Warehouse Case Data and Experimental Environment Setup

The experiment is conducted using operational data from a large automated warehouse in Europe to validate the predictive modeling and capacity optimization algorithms. The dataset, collected from the warehouse management and energy monitoring systems, includes inbound and outbound volumes, order density, inventory turnover rates, and energy consumption records. Data were sampled every 30 minutes over a 30-day period, resulting in more than 12,000 valid records. After standardizing the time series and interpolating missing values, the processed data were used to construct temporal features of capacity demand variations. The experimental environment consisted of high-performance computing nodes and a distributed database. Model training was implemented using Python and TensorFlow frameworks on servers equipped with 32-core CPUs, 128 GB RAM, and dual A100 GPUs to support parallel computation. The optimization algorithm ran on the same platform, calling prediction results in real time through interface communication. A unified data bus ensured seamless integration between the prediction and optimization modules, providing a reproducible setup for subsequent performance comparison and parameter validation.

3.2 Comparative Analysis of Predictive Model Performance and Results

To evaluate the applicability of predictive modeling in capacity optimization, three representative models were compared: a traditional ARIMA model, an XGBoost-based regression model, and a Long Short-Term Memory

(LSTM) network. All models were trained on the same dataset and input features, using a time-step length of 24 and a prediction horizon of four hours. Performance was assessed using mean squared error (MSE), mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). The training-testing split was 70:30, with 200 training epochs, a batch size of 64, and an initial learning rate of 0.001 optimized by the Adam optimizer [6].

The ARIMA model parameters were determined by the Akaike Information Criterion with optimal orders (p, d, q) = (2, 1, 2), capturing long-term trends but responding slowly to abrupt fluctuations. The XGBoost model employed 500 trees with a maximum depth of six and early-stopping regularization. The LSTM model contained two hidden layers with 128 neurons each and a dropout rate of 0.2 to enhance sequence memory. Results indicated that the LSTM achieved the lowest errors and highest stability under highly dynamic conditions. The convergence curve flattened after approximately 50 epochs, with training and validation losses decreasing synchronously, indicating stable learning without overfitting. Each experiment was repeated five times to ensure consistency.

Table 1. Performance Comparison of Predictive Models

Model Type	MSE	MAE	RMSE	MAPE (%)
ARIMA	0.0248	0.1123	0.1576	4.31
XGBoost	0.0172	0.0917	0.1312	3.27
LSTM	0.0095	0.0674	0.0975	2.14

The comparison shows that the LSTM outperforms the other models across all metrics, with MSE and RMSE values reduced by approximately 40-50%. XGBoost maintains strong stability for mid-term forecasting tasks and generalizes well under volatile conditions, while ARIMA performs reasonably in stationary intervals but struggles with nonlinear patterns. The verification process demonstrates that deep recurrent models are more capable of capturing dynamic variations in capacity demand, producing smoother and more concentrated error distributions that provide reliable input for subsequent optimization algorithms [7].

3.3 Validation and Evaluation of Capacity Optimization Performance

To verify the adaptability and stability of the optimization algorithm under different operational conditions, the capacity demand sequences predicted by the LSTM model were integrated into the optimization module for dynamic allocation experiments. The improved Particle Swarm Optimization (PSO) algorithm was employed, with objective function coefficients set to α_1=0.5, α_2=0.3, and α_3=0.2, and inertia weight linearly decreasing from 0.9 to 0.4. Three typical scenarios—peak load, steady state, and low load—were simulated. Each scenario was iterated 200 times while recording key indicators such as space utilization, average operation duration, energy consumption, and convergence precision to evaluate the algorithm's responsiveness and convergence performance under multi-constraint conditions.

The experimental results showed that the algorithm effectively adjusted slot allocation and task sequencing during high-load conditions, maintaining an average space utilization rate above 92%, approximately 15% higher than that of static allocation strategies. Under low-load conditions, the constraint-penalty and adaptive energy scheduling mechanisms significantly reduced idle equipment operation, resulting in an average 12% reduction in total energy consumption. The convergence curve stabilized after about 120 iterations, with fitness variation below 1.5%, indicating robust convergence and solution stability. Analysis of the capacity scheduling matrix revealed that the system dynamically adjusted storage unit occupancy ratios based on real-time predictions and optimized operational paths for energy efficiency, achieving dual coordination between spatial configuration and task execution while maintaining multi-objective balance in time, space, and energy dimensions.

Multi-cycle validation further demonstrated that the integrated prediction-optimization framework maintained high scheduling consistency and computational stability during continuous operation. The rolling update mechanism of the prediction module effectively mitigated lag and oscillation issues common in static optimization systems, allowing stable scheduling and energy control even under high-frequency demand fluctuations. Throughout the 30-day experimental period, the warehouse system operated smoothly without task congestion or resource idling [8]. These findings highlight the practical significance of integrating predictive analytics with optimization algorithms, demonstrating that data-driven coordination can substantially improve warehouse resilience and decision autonomy. The algorithm achieved dynamic coordination among throughput efficiency, energy utilization, and operational tempo, confirming the scalability and engineering applicability of the predictive-model-based capacity optimization framework in complex dynamic environments and providing a reliable technical foundation for long-term intelligent warehousing and multi-warehouse coordination.

4. Conclusion

This study established a warehouse capacity optimization framework driven by predictive modeling, achieving dynamic coordination between demand forecasting and resource allocation. By integrating time-series prediction with multi-objective optimization, the system demonstrated strong stability and adaptability under varying load conditions. Experimental validation confirmed that the framework maintained high space utilization and balanced energy consumption in fluctuating environments, providing algorithmic support for real-time decision-making in intelligent warehousing. The proposed approach offers a reusable technical pathway for integrating forecasting and optimization methods, laying a theoretical foundation for future applications in multi-warehouse coordination and cross-regional logistics systems, with future work aiming to extend this framework toward real-time capacity adjustment across interconnected warehouse networks.

References

[1] Fu Y, Li J, Cui L, et al. Optimization Study of Electric-Hydrogen Hybrid Energy Storage Capacity in Warehouse Park for Smart Grid. Annals of Data Science. 2025 Feb 2 [Epub ahead of print]. doi:10.1007/S40745-025-00649-1.

[2] Shinji I, Masaki K, Ryu N, et al. A Proposal of Path Planning for Robots in Warehouses by Model Predictive Control without Using Global Paths. IFAC PapersOnline. 2022;55(37):573-8. doi:10.1016/J.IFACOL.2022.11.244.

[3] Anonymous. MHS launches new software, predictive maintenance solutions for warehouses. Modern Materials Handling. 2020;75(4):36.

[4] Ansari M, Smith SJ. Warehouse Operations Data Structure (WODS): A data structure developed for warehouse operations modeling. Computers & Industrial Engineering. 2017;112:11-9. doi:10.1016/j.cie.2017.08.009.

[5] Nejad YMA, Ali A, Mohsen A, et al. A Highly Effective Optimization Approach for Managing Reverse Warehouse System Capacity Across Diverse Scenarios. Process Integration and Optimization for Sustainability. 2023;8(2):455-71. doi:10.1007/S41660-023-00388-X.

[6] Xian-hao X, Wei-hong D, Hongxia P. A Robust Optimization of Capacity Allocation Policies in the Third-Party Warehouse. Mathematical Problems in Engineering. 2015;2015:810798.1-810798.10. doi:10.1155/2015/810798.

[7] Yi G, Reklaitis VG. Adaptive model predictive inventory controller for multiproduct warehouse system. Automatica. 2014;50(9):2245-53. doi:10.1016/j.automatica.2014.07.022.

[8] Wang S, Lee W, Chang C. Modeling the consignment inventory for a deteriorating item while the buyer has warehouse capacity constraint. International Journal of Production Economics. 2012;138(2):284-92. doi:10.1016/j.ijpe.2012.03.029.