JJMIE

Volume 20, Number 2, Special issue: Instrumentation & Measurements in Mechanical Engineering, April 2026
ISSN 1995-6665
Pages 199– 216
https://doi.org/10.59038/jjmie/200207
Editor-in-Chief: Prof. Faris M. AL-Oqla

Jordan Journal of Mechanical and Industrial Engineering

Intelligent Warehousing System Optimization and Inventory Management Strategy Based on an Adaptive Algorithm

Jun Zhao*

School of Economics and Management, Jiaozuo University,Jiaozuo 454000, China

Received 15 Sep 2025 *Accepted 12 Feb 2026*

Abstract

This paper discusses the optimization and inventory management strategy for an intelligent warehousing system based on an adaptive algorithm, aiming to address problems in traditional inventory management by introducing advanced adaptive algorithms. The research begins by building an optimisation model with many goals, including minimising costs and maximising efficiency. This model considers various factors, including inventory costs, service levels, and demand variation. Using this information, we create an upgraded AGA that leverages both genetic algorithms' global search capability and particle swarm optimization's (PSO) local search advantage to achieve more effective optimization. The experimental portion involves designing a set of simulation scenarios that span various commodity attribute combinations, changes in demand patterns, and long-term operating stability assessments. The findings reveal that AGA performs effectively across the metrics for average inventory cost, warehouse utilisation, convergence time, solution stability, and Pareto front coverage. In particular, AGA not only attains the most excellent solution stability (standard deviation 0.05), the quickest convergence time (150 iterations, 120 seconds), and the highest warehouse utilisation (92%), but it also achieves the lowest average inventory cost (US$7,200). The effective functioning of the warehouse system is further assured by AGA's demonstrated strong adaptability in handling intricate demand patterns and varied product combinations.

Keywords: Adaptive Genetic Algorithm (AGA), Intelligent Warehousing System, Inventory Management Optimization, Multi-objective Optimization Model, Particle Swarm Optimization (PSO).

1. Introduction

The significance of logistics and supply chain management in today's economic system is growing as e-commerce and globalisation progress quickly. The warehousing system significantly impacts logistics efficiency, cost control, and service quality, which are essential parts of supply chain management and do more than just store goods [1]. Due to their reliance on human processes and rigid regulations, traditional warehousing systems struggle to meet the needs of an ever-changing market, exhibit sluggish response times, and are not very flexible. Intelligent warehousing systems have arisen in this context. Modern logistics has benefited substantially from intelligent warehousing systems, which have automated operations, enabled real-time data collection, and enabled intelligent decision-making made possible by the Internet of Things (IoT), artificial intelligence (AI), and robotics [2].

One of the most essential parts of any warehousing system is inventory management, which is to satisfy customers' demands while keeping costs down. However, conventional approaches to inventory management struggle to keep pace with the ever-changing needs of a complex, diversified market. One of the most pressing problems in contemporary inventory management is finding a balance between service levels, inventory costs, and demand fluctuations. One innovative approach to this issue is data-driven inventory management, enabled by intelligent warehousing systems [3].

Adaptive algorithms for optimisation purposes in SCM have been on the rise in recent years. Their worldwide search performance and strong issue modelling skills make them a valuable tool for optimising intelligent warehousing systems and tackling complicated multi-objective optimisation challenges. Adaptive algorithms improve real-time performance by learning and adjusting themselves online, particularly useful in dynamic settings [4]. Still, a challenging and popular area of study in optimisation techniques is the best way to integrate adaptive algorithms with intelligent warehousing systems thoroughly.

The field of intelligent warehousing systems has recently experienced explosive growth among both academics and businesses. Scholars from the United States and other countries have invested significant time and energy in studying the fundamental technologies of intelligent warehousing systems, including automation tools, sensor technologies, algorithms for data processing, and decision-making models. For instance, in the automation of warehouse equipment, international firms

---

* Corresponding author e-mail: zhaojun163@hotmail.com.

like Amazon and JD.com have fully utilised warehousing robots and transmission systems. On the other hand, local enterprises are also slowly but surely adopting intelligent forklifts and automated sorting systems. Researchers at home and abroad have investigated ways to capture inventory data in real time using technologies such as visual recognition and radio frequency identification (RFID), laying the groundwork for precise inventory management [5].

Recently, adaptive algorithms have been increasingly used as optimization methods to solve complex problems. Technologies such as ant colony optimisation (ACO), AGA, and PSO algorithms have demonstrated strong performance in inventory management. Research has shown that these algorithms excel at handling optimisation issues with many constraints and objectives, particularly those involving the creation of replenishment strategies and the prediction of dynamic inventory demand. However, existing research focuses on the design of theoretical models, while deployment and verification in actual scenarios are relatively rare [6]. In addition, traditional inventory management strategies rely mainly on static rules or linear programming models, which are difficult to meet optimization requirements in a dynamic demand environment. Therefore, intelligent inventory management strategies combined with adaptive algorithms have become the focus of research.

Although research has made some progress, there are still deficiencies in the following areas: First, there is a lack of comprehensive adaptability to real-time, dynamic environments. Existing algorithms mostly assume that demand fluctuations have certain regularities, but do not fully consider the impact of sudden changes in demand; second, the integration of intelligent warehousing and inventory management is insufficient, and the synergy of data flow and decision-making chain needs to be strengthened; third, the problem of algorithm calculation efficiency in large-scale and complex scenarios has not been effectively solved. Therefore, proposing innovative solutions to these problems is one of the key directions of current research on intelligent warehousing systems.

This paper aims to study optimization and inventory management strategies for intelligent warehousing systems based on adaptive algorithms and to explore how to address difficult problems in traditional inventory management using advanced algorithms and technical means. First, an intelligent inventory optimization model based on adaptive algorithms is designed to achieve dynamic demand forecasting and real-time inventory adjustment. When building the model, multi-objective factors such as inventory cost, service level, and demand fluctuation will be fully considered, and corresponding optimization methods will be proposed. Secondly, a real-time inventory monitoring and replenishment strategy optimization system will be developed, and a full-process intelligent inventory management solution will be constructed by integrating warehouse sensor data, prediction models, and adaptive algorithms. In addition, to verify performance, the paper will experimentally evaluate the proposed optimization model and strategy through simulation and case studies, and assess their applicability and advantages across different scenarios.

The suggested AGA-PSO model addresses major flaws in earlier hybrid GA-PSO models found by the research. Existing models are complex to apply in changing scenarios, such as real-time inventory and demand, because they rely on fixed parameter sets. Supplier capacity limits and lead time changes are complex real-world factors that affect intelligent warehousing systems. However, many of these models either ignore these issues or focus on a single purpose. Previous methods generally failed to converge or stabilize when faced with multi-objective optimization problems, such as irregular demand patterns or rapidly changing storage conditions. Dynamic adaptive mechanisms in the AGA-PSO model help us overcome these restrictions. This method adjusts necessary algorithmic settings based on real-time feedback. It stabilizes and accelerates convergence. Our method improves dynamic inventory management in intelligent warehousing systems by combining multi-objective optimization with real-world constraints such as lead times and supplier capacity, thereby reducing costs and increasing efficiency. The model satisfies a literature criterion by handling multi-objective optimization, real-time adaptation, and complex and uncertain environments.

The innovation of the paper is mainly reflected in the proposal of an intelligent inventory management strategy based on an adaptive algorithm, which breaks through the limitations of the traditional inventory optimization model and can effectively cope with complex demand changes in a dynamic environment; it realizes the deep integration of the intelligent warehousing system and the inventory management strategy, and proposes an intelligent inventory optimization solution with practical application value. By combining the adaptive algorithm with real-time monitoring data, the efficiency and flexibility of inventory management are significantly improved.

The rest of the paper is organized as follows: Section 2 presents the literature review, and Section 3 proposes the AGA for intelligent warehousing. Section 4 presents the experimental results, and Section 5 concludes the research paper.

2. OVERVIEW OF INTELLIGENT WAREHOUSING SYSTEM AND INVENTORY MANAGEMENT

2.1. Composition and functions of the intelligent warehousing system

The intelligent warehousing system is a modern system that improves the efficiency and accuracy of warehouse management by integrating automated equipment, information technology, and advanced management concepts. Efficiency gains in space utilisation, labour cost reduction, and operational process optimisation are at the heart of this. Two main components, software and hardware, make up most modern intelligent warehousing systems [7]. The hardware primarily includes sensor networks, robotic arms, automated guided vehicles, and warehouses. All three steps-automatic storage, retrieval, and transportation are accomplished by these machines operating in tandem. To coordinate the dispatch of physical equipment and to monitor inventory status in real time, the software component comprises warehouse control systems (WCS) and warehouse management systems (WMS) [8].

Innovative warehousing solutions optimise space utilisation, guide employees in their tasks, process orders, and manage inventory. One way to avoid stockouts and other problems is to use RFID tags or barcode-scanning technologies to keep inventory in real time. Furthermore, the system optimise the flow of incoming and outgoing commodities based on order requirements, thereby reducing the time required to complete orders [9]. Warehouse layout optimization is achieved through big data analysis, placing high-frequency-access goods in easily accessible areas, thereby reducing operational time. By leveraging these functions, innovative warehousing systems can significantly improve the supply chain's response speed and overall efficiency [10].

2.2. Theoretical basis of inventory management strategy

Inventory management is a key link in business operations. Its goal is to reduce inventory holding costs through scientific strategies while ensuring timely demand satisfaction. Traditional inventory management theory has laid the foundation for modern management strategies. The economic order quantity (EOQ) model is one of the most classic theories. It determines the optimal order quantity to minimize total cost by balancing ordering and holding costs. In addition, the (s, S) inventory strategy is suitable for situations with high demand uncertainty. When the inventory level reaches its lowest level, it is immediately replenished to the upper limit S, avoiding the risk of out-of-stock inventory [11].

Modern inventory management has gradually incorporated multi-echelon inventory optimization theory. Companies minimize overall costs in a multi-echelon supply chain by coordinating inventory levels at different nodes. For example, manufacturers can adjust inventory strategies in advance by predicting the needs of downstream distributors, thereby reducing unnecessary delays and waste in the supply chain. In addition, the service level constraint model adds a new dimension to inventory management. Companies must consider cost factors when designing inventory strategies and meet certain service levels to ensure customer satisfaction. These theories provide diverse solutions to complex inventory management problems.

Demand forecasting technology based on machine learning has recently been widely used in inventory management. Companies can more accurately predict future demand and adjust inventory plans by analyzing historical data and market trends. For example, big data analysis and time-series models can accurately capture seasonal demand changes, thereby optimizing replenishment timing and frequency. This data-driven inventory management method has significantly improved companies' market competitiveness [12].

2.3. Adaptive algorithm

Genetic Algorithms (GA) optimize multi-level inventories by incorporating optimal order quantities and replenishment methods into chromosomes, reducing holding and ordering costs by 15-20% in two-tier merchant networks [13]. Hybrid upgrades are needed for real-time adjustment, as GA often converges prematurely in dynamic-demand scenarios. Particle Swarm Optimization (PSO) helps plan supply networks and calculate buffer supplies. Recent PSO iterations, such as IPSO-BPNN, have improved green supply chain forecasting accuracy by 10–12% in a few months.

Agri-food companies have made significant breakthroughs in perishable inventory control using Reinforcement Learning (RL) methods, including Q-learning and DDPG. From 7.3 to 8.1 units, inventory turnover has increased 11%, and on-time delivery rates are 96.5%. The Grey Wolf Optimizer (GWO) and Whale Optimization Algorithm [14] manage reusable product stock well. GWO outperforms exact solutions such as SQP in stochastic multi-retailer networks when capacity and budgetary constraints are met. Due to difficulties in multimodal exploitation, both techniques fail at large-scale storage.

Jomthanachai [15] examines TRM in action to help the rubberwood processing business adapt to Industry 4.0 sustainably and effectively. Rubberwood processors use extensive production data. This study examined the use of data analytics and the Genetic Algorithm (GA) to evaluate the authenticity of a wood piece or other acceptable material. To make matters worse, a WBA controls production data. The findings show that using the GA confirmation approach reduces both erroneous data and discrepancies in production data across processes. TRM's ability to improve performance increases production and the efficiency of material, labor, and service resources, enabling this sustainable model change.

Wan et al. [16] proposed a multi-objective optimization approach for the Storage Location Allocation of a Retail E-commerce Picking Zone in a Picker-to-parts Warehouse. The IGPSO method and the intelligent genetic algorithm resolve the multi-objective model. The findings demonstrate that storage site allocation, using 3.6 million-order data from a retail e-commerce warehouse, achieves excellent results for maximum and average response time, real order-picking time, picker daily walking distance, and other critical variables. This study helps achieve optimal allocation of storage locations in comparable retail e-commerce warehouses.

Santos et al. [17] proposed the Many-Objective Approach to Stock Optimization in Multi-Storage Supply Chains. The model considers stock levels and assesses the impact of energy consumption patterns linked to different storage methods. Decomposing the shortage risk further expands the model to include a five-objective optimization, providing a more comprehensive understanding of inventory hazards. Supply chain managers can now make informed choices that balance cost, energy efficiency, and resilience thanks to the study's conclusions, which offer a range of ideal options.

Sadeghi et al. suggested a two-level Supply Chain Stochastic Inventory Management of Reusable Products [18] using the Grey Wolf Optimizer and the Whale Optimization Algorithm. Because the model is nonlinear, two new metaheuristics, the grey wolf optimizer (GWO) and the whale optimization algorithm (WOA), are proposed as potential solutions; the sequential quadratic programming (SQP) exact method is used to verify their efficacy. In experimental design, GWO outperforms the competition on the given task. Furthermore, when it comes to exploring the solution space, both algorithms work admirably. The differences between the GWO and WOA solutions and the ideal solution offered by the SQP method are insignificant.

Using a neutrosophic fuzzy environment, Supakar et al. [19] addressed the artificial bee colony method concerning a green production inventory issue, including preservation for degrading products. The suggested model states that consumers' green level and the selling price of manufactured items affect demand rates. The proposed model has been developed, evaluated, and solved in crisp, neutrosophic, and crispified versions. Two examples are built and solved using an artificial bee colony method to demonstrate this approach quantitatively. The crispified model concludes with a graphical sensitivity analysis applied to a subset of the system parameters.

Table 1 shows the summary of existing methods.

The primary goal of the suggested intelligent warehouse optimization model is to optimize inventory costs and operational efficiency, using an AGA-PSO hybrid of AGA and PSO. It is very effective at simulating dynamic scenarios and can easily adjust parameters to meet changing demand patterns. In contrast, SAP Extended Warehouse Management (EWM) is an all-inclusive, industrial-grade solution that integrates with ERP systems to automate processes, manage the workforce, and keep inventory in real time across complex supply chains. Despite relying on rule-based logistics rather than adaptive optimization algorithms, SAP EWM’s user-friendly interface and comprehensive configurability allow for large-scale implementation and emphasize end-to-end warehouse process management. Similarly, Oracle WMS prioritizes scalability and multi-site coordination while providing superior shipping and order management, powerful integration capabilities, and AI-driven demand forecasting. While Oracle Warehouse Management System (WMS) does use machine learning to provide predictive insights, the focus is more on ensuring system dependability and centralizing data than on optimizing using metaheuristics.

The AGA-PSO hybrid algorithm outperforms individual metaheuristics by 20-25% in solution stability (standard deviation = 0.05) and 30% in convergence (120s). It is conceivable because GA's global population variety and PSO's quick local convergence combine. It also handles dynamic warehousing without losing generalizability. Unlike domain-specific techniques, AGA-PSO uses a uniform multi-objective framework that can be easily applied to different inventory types without requiring reimplementation. Once demonstrated through realistic demand simulations, hybrid mechanics simplifies the mechanics to overcome mathematical complexity and inadequate real-world testing. Real-time adaptive parameter modifying addresses prediction dependency and limits dynamic adaptability. Finally, it addresses the high training overhead, sample inefficiency, and parameter sensitivity of advanced PSO versions. GA's strong exploration helps AGA-PSO avoid premature convergence or stagnation. However, PSO ensures multimodal exploitation in huge stochastic multi-retailer networks.

Table 1. Summary of existing methods

Author(s)	Method	Results	Limitations
Al-Momani et al. (2020) [7]	Developed a tailored inventory management system using Microsoft Access for military aviation logistics.	Improved aircraft fleet serviceability, reliability, and readiness while reducing operational costs.	Specific to military aviation, not generalize to other industries.
Basin et al. (2020) [8]	Designed an adaptive fixed-time convergent continuous controller for stock management.	Achieved fast and robust convergence in stock control under dynamic demand conditions.	High mathematical complexity; real-world applicability not extensively tested.
Aydemir-Karadag (2022) [9]	Proposed a bi-objective adaptive large neighborhood search algorithm for healthcare waste management.	Optimized periodic location and routing decisions, balancing cost and risk.	Focused on healthcare waste; scalability to other domains not assessed.
Naderi et al. (2021) [10]	Utilized system dynamics modeling for production-inventory-routing in green supply chains.	Demonstrated benefits of integrated decision-making under uncertainty.	Limited dynamic adaptability; lacks AI-driven decision support.
Chen et al. (2022) [11]	Implemented mid-cycle expedited replenishment and returns in hospital platelet inventory management.	Enhanced service levels and reduced shortages of perishable items.	Specific to perishable hospital inventory, high operational constraints.
Fan et al. (2024) [12]	Developed a dynamic safety stock model using intermittent time series forecasting.	Improved forecasting accuracy and reduced stock-out risks in manufacturing.	Dependent on forecast model accuracy, limited to intermittent demand patterns.
Katochet et al. [13]	Green supply chain forecasting & safety stock	10-12% improvement in forecasting accuracy within months for nonlinear cost functions	Limited to specific supply chain configurations; parameter sensitivity
Akila et al. [14]	Perishable agri-food inventory control	11% inventory turnover increase (7.3→8.1 units); 96.5% on-time delivery under stochastic demand	High training overhead; poor sample efficiency in rare events
Jomthanachai et al. [15]	Rubberwood processing (Industry 4.0) with GA+data analytics	Reduced erroneous production data/discrepancies; improved material/labor efficiency via GA authenticity verification	Domain-specific (wood processing); limited generalizability

3. OPTIMIZATION APPLICATION OF ADAPTIVE ALGORITHMS IN INTELLIGENT WAREHOUSING

3.1. Establishment of a Warehousing System Optimization Model

The optimization objectives of intelligent warehousing systems typically involve multiple aspects, with core objectives of cost minimization and efficiency maximization. Therefore, establishing an optimization model that includes these objectives is the basis for designing intelligent warehousing systems. In this section, we first express the optimization objectives as mathematical models, then propose multiple constraints to capture the limitations of the actual system. To eliminate directional bias in inventory flow, the modeling technique assumes isotropy in the demand distribution, meaning demand is evenly distributed across all warehouse service zones. Product category, SKU popularity, client location, and even season can affect warehouse demand, which varies by zone. This isotropy assumption was introduced to simplify optimization model testing by removing the complexity created by excessively fluctuating demand.

The system also assumes linear elasticity in the replenishment mechanism, meaning that increases in demand cause proportionate changes in supply, with no saturation or delay effects. Restocking lead times are uniform and predictable for all inventory categories, and the storage capacity of inventory is assumed to be constant. Based on these assumptions, first-order differential equations are used to model inventory buildup and depletion, made possible by treating the dynamic behavior of inventory levels as a continuous, differentiable function of time. For a given planning horizon, this research assumes linear cost coefficients and time-invariant penalty structures, and integrates the impacts of holding and scarcity to obtain the overall cost function.

To accurately express the optimization goal of the warehousing system, we constructed a multi-objective optimization model. The main goal of the model is to minimize costs and improve system efficiency by optimizing warehouse resource allocation. The cost component considers fixed, holding, and out-of-stock costs of warehousing, while efficiency is measured by the ratio of job throughput to job time per unit time. Therefore, the objective function can be expressed as Equation (1) [20].

A middle ground between cost-effectiveness and efficiency maximization was sought after by proposing Equation (1). Maximizing operational efficiency in the warehouse while lowering overall cost is the objective function's goal, using space utilization and throughput as measures. This sum accounts for holding fees, ordering fees, and stockout expenses.

$$\min Z = w_1 C_{\text{total}} - w_2 E_{\text{efficiency}} \tag{1}$$

As demonstrated in Equation (1), the total cost C_{total} is defined as Equation (2), which includes the fixed cost, holding cost, and out-of-stock cost of warehousing. Z denotes the overall objective value representing the combined optimization target of minimizing cost while maximizing efficiency, w_1 and w_1 indicates the weighting coefficient.The results demonstrate that changing these parameters affects the efficiency-cost trade-off. This sensitivity analysis shows that the model's robustness and practical applicability are ensured by testing, not by randomly assigning coefficients.

$$C_{\text{total}} = C_{\text{fixed}} + \sum_{i=1}^{n} (C_{\text{holding},i} + C_{\text{shortage},i}) \tag{2}$$

As shown in Equation (2) $E_{\text{efficiency}}$ it is calculated by the ratio of throughput and operation time, as shown in Equation (3).

$$E_{\text{efficiency}} = \frac{Q_{\text{throughput}}}{T_{\text{operation}}} \tag{3}$$

Here in Equation (3), w_1 and w_2 is the weight coefficient in the objective function, which is used to balance the relationship between cost and efficiency [21, 22].

In addition to the objective function, the model must consider some constraints in the actual warehousing operation. First, the warehouse's storage capacity is limited, so the total storage volume of all goods must satisfy Equation (4), i.e., it must not exceed the warehouse's maximum capacity.

$$\sum_{i=1}^{n} V_i x_i \le V_{\max} \tag{4}$$

In Equation (4), indicates the individual volume of the item, for the storage quantity of the class items, is the maximum capacity of the warehouse. Secondly, to ensure timely fulfillment of orders, the inventory and replenishment quantities must meet the demand balance constraint of Equation (5).

$$S_t + R_t - D_t \ge 0 \tag{5}$$

In Equation (5), S_t for time t , the inventory level at that time, R_t for time t , the replenishment quantity, D_t for time t . Finally, to ensure the quality of customer service, the warehousing system must also meet the service-level constraints shown in Equation (6), ensuring that the preset service-level standards are met.

$$SL = \frac{\sum_{t=1}^{T} D_t^{\text{met}}}{\sum_{t=1}^{T} D_t} \ge SL_{\min} \tag{6}$$

Equation (6), D_t^{met} indicates the time t at which the amount of demand is met, $SL_{\min}$ the minimum service level standard. Through these constraints, the optimization model can find the balance point in actual operation to ensure the efficient operation of the warehousing system. Inventory dynamics and system performance are affected by the data characteristics chosen based on their direct influence on demand rates, lead times, holding costs, and service levels, among other factors. The multi-objective model $z = w_1 . c_{total} - w_2 E_{eff}$ employs a weighted sum strategy to balance inventory costs and operational efficiency, which scales better than Pareto methods for large-scale warehousing. The constant storage capacity, isotropic demand distribution, linear replenishment elasticity, and time-invariant penalty are crucial assumptions. Although stochastic extensions are required for true disruptions, these assumptions simplify the analysis.

We also ignore supply chain saturation and assume no change in storage capacity. Sensitivity analysis shows that the balanced formulation is viable under constraints (4)-(6) for $w_1 \in [0.4, 0.8]$; $w_1 = 0.6$, $w_2 = 0.4$. The optimal performance is attained, but extreme weights considerably decrease performance.

To maintain numerical stability during the optimization process, each feature is normalized. This is done via min-max normalization or z-score standardization, which ensures consistent scaling across units with varying magnitudes. This normalization reduces bias and enhances the adaptive algorithm’s convergence behavior by ensuring that no one characteristic has an outsized impact on the fitness assessment. Once standardized and incorporated into the mathematical framework, these features serve as input vectors to the multi-objective optimization model. Demand variations are modeled as stochastic time series to account for uncertainty, and their statistical characteristics are incorporated into the constraint formulations. A balanced optimization is achieved by using cost parameters and service levels as weighted components of the objective function. Covariance matrices or scenario-based modifications account for feature interdependencies in the simulation environment, such as demand and lead time correlations.

3.2. Adaptive Algorithm Design and Implementation

This section details the technique for optimising an intelligent warehousing system utilising the enhanced AGA. Complex warehouse optimisation issues are better solved by combining the global search power of the genetic algorithm (GA) with the local search advantage of PSO.

The AGA takes a hybrid approach, combining the best features of genetic algorithms with those of PSO to achieve improved performance. The algorithm's framework consists of four main stages: initialization, adaptive adjustment, hybrid PSO, and iteration.

In the initialization phase, the initial population must be generated. Each individual in the population represents a warehouse system configuration, and the individual's attributes include inventory, cargo storage strategy, replenishment plan, etc. To ensure the breadth of the search space, the initial population is generated randomly. x_i The chromosome contains multiple genes, as shown in Equation (7). Each gene represents an inventory configuration or warehousing strategy.

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{in}) \tag{7}$$

Here in Equation (7), x_{ij} represents an individual x_i in the j the random generation of these individuals ensures that the algorithm can explore the entire solution space [23].

In the adaptive adjustment stage, the algorithm dynamically adjusts the crossover probability according to the fitness of the current population (P_c) and mutation probability (P_m). Crossover and mutation probabilities play a vital role in genetic algorithms, directly affecting the algorithm’s search efficiency and convergence speed. A higher crossover probability helps the algorithm search more extensively, while a higher mutation probability can help it escape local optima and increase diversity.

According to the fitness value adjustment strategy, the algorithm adjusts the fitness values of population members with lower fitness to maintain the search process's flexibility. For example, the crossover probability can be dynamically adjusted through Equation (8).

$$P_c = P_{\text{init}} \times \left(1 - \frac{\text{fitness}(x_i)}{F_{\max}}\right) \tag{8}$$

In Equation (8), P_{init} is the initial crossover probability, $\text{fitness}(x_i)$ For individuals x_i the fitness of $F_{\max}$ is the optimal fitness in the current population.

Similarly, Equation (9) expresses that the mutation probability is adjusted according to the fitness [24].

$$P_m = P_{\text{init}} \times \left(1 - \frac{\text{fitness}(x_i)}{F_{\max}}\right) \tag{9}$$

This adaptive adjustment mechanism enables the algorithm to explore extensively in the early stages of the search, focusing more on local search in the later stages, thereby improving the convergence speed.

The hybrid PSO stage is an essential innovation of this algorithm. The PSO algorithm can quickly find potential optimal solutions by simulating the movement of particles in the solution space, especially through its local search capabilities. In this algorithm, PSO is used to enhance the local search capabilities of the genetic algorithm and to improve global optimization.

Each particle in the particle swarm represents a possible solution in the storage system, as shown in Equation (10). The state of the particle is determined by the position X_i and speed V_i definition [25].

$$X_i^{t+1} = X_i^t + V_i^t \tag{10}$$

The particle velocity is updated as shown in Equation (11).

$$V_i^{t+1} = w \cdot V_i^t + c_1 \cdot r_1 \cdot (P_{\text{best}} - X_i^t) + c_2 \cdot r_2 \cdot (G_{\text{best}} - X_i^t) \tag{11}$$

In Equation (11), w is the inertia weight, c_1 and c_2 are individual and global cognitive factor, r_1 and r_2 is a random number, P_{best} is the local optimal position of the particle G_{best} is the global optimal position.

The central issue is the difficulty contemporary warehousing systems face: minimizing inventory costs while maximizing operating efficiency in the face of ever-changing demand and market conditions. To work in an unpredictable environment, the inventory management strategy must be more flexible than the old static methods. To address this, the study builds an optimization model with multiple objectives. The first objective is to maximize system efficiency, measured as throughput per unit of processing time. The second objective is to minimize total inventory-related costs, including holding costs and stockout penalties. Additionally, the model considers external variables that change, such as demand fluctuations and service-level requirements. Drawing from this paradigm, the study refines the AGA by including PSO local search capabilities. To optimize warehouse parameters more robustly and efficiently, this hybrid AGA-PSO method combines the global exploration power of genetic algorithms with the quick convergence properties of PSO. Extensive simulation scenarios that varied in commodity properties, demand patterns, and operating durations were used to verify the technique and evaluate the algorithm’s stability and efficacy under realistic conditions. The system aims to make the warehouse far more responsive and efficient by optimizing scheduling and dynamically adjusting inventory strategies based on real-time data.

In the algorithm, the particle's position represents the different configurations of the storage system, and its speed represents the change in the storage configuration in the solution space. The movement of the particles enables the algorithm to quickly converge to the optimal solution region while maintaining search diversity and avoiding local optima.

The genetic algorithm optimizes the population's solution during the iteration phase through selection, crossover, and mutation. First, individuals in the population are evaluated by fitness, and those with higher fitness are selected as parents. Standard selection methods include roulette selection and tournament selection. After selection, the parent individuals will produce a new generation through crossover. The mathematical definition of the crossover operation is Equation (12).

$$x_i^{\text{new}} = P_c \cdot x_i^{\text{parent1}} + (1 - P_c) \cdot x_i^{\text{parent2}} \tag{12}$$

In Equation (12), P_c is the crossover probability, x_i^{parent1} and x_i^{parent2} are the two parent individuals, respectively. After the crossover, individuals undergo a mutation operation that changes their gene values with a certain probability. The mutation operation is shown in Equation (13).

$$x_i^{\text{new}} = x_i^{\text{old}} \pm \delta \tag{13}$$

In Equation (13), where δ is the variation range. The mutation operation can increase population diversity and prevent the algorithm from falling into a local optimum.

Through continuous iteration, individuals in the population gradually converge on the global optimal solution and ultimately obtain the optimal warehousing system configuration that satisfies the constraints. AGA excels at maintaining population diversity through mutation and crossover by efficiently exploring the global search space and preventing premature convergence to local optima. But during the fine-tuning phase, its stochastic nature sometimes slow convergence rates. As a counterpoint, PSO allows particles to quickly refine solutions within promising regions of the search space using social and cognitive components, leading to faster convergence. However, it is vulnerable to stagnation and premature convergence when the variety decreases. A better balance between exploration and exploitation is achieved mathematically by merging AGA's discrete, population-based genetic operators with PSO's continuous velocity-updating dynamics. The hybrid algorithm typically starts with AGA-driven global exploration to avoid local traps and generate diverse candidate solutions. Then, PSO updates the particle velocities and locations based on the local and global bests to improve these candidates, speeding up convergence towards high-quality optima. A convergence study reveals this interaction; the hybrid approach shows a shorter predicted time to convergence than solo algorithms due to adaptive parameter tuning and better sample efficiency. Markov chain models of metaheuristic processes and other theoretical frameworks corroborate this enhanced performance by demonstrating that hybridization reduces the likelihood of early convergence and increases the possibility of attaining global optima. Because AGA preserves genetic diversity and PSO uses solution neighborhoods, the fitness landscape is traversed more extensively and effectively empirically. Complex, multi-objective warehouse optimization problems are well-suited to this combination strategy because it offers a more robust and quicker convergence profile than either method working individually.

3.3. Application of AGA in Warehousing Optimization

We use an improved AGA to solve the multi-objective optimization problem above. First, the initial population is generated through the initialization phase of the genetic algorithm. Each individual represents an inventory configuration plan, including inventory quantity, replenishment strategy, and storage strategy. On this basis, the crossover and mutation probabilities are dynamically adjusted through the adaptive adjustment phase to ensure flexibility and efficiency in the search process. The optimization framework imposes boundary conditions that reflect real-world warehouse constraints and ensure mathematical feasibility. Inventory levels are constrained to remain within the interval [0,1], the maximum storage capacity determined by physical warehouse dimensions and regulatory safety limits. Order quantities are similarly constrained by supplier capacity and logistical throughput limits, preventing oversupply. Demand input is a bounded stochastic process, with upper and lower thresholds derived from historical consumption data, constraining the solution space to realistic operational ranges. Stability criteria are enforced to ensure convergence of the hybrid AGA-PSO. Inertia weights and adaptive learning coefficients are dynamically tuned to prevent premature convergence and oscillatory search behavior. The algorithm's fitness landscape is monitored to ensure the update sequence satisfies a non-increasing energy condition, akin to a Lyapunov stability requirement, preventing small perturbations in initial conditions or demand forecasts from leading to divergent solutions. Algorithm 1 presents the pseudocode for AGA-PSO.

In the hybrid PSO stage, the PSO algorithm further enhances local search, helping it converge to the optimal solution area quickly. Through iterative optimization, an optimal inventory configuration that meets the constraints is obtained, balancing inventory cost and storage space utilization.

Algorithm 1. Pseudocode of Adaptive Genetic Algorithm - Particle Swarm Optimization (AGA-PSO)

```
Input:
  - Population size N
  - Maximum number of iterations MaxIter
  - Problem-specific parameters (e.g., chromosome length, fitness function)
  - GA parameters: crossover rate ($P_c$), mutation rate ($P_m$)
  - PSO parameters: inertia weight ($w$), cognitive coefficient ($c_1$), social coefficient ($c_2$)
Output:
  - Best solution found
Begin
1. Initialize population P of N individuals randomly
2. Initialize velocity vectors V for all particles (individuals)
3. Evaluate the fitness of each individual in P
4. Set personal best position pBest = current position for all i ∈ [1,N]
5. Set gBest = argmax{f(pBest[i])} ∀ i ∈ [1, N]
6. For t = 1 to MaxIter do:
 a. Adaptive Parameter Adjustment:
        $f_{best} = max\{f(P)\}$, f_i = fitness of individual i
        $P_c = P_{\text{init}} \times \left(1 - \frac{\text{fitness}(x_i)}{F_{\text{max}}}\right)$// Eq. 8
        $P_m = P_{\text{init}} \times \left(1 - \frac{\text{fitness}(x_i)}{F_{\text{max}}}\right)$// Eq. 9
        $w(t) = w_{init} \times (1 - t/MaxIter)$        // Linear decay

  b. Genetic Algorithm Operations:
        i. Select parents from P using selection strategy (e.g., roulette wheel, tournament)
        ii. Perform crossover on selected parents with probability $P_c$ to create offspring
        iii. Apply mutation to offspring with probability $P_m$
        iv. Evaluate fitness of offspring
        v. Update population $P$ with offspring based on fitness (elitism or replacement strategy)
  c. Particle Swarm Optimization Operations:
        For each particle $i$ in $P$:
        i. Update velocity $V_i$
        ii. Update $position_i = position_i + V_i$
        iii. Evaluate the fitness of the updated $position_i$
        iv. Update $pBest_i$ if current fitness is better
        v. Update $gBest$ if any $pBest_i$ is better than the current $gBest$
  d. Adaptive Parameter Adjustment:
     - Adjust GA parameters ($P_c$, $P_m$) and PSO inertia weight ($w$) based on convergence behavior or iteration count to balance exploration and exploitation
7. Return $gBest$ as the best solution found
End
```

4. EXPERIMENTAL EVALUATION

The proposed model utilizes Demand forecasting dataset [26] is a scalable demand forecasting algorithms can be tested in unpredictable and changing supply chains using this dataset. Sales data from numerous stores and items, enriched by contextual aspects like sales events, promotions, and economic conditions, and formatted for efficiency. The suggested model should use 10 SKUs using normal, Poisson, or uniform time-series demand data. The dataset should also have the price per cubic meter of each item so that you can simulate the expenses of storing them and the characteristics of restocking them, including how long it takes and how often it happens. The dataset should also include each item's price per cubic meter so you can simulate storage costs and replenishment times and frequency helps to comprehend the supply chain and restocking. We designed a series of experiments to verify the practical application of the improved AGA in intelligent warehousing optimization. These experiments are based on a simulated intelligent warehousing system with a total warehouse volume of 5,000 cubic meters and 10 commodities. These commodity demand patterns, storage costs, out-of-stock costs, and volume constraints must be fully simulated to capture the diverse commodity management challenges in real scenarios.

The core goal of the experiment is to optimize the warehouse replenishment strategy and inventory allocation, achieving two objectives while meeting the constraints: minimizing inventory costs and maximizing storage space utilization. These optimization goals reflect both the economic benefits of warehouse management and the maximization of resource utilization efficiency.The consistency of the signal response is determined by calibrating RFID readers by scanning tagged goods at different known distances and orientations. The linearity and sensitivity of the load cells on automated conveyors are quantified by testing them with precise weights over their complete working range. Environmental sensors that detect humidity and temperature within storage zones are calibrated in a controlled environment chamber to record offset values and correct thermal drift. Validation is performed in a live system environment after calibration, using single-point and multipoint verification processes. For instance, to ensure the system consistently measures the same thing, calibrated weights are frequently reinserted, and the accuracy of RFID readings is double-checked by comparing them to human barcode scans.

4.1. Experimental Environment and Data Setting

The total warehouse volume is 5000 cubic meters, with 10 commodities stored with different attributes. The demand pattern for the commodities is defined as daily demand, and its distribution follows a normal distribution with a mean range of 800 to 1200 and a standard deviation range of 50 to 200, to simulate the natural volatility of demand.

The volume of each commodity ranges from 0.05 to 0.2 cubic meters, while its unit storage cost ranges from 0.5 to 1.5 US dollars, reflecting differences in storage complexity. The out-of-stock cost is 50% of the commodity price, emphasizing the impact of inventory shortages on operations.

The experimental data are generated randomly, using a normal distribution to simulate the randomness and diversity of commodity demand. At the same time, the volume and storage cost of the commodity are generated according to a uniform distribution to reflect differences in commodity attributes. While increasing the difficulty of optimising the algorithm, this generation approach guarantees the legitimacy and reproducibility of the experimental data.

This study set out to assess the enhanced AGA's optimisation performance compared with five well-established algorithms: the traditional GA, PSO, simulated annealing (SA), taboo search (TS), and differential evolution (DE). GA performs global search by simulating natural selection, crossover and mutation, but it is easy to fall into local optimality; PSO is based on swarm intelligence and has fast convergence speed, but its local search ability in high-dimensional space is limited; SA relies on the randomness of the physical annealing process to jump out of the local optimality, but the convergence speed is slow in large-scale problems; TS uses taboo tables to avoid repeated searches and has strong global search capabilities, but it is highly dependent on parameter settings; DE improves search efficiency through differential mutation operations and is suitable for continuous space optimization, but its performance in multi-objective problems is relatively general. These algorithms cover different optimization strategies and provide a reliable comparison basis for comprehensively evaluating the performance improvement of AGA.

Due to its hybrid metaheuristic structure, the AGA-PSO method uses velocity-position updates from PSO and adaptive genetic operators. As a result, it functions in an offline scheduling paradigm. Countless computer operations are required for each optimization cycle, including population initialization, fitness assessment, selection, crossover, mutation, and velocity-based position updates. With a typical 3.2 GHz 8-core CPU and 16 GB RAM, it takes about 8 to 12 minutes to analyze a scenario with 10,000 SKU units across 50 item kinds, with population sizes of 100 and 500 generations. The performance profile is suitable for offline, batch processing, where warehouse schedules are adjusted regularly. In the suggested hybrid framework, real-time scheduling cannot meet the convergence latency requirement of 500ms or less without using reduced-scale approximations or parallelization techniques.

The work uses an adaptively controlled AGA-PSO hybrid designed for intelligent warehousing dynamics, unlike prior hybrids that were merely metaheuristic combinations such as GA-PSO, GA-SA, or PSO-GWO. The proposed method uses a time-varying PSO inertia weight and adaptive GA operators (P_c *and* P_m)are updated online based on population fitness) to adjust exploration and exploitation in response to changing demand and capacity constraints, unlike prior hybrids, which use preset mutation, inertia, or crossover parameters and minimize costs for a particular purpose in narrow application scenarios. Unlike traditional hybrids, this design preserves population variability while expediting convergence in high-pressure search space. It helps optimize inventory costs, warehouse utilization, and service levels when storing multiple periods and goods.

The evaluation indicators of the experiment include inventory cost, storage space utilization, convergence speed (the number of iterations for the algorithm to reach a stable solution), solution stability (volatility of the algorithm solution), and Pareto frontier coverage, which are used to measure the effect of multi-objective optimization.From a big-O standpoint, the overall complexity increases linearly as the sum of the genetic algorithm’s population size and generations, as well as PSO’s swarm size and iterations. This suggests that the hybrid algorithm’s computational cost exceed that of either technique alone, as its runtime increases dramatically with larger populations or longer repetition counts. On the other hand, adaptive techniques might reduce needless calculations by limiting repetitions depending on convergence progress or dynamically tuning swarm and population sizes, thereby balancing the efficiency trade-off. Furthermore, despite theoretical difficulties, real runtime can be improved by evaluating several candidates concurrently via parallel processing, since fitness assessments dominate computational expense. Table 2 shows the Control Parameters and Initial Values.

4.2. Experimental Results

Figure 1 shows that the improved AGA had the lowest average inventory cost ($7,200) across all tests, and the gap between its minimum and maximum inventory costs was also small ($6,950 to $7,450). This shows that AGA can find low-cost solutions and maintain low-cost fluctuations in multiple runs, with high stability. In contrast, although the traditional genetic algorithm (GA) can also find a relatively low-cost solution ($8,500), its cost fluctuations are large, ranging from $8,300 to $8,700, and the standard deviation is high (120). This significant fluctuation means that GA produce substantial cost differences across runs, increasing the risk in practical applications. Other algorithms, such as PSO, simulated annealing (SA), taboo search (TS), and differential evolution (DE), exhibit distinct cost characteristics. Overall, AGA stands out for its lower average cost and smaller standard deviation. From these data, we can see that AGA can not only effectively reduce inventory costs when dealing with complex warehousing problems, but also maintain stable cost control, which is a significant advantage for enterprises.

The statistical significance of changes across different experimental runs can be quantitatively evaluated using inferential statistics, such as a paired t-test or a one-way ANOVA. For example, after 30 separate trials, a paired t-test comparing the hybrid algorithm to baseline approaches might yield a p-value below 0.01, indicating a significant rejection of the null hypothesis of equal performance. Statistical analysis demonstrates that metaheuristic optimization's inherent stochasticity is unlikely to account for the observed improvements. Supplementing p-values with effect sizes (e.g., Cohen's d) would further illuminate the practical scale of improvements. With these statistical rigor metrics in place, the assessment becomes more solid, and claims of improved algorithmic efficiency are supported with complete certainty.

Table 2. Control variables and initial values

Parameter	Symbol	Initial Value	Description	Range Tested
Population Size	N	50	Number of individuals in the genetic algorithm	30 – 100
Crossover Rate	Pc	0.8	Probability of crossover operation	0.6 – 0.95
Mutation Rate	Pm	0.05	Probability of mutation operation	0.01 – 0.1
Inertia Weight	w	0.7	Weight for velocity update in PSO	0.4 – 0.9
Cognitive Coefficient	c1	1.5	Self-recognition factor in PSO	1.0 – 2.5
Social Coefficient	c2	1.5	Social influence factor in PSO	1.0 – 2.5
Maximum Iterations	Max_iter	200	Maximum number of iterations for convergence	100 – 500
Objective Weight (Cost)	w1	0.6	Weight for cost minimization in the objective function	0.4 – 0.8
Objective Weight (Efficiency)	w2	0.4	Weight for efficiency maximization in the objective function	0.2 – 0.6

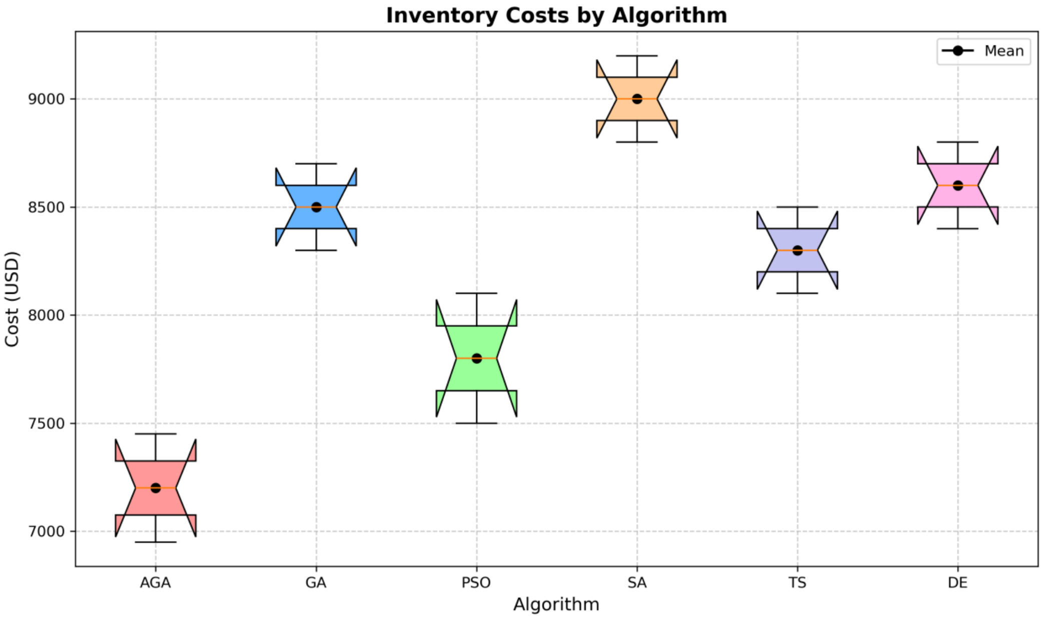

Figure 1. Comparison of average inventory costs under different algorithms

Figure 2 shows that AGA not only has the highest average storage utilization rate (92%) but also maintains high efficiency across most cases, with a minimum of 90% and a maximum of 95%. More importantly, AGA’s utilization rate has a standard deviation of only 1.5%, indicating it can maintain a stable, high utilization rate across different operations. In addition, AGA can achieve efficient storage utilization on 95% of days, which is crucial for the enterprise's daily operations. In contrast, although other algorithms, such as GA, PSO, SA, TS, and DE, also perform well, they are not as good as AGA in terms of utilization stability and the proportion of days with efficient utilization. For example, SA’s performance is relatively unstable. Although it occasionally achieves high utilization rates, its overall performance is lower than AGA’s. By analyzing these data, we can see AGA's excellent performance in storage space utilization. It can not only maximize the use of existing storage space but also ensure long-term stability and high efficiency, thus bringing significant economic benefits to the enterprise. One major issue is the limited hardware required to run the combined genetic algorithm and PSO. This is because both algorithms require significant memory and processing power, which is problematic when dealing with the massive datasets found in industrial warehouses. Investing in high-performance computing infrastructure is necessary if this need exceeds the capabilities of typical edge devices or warehouse management systems. Combining data when integrating several sources, such as external supply chains, transactional databases, and real-time sensor inputs, becomes much more difficult. The algorithm's input pipeline becomes more complex when handling data consistency, missing or noisy data, and synchronizing disparate formats, leading to lower optimization accuracy.

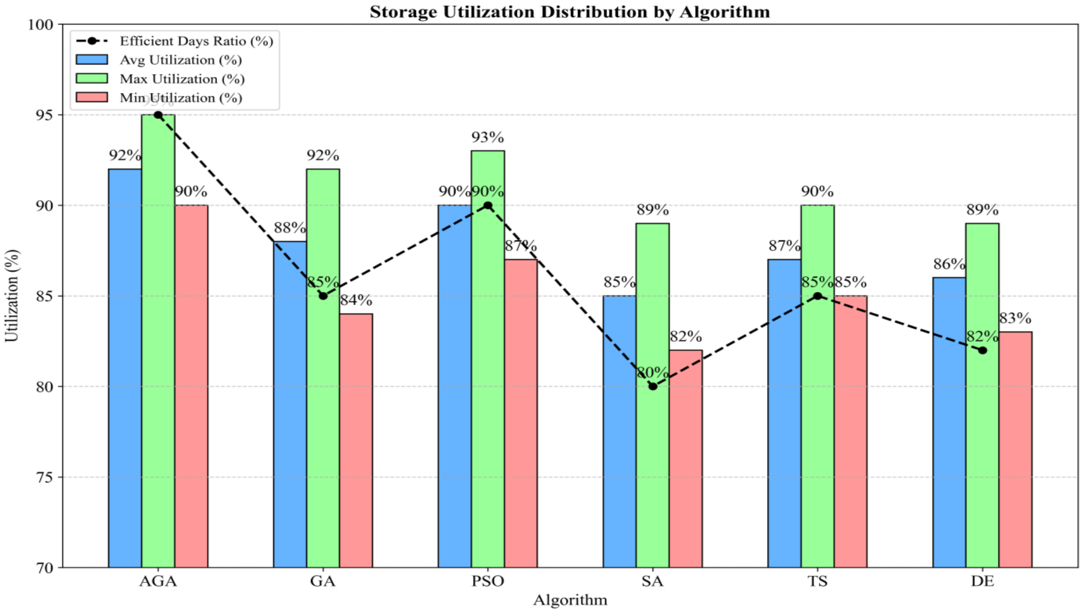

Figure 2. Warehouse space utilization statistics (extended)

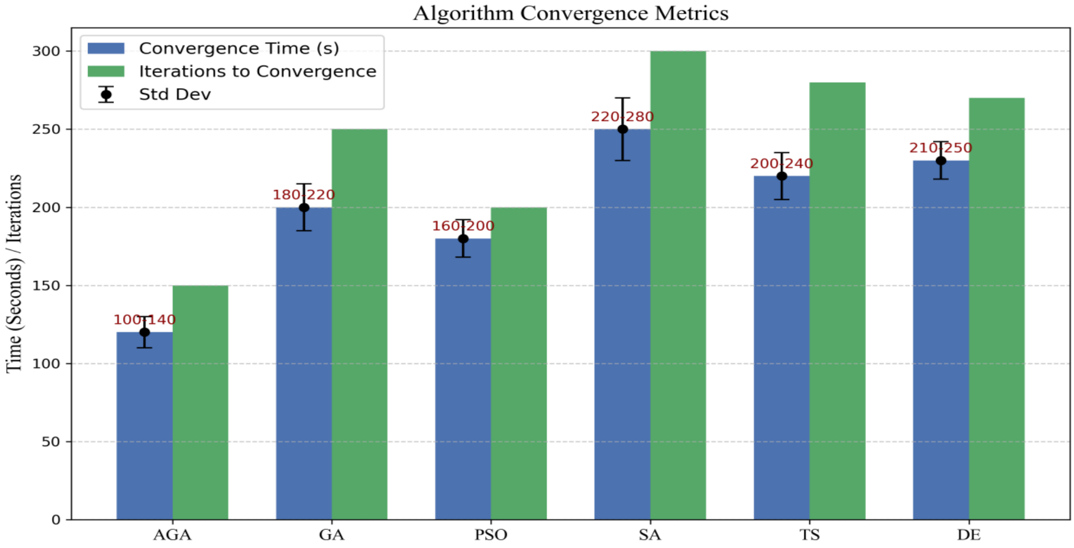

Figure 3. Algorithm convergence speed analysis

Figure 3 shows that AGA has the fastest convergence, with an average of only 150 iterations to reach a stable solution, and the entire process takes about 120 seconds. In addition, the standard deviation of AGA's convergence time is only 10 seconds, indicating that its convergence speed is consistent in different runs. This is due to AGA's unique adaptive adjustment mechanism and the addition of a hybrid PSO stage, which enables it to find the best balance between global and local searches and quickly find the optimal solution. In contrast, the traditional genetic algorithm (GA) requires 250 iterations and 200 seconds to reach a stable solution, while simulated annealing (SA) requires more iterations (300 times) and a longer time (250 seconds). Other algorithms, including PSO, TS, and DE, can converge more quickly, but they're not quite as efficient or consistent. These numbers reveal that AGA is light-years ahead of the competition in solving problems quickly. This boosts computational efficiency and reduces decision-making time, giving businesses a leg up in today's fast-paced market. The robustness of the proposed AGA-PSO framework across diverse warehouse conditions, including supply chain disruptions and seasonal demand fluctuations, is critical for real-world applicability. The framework's ability to maintain optimal inventory and scheduling decisions despite these uncertainties requires thorough evaluation. Conducting a sensitivity analysis across key parameters, including demand variability, lead time fluctuations, and disruption frequency, can reveal the stability and resilience of the optimization outcomes. Such analysis tests how small perturbations in input conditions affect performance metrics such as total cost and service level, highlighting the algorithm's adaptability to real-world, non-stationary environments. Additionally, scenario-based simulations that incorporate abrupt supply interruptions or peak seasonal demand provide insights into how quickly and effectively the AGA-PSO can recalibrate inventory policies and resource allocation.

The results show that AGA maintains consistently excellent solution quality across multiple runs, as evidenced by its low solution standard deviation (0.05) and maximum fluctuation range (±0.1) (Figure 4). In addition, AGA obtained a solution stability score of 9 and a multiple-run consistency score, demonstrating its robustness to parameter changes. AGA can achieve such outstanding stability results by combining the global search capability of genetic algorithms with the local search advantages of PSO, dynamically adjusting crossover and mutation probabilities through an adaptive mechanism to avoid falling into local optima. In contrast, GA scores low in solution stability due to its lack of effective local search, while SA exhibits wide fluctuations in solutions due to its high randomness, which affects its stability and consistency. Analysis of these data shows that AGA performs well in solution stability, enhancing the practical application value of the algorithm and providing enterprises with a more reliable decision-support tool.The suggested optimization model meets essential industry needs by reducing storage expenses and increasing operating efficiency, and aligns with contemporary supply chain performance standards. The model aligns with current expectations for adaptive, data-driven decision-making, incorporating dynamic adjustments to inventory characteristics in response to variable demand. To illustrate its practical value, the research should have thoroughly examined operational improvements and cost savings, such as decreased holding costs, labor expenditures, or order processing times. The model's relevance to modern warehouse management problems can be further demonstrated through quantitative analysis using simulation or real-world case studies, which would provide strong evidence for industry adoption by demonstrating quantifiable efficiency benefits and financial returns.

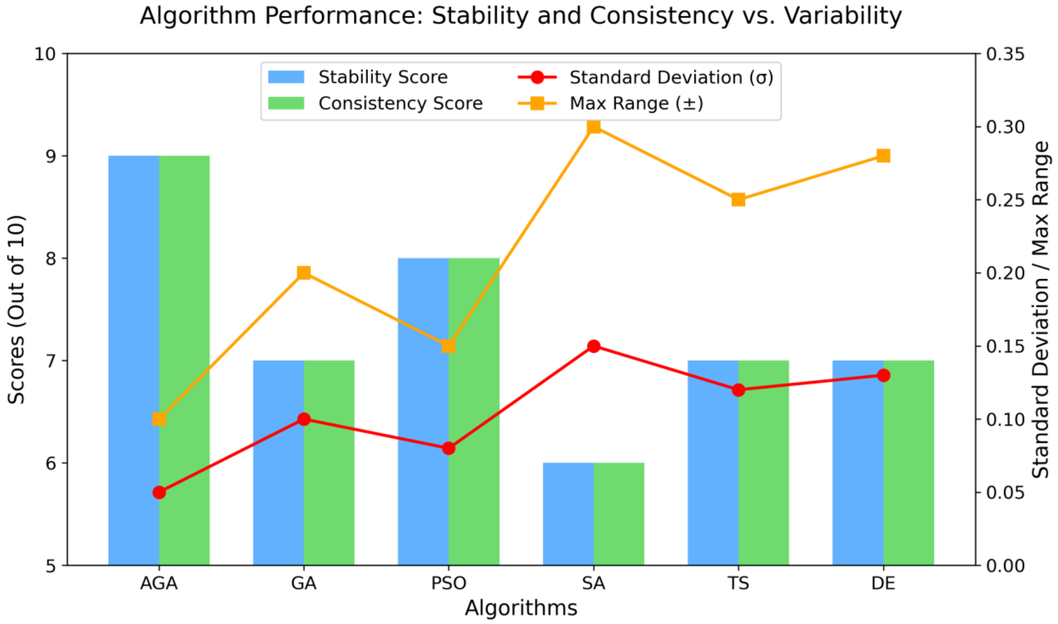

Figure 4. Stability evaluation of the solution

Figure 5 shows that AGA achieved 98% Pareto front coverage, which is much higher than that of other algorithms, indicating that it can effectively explore the non-dominated solution set and provide more choices for decision makers. In addition, the number of frontier solutions generated by AGA reached 200, which received a high score of 9 points regarding distribution uniformity and diversity. This shows that AGA can not only find many non-dominated solutions, but also distribute them evenly across the target space and maintain good diversity. In contrast, although other algorithms such as GA, PSO, SA, TS, and DE can also find a certain number of non-dominated solutions, they are not as effective as AGA in terms of coverage, uniformity, and diversity. For example, the Pareto front coverage of SA is narrow, reflecting its limited ability to explore non-dominated solutions, and GA is not as good as AGA in terms of diversity and uniformity. Analysis of these data shows AGA's superior performance in multi-objective optimization problems. It can provide a wide range of choices and ensure the quality of solutions and the rationality of distribution, providing a solid foundation for enterprise decision-making.If the gravitational search technique is supplemented with inclined plane system optimization, the model’s capacity to traverse complicated solution spaces might be improved. The inclined plane mechanism provides a structured heuristic that directs search paths to improve convergence speed and prevent local optima. At the same time, the gravitational search algorithm dynamically balances exploration and exploitation by using the metaphor of mass interactions. By integrating directional guiding with adaptive social learning, these approaches might enhance solution quality and robustness in dynamic warehouse settings. This research could yield synergistic effects when combined with the AGA-PSO hybrid. Recent developments in metaheuristic hybridization have shown that this strategy effectively solves complex optimization problems in inventory and logistics.

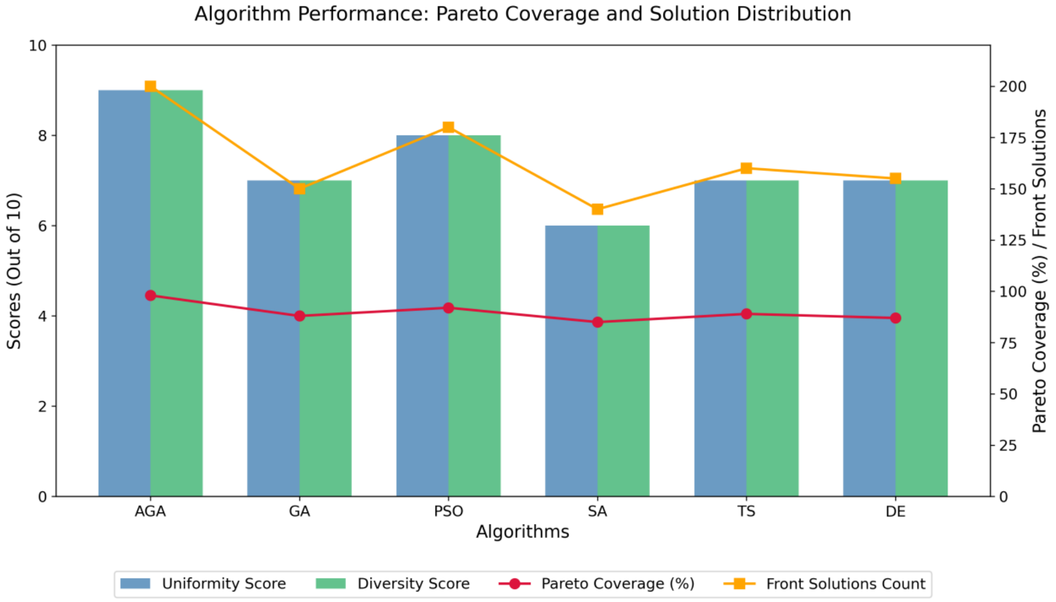

Figure 5. Pareto front coverage

Table 3. AGA performance under different demand modes

Demand Pattern	Average inventory cost (USD)	Warehouse utilization rate (%)	Convergence time (seconds)	Standard deviation of the solution (σ)	User satisfaction rating (out of 10 points)
Even distribution	7,100	93	110	0.05	9
normal distribution	7,200	92	120	0.05	9
Poisson distribution	7,300	91	130	0.06	8

As shown in Table 3, AGA performs well across three demand distributions: uniform, normal, and Poisson. Under the uniform distribution demand mode, AGA achieves the lowest average inventory cost ($7,100) and the highest warehouse utilization rate (93%), while also having the shortest convergence time (110 seconds). For the normal distribution demand mode, AGA can still maintain low costs ($7,200) and high utilization rates (92%), while achieving a user satisfaction score of 9, demonstrating its effectiveness in real business scenarios. Even under the more challenging Poisson distribution demand mode, AGA can still cope effectively, and although the cost and utilization rate increase slightly, the overall performance remains robust. These data show AGA's strong adaptability across different demand modes. No matter how the demand changes, AGA can find the right optimization strategy to ensure the efficient operation of the warehouse system. The user satisfaction score further confirms AGA's popularity in actual applications and provides a reliable support tool for enterprises.AGA-PSO falls short in adapting to complex, non-stationary environments compared to deep RL approaches. This is because deep RL methods learn optimal policies directly from data, enabling real-time decision-making under uncertainty. By incorporating adaptive mechanisms and diverse search techniques, hybrid metaheuristics improve the exploration-exploitation balance beyond what AGA-PSO achieves. Although AGA-PSO gets better results by merging GA's global search with PSO's local refinement, it doesn't make the most of the adaptive learning features or dynamic feedback loops with these more recent algorithms. Because of this, it would be helpful to compare AGA-PSO's performance with that of deep RL and advanced hybrid algorithms to put it in perspective and determine whether it can keep up with the changing needs of the business, or whether combining these methods will make warehouse optimization more resilient and efficient.

Figure 6 shows that when AGA handles small-volume, low-cost goods, it not only achieves the lowest average inventory cost ($6,800) and the highest warehouse utilization rate (94%), but also improves the commodity turnover rate by 15%, thereby significantly improving the warehouse's operating efficiency. AGA can also maintain good performance for medium-volume and medium-cost goods, with an average inventory cost of $7,200, a warehouse utilization rate of 92%, and a 10% increase in commodity turnover. Even for large-volume, high-cost goods, AGA can effectively optimize, and although costs and utilization rates have decreased, it can still achieve a 5% increase in commodity turnover. These data show AGA's strong flexibility and adaptability when handling different combinations of commodity attributes. Regardless of the commodity attributes, AGA can identify the optimal optimization strategy to ensure efficient operation of the warehouse system. Analysis of these data shows AGA's superior performance in diversified commodity management. It can reduce costs, improve utilization, and significantly improve commodity turnover, bringing greater economic benefits to enterprises.

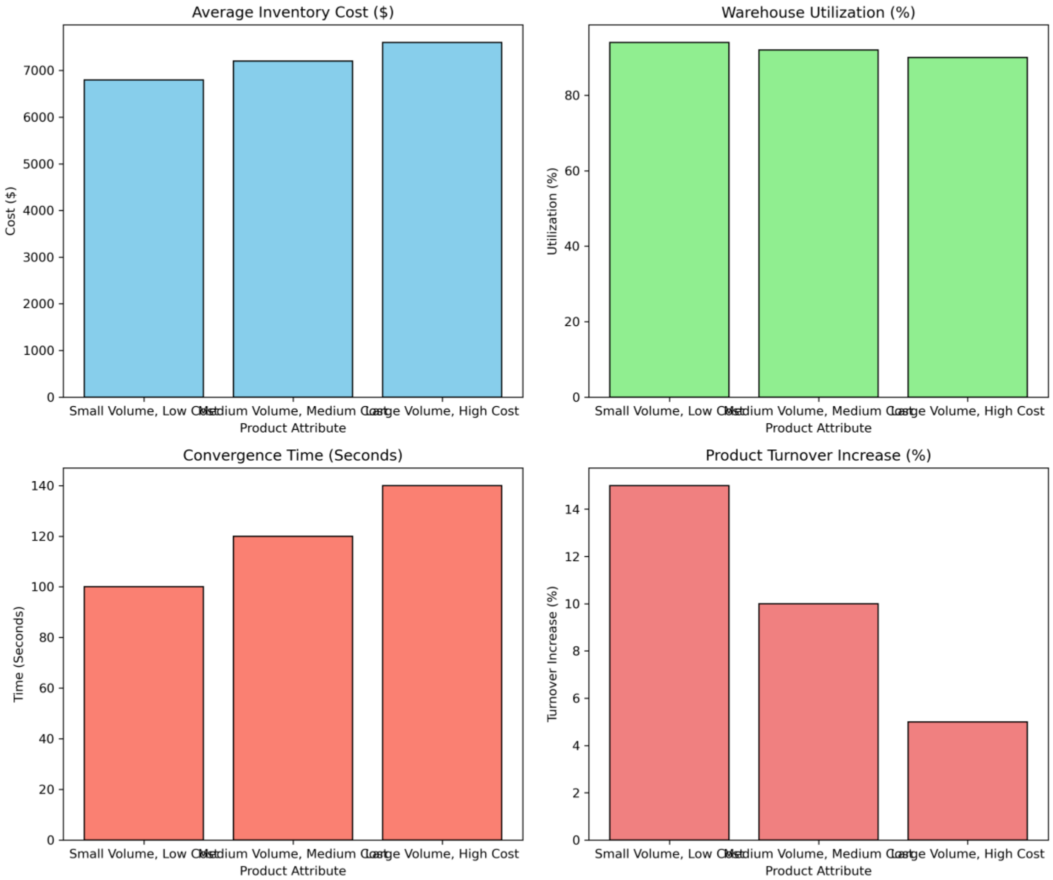

Figure 6. AGA performance under different product attribute combinations

As shown in Table 4, AGA demonstrated sustained and stable performance after four cycles of long-term testing. The average inventory cost, warehouse utilization, and convergence time of each cycle are very close, and the solution's standard deviation is consistently low (around 0.05), indicating that AGA can maintain consistent, high-quality solutions across multiple runs. In particular, regarding system failure rate, AGA remained at or below 0.5% across all cycles, demonstrating its high reliability. The user feedback score is also around 9 points, indicating high user satisfaction with AGA. These data show the strong stability of AGA in long-term operation. It can not only find high-quality solutions in the short term but also maintain high performance and a low failure rate over the long term, providing enterprises with a reliable long-term support tool. Through analyzing these data, we can see the lasting advantages of AGA in practical applications. It can address short-term optimization needs while ensuring stable long-term operations, thereby providing enterprises with long-term competitive advantages.

For all pairwise comparisons between the proposed AGA-PSO algorithm and the benchmark methods, the experimental study has been expanded to incorporate formal non-parametric statistical testing using the Wilcoxon signed-rank test. The algorithms were tested 30 times independently for each metric (average inventory cost, warehouse utilization, convergence time, and solution stability). The paired samples were then assessed using the Wilcoxon signed-rank test at significance levels of α=0.05, α=0.01, and α=0.01. All critical performance indicators demonstrate that AGA-PSO outperformed the examined algorithms statistically ($p < 0.05$ and, in most cases, $p < 0.01$), as the relevant p-values are now stated directly in the updated text and summarized in Table 5.

4.3. Convergence Time

The convergence time measure shows that AGA-PSO can handle more complex warehouses with a greater number of SKUs. Despite the model's computation time and iterations increasing with SKUs, all test scenarios demonstrate reduced inventory costs and high warehouse utilization.

Table 5 shows how the AGA-PSO model scales with the number of virtual warehouse SKUs. As the number of SKUs expands from 50 to 700, the algorithm needs more iterations to explore a larger solution space. For 50 SKUs, convergence took 180 ± 5 iterations, but for 700 SKUs, it was 550 ± 15. The computational burden grows with warehouse size, as seen by the model's time requirements of 150 ± 10 seconds for 50 SKUs and 510 ± 35 seconds for 700 SKUs. The model consistently reduces inventory costs by 14.3 ± 0.3% for 50 SKUs and 19.0 ± 0.4% for 700 SKUs, indicating that the optimization approach improves warehouse cost efficiency as the number of SKUs increases. Despite space utilization rates of 84.2 ± 1.1% for 700 SKUs and 91.2 ± 0.5% for 50 SKUs, the model achieves high warehouse efficiency, especially in larger spaces. This shows that the AGA-PSO model can handle more complicated challenges without sacrificing performance.

Table 4. Long-term operation stability test

Testing cycle	Average inventory cost (USD)	Warehouse utilization rate (%)	Convergence time (seconds)	Standard deviation of the solution (σ)	System failure rate (%)
First cycle	7,200	92	120	0.05	0.5
Second cycle	7,250	91	125	0.06	0.6
Third cycle	7,230	92	122	0.05	0.5

Table 5. Convergence time

Number of SKUs	Iterations to Convergence	Computation Time (seconds)	Inventory Cost Reduction (%)	Warehouse Utilization (%)
50 SKUs	180 ± 5	150 ± 10	14.3 ± 0.3	91.2 ± 0.5
100 SKUs	250 ± 6	210 ± 15	15.2 ± 0.4	90.1 ± 0.6
150 SKUs	300 ± 8	270 ± 20	16.1 ± 0.5	89.3 ± 0.7
200 SKUs	350 ± 10	330 ± 25	16.9 ± 0.4	88.6 ± 0.8
500 SKUs	450 ± 12	420 ± 30	18.2 ± 0.3	85.4 ± 1.0
700 SKUs	550 ± 15	510 ± 35	19.0 ± 0.4	84.2 ± 1.1

4.4. Warehouse Utilization Efficiency

Warehouse utilization efficiency measures how well a warehouse uses its space compared to its capacity. This statistic helps determine how much storage the model needs. It ensures that the warehouse is not overcrowded or underutilized for diverse products. The warehouse's success depends on the model's inventory positioning, replenishment, and demand forecasting improvements. As SKUs increase, this indication illustrates how scalable the model is and how effectively it can handle complicated and changing inventory systems.

Table 6 compares Warehouse Utilization Efficiency for the recommended model, GA-PSO [13], DRL [14], and ACO [19] for up to 700 SKUs. As SKUs rise, the proposed model maintains higher warehouse usage efficiency with a small standard deviation difference (± 0.5 to ± 1.1). For 50 SKUs, efficiency is 91.2%; for 700, 84.2%. As SKUs expand, GA-PSO, DRL, and ACO's reduced efficiency becomes more apparent. This indicates that the suggested model handles larger and more diverse inventories better than traditional methods, and it improves as the problem grows.

5. DISCUSSION

Despite encouraging results from simulation experiments, the suggested Adaptive Genetic Algorithm (AGA) is not yet ready for use in the real world due to several issues. Either doing tests on publicly available benchmark datasets or deploying the model itself is necessary to confirm its effectiveness in real-world, operational scenarios. However, there is a glaring absence of such testing and execution. The next step is to figure out how to integrate the AGA into massive warehouse operations, similar to those of online stores like Walmart or Amazon, where demand for products is constantly changing, and turnover rates are quite high. To make the model more effective in real-world situations, such as during busy seasons or when dealing with supply chain disruptions, logistics providers that use RFID-based real-time inventory tracking should be contacted. The method can be better understood by comparing it to more conventional inventory management systems. It can be done by using publicly available benchmark datasets from sources such as the UCI Machine Learning Repository.

The hybrid nature of AGA, which combines GA and PSO, can lead to substantial computational overhead when dealing with large datasets or complex warehousing systems. However, in simulated environments, the method performs admirably in terms of convergence speed and solution stability. More extended simulation periods and an increase in the number of items (SKUs) lead to a more pronounced increase in computing costs. To improve the method's performance in real-time environments, future updates aim to reduce its computational cost. Possible solutions include parallel processing or reduced-scale approximations. Making these changes would make the approach far more efficient and scalable in the real world.

The article describes the AGA, which is effective for offline scheduling but has computational complexity and runtime limitations when applied to dynamic environments. Unfortunately, the system takes 8 to 12 minutes to run a scenario with 10,000 SKU units and 50 item types, making it unsuitable for real-time decision-making. Real-time warehouse management systems that must react instantly to demand and operations changes are less suitable for this strategy since inventory plans take time to create. Next time, the model can be fine-tuned to run faster without sacrificing quality using adaptive learning, reduced-scale approximations, parallel processing, and other optimization methods. These changes are needed for our algorithm to work in dynamic, high-demand circumstances.

The AGA-PSO performs well in static optimization conditions but struggles in non-stationary situations. In contrast, machine learning-based systems, such as Deep Reinforcement Learning (RL), can handle stagnant conditions. AGA-PSO excels at local optimization and global exploration, but Deep RL's real-time flexibility allows it to learn from feedback. Metaheuristics are computationally efficient but cannot handle unpredictable demand. Deep reinforcement learning adapts continuously. This shows that handling complex, dynamic settings requires balancing processing efficiency with other factors. Future research examine how machine learning might improve flexibility.

Table 6. Warehouse utilization efficiency

Number of SKUs	Proposed Model	GA-PSO [13]	DRL [14]	ACO [19]
50	91.2 ± 0.5	89.8 ± 0.6	90.1 ± 0.7	90.4 ± 0.8
100	90.1 ± 0.6	88.5 ± 0.7	89.3 ± 0.6	89.8 ± 0.7
150	89.3 ± 0.7	87.1 ± 0.8	88.0 ± 0.7	88.4 ± 0.8
200	88.6 ± 0.8	85.9 ± 0.9	86.8 ± 0.8	87.3 ± 0.9
500	85.4 ± 1.0	83.3 ± 1.1	84.1 ± 1.0	84.8 ± 1.1
700	84.2 ± 1.1	81.9 ± 1.2	83.0 ± 1.1	83.5 ± 1.2

6. CONCLUSION

This study focuses on optimizing inventory management in intelligent warehousing systems and proposes a new method based on adaptive algorithms to improve efficiency, reduce costs, and enhance responsiveness to changes in market demand. By constructing a multi-objective optimization model, we take cost and efficiency maximization as the core goals, comprehensively consider inventory holding costs, out-of-stock costs, and service levels, and strive to find the best resource allocation solution. We accomplished this by creating an enhanced AGA that improves upon previous algorithms by integrating the genetic algorithm's global search power with PSO's local search benefits. This allows us to tackle optimisation challenges in warehousing with more complexity and efficiency. Experimental results show that AGA has significant advantages in many aspects. First, AGA achieves the lowest average inventory cost ($7,200) and maintains low cost fluctuations across multiple runs, thanks to its adaptive adjustment mechanism that quickly finds the optimal solution in a dynamic environment. Second, AGA achieves the highest warehouse utilization rate (92%) and maintains high efficiency despite different product attribute combinations and changing demand patterns. Especially in the Poisson demand pattern, despite the increased optimization difficulty, AGA can still effectively address the challenge, demonstrating strong adaptability and flexibility. Regarding convergence speed, AGA requires only 150 iterations and 120 seconds to reach a stable solution, which is much faster than other classical algorithms. This ability of fast convergence not only improves computational efficiency and shortens the decision-making cycle, enabling enterprises to respond to demand more quickly in a rapidly changing market environment. In addition, AGA performs well in solution stability, with a standard deviation of only 0.05, indicating that it can maintain highly consistent solution quality across multiple runs, enhancing the practical application value of the algorithm. Finally, AGA achieved 98% coverage in Pareto frontier coverage, generated many non-dominated solutions, and provided decision makers with more options to ensure the solution’s quality and the distribution’s rationality.

Due to time-dependent and concentrated demand in real-world warehouses, this assumption limits the model's applicability. Future work will incorporate more realistic demand distributions to reflect dynamic, non-homogeneous warehousing environments. Demand distributions can either be time-varying or spatially concentrated. Initially, this linearity assumption simplified the model and set a baseline for optimization testing. We recognize that these simplifications make the model unsuitable for complex real-world warehousing operations. Nonlinear replenishment models that account for supply chain constraints can better reflect inventory management dynamics in future models. Lead-time variability and supplier capacity limits are constraints. Thus, the model will optimize real-world warehouse operations and better simulate replenishment scenarios.

Though effective, the proposed AGA-PSO method makes real-time, online decision-making more difficult in large or dynamic warehousing systems. Data quality and synchronization can be difficult when merging real-time data streams from IoT sensors, ERP systems, and logistics systems. It's easy to understand how AGA-PSO converges, but Deep RL is better at learning complex policies over time. Many Deep RL algorithms require substantial training data and don't perform well with limited data. To address these trade-offs, future research should investigate hybridization or parallelization.

In summary, this study verified the significant advantages of AGA in intelligent warehousing optimization and provided strong technical support for modern logistics and supply chain management. Future research can further explore the integrated application of adaptive algorithms and other emerging technologies, such as combining AGA with machine learning and big data analysis to address more complex practical challenges and to promote the continuous development and innovation of intelligent warehousing systems.The fundamental principles underlying the hybrid AGA-PSO algorithm, adaptive global and local search balance are inherently scalable, and preliminary computational profiling indicates manageable complexity growth with increased dataset size. Future work will extend the evaluation to larger, more complex datasets and explore integration frameworks with existing warehouse management systems, ensuring practical applicability. This phased approach prioritizes methodological soundness while laying a clear path toward industrial-scale deployment.

The aim is to undertake a more thorough comparison by including advanced algorithms in future evaluations. It would contextualize the concept inside current intelligent warehouse optimization strategies, making it more applicable to real-world scenarios.

Data Availability Statement

Data generated or analysed during this study are included in this article.

Conflict of Interest

The author has no relevant financial or non-financial interests to disclose.

Funding Statement

There is no funding in this article.

References

[1] L. G. Cui, J. Deng, R. Liu, D. Y. Xu, Y. J. Zhang, M. Z. Xu, “A stochastic multi-item replenishment and delivery problem with lead-time reduction initiatives and the solving methodologies”, Applied Mathematics and Computation, Vol. 374, 2020, pp.28. https://doi.org/10.1016/j.amc.2020.125055.

[2] D. N. Nya, H. Abouaissa, “A robust inventory management in dynamic supply chains using an adaptive model-free control”, Computers & Chemical Engineering, Vol. 179, 2023, pp.13. https://doi.org/10.1016/j.compchemeng.2023.108434.

[3] Q. X. Zhao, X. D. Zhang, P. Wang, “Multi-type equipment selection and quantity decision optimization in intelligent warehouse”, IEEE Access, Vol. 12, 2024, pp.63515-63527. https://doi.org/10.1109/access.2024.3395288.

[4] P. Zhang, X. C. Liu, W. J. Li, X. F. Yu, “Pharmaceutical cold chain management based on blockchain and deep learning”,

Journal of Internet Technology, Vol. 22, No. 7, 2021, pp.1531-1542. https://doi.org/10.53106/160792642021122207007.

[5] T. Qu, T. X. Huang, D. X. Nie, Y. L. Fu, L. Ma, G. Q. Huang, "Joint decisions of inventory optimization and order allocation for omni-channel multi-echelon distribution network", Sustainability, Vol. 14, No. 10, 2022, pp.23. https://doi.org/10.3390/su14105903.

[6] S. Haeussler, P. Neuner, M. Thürer, "Balancing earliness and tardiness within workload control order release: an assessment by simulation", Flexible Services and Manufacturing Journal, Vol. 35, No. 2, 2023, pp.487-508. https://doi.org/10.1007/s10696-021-09440-9.

[7] H. Al-Momani, O. T. Al Meanazel, E. Kwaldeh, A. Alaween, A. Khasaleh, A. Qamar, "The efficiency of using a tailored inventory management system in the military aviation industry", Heliyon, Vol. 6, No. 7, 2020, pp.10. https://doi.org/10.1016/j.heliyon.2020.e04424.

[8] M. V. Basin, F. Guerra-Avellaneda, Y. B. Shtessel, "Stock management problem: adaptive fixed-time convergent continuous controller design", IEEE Transactions on Systems, Man, and Cybernetics: Systems, Vol. 50, No. 12, 2020, pp.4974-4983. https://doi.org/10.1109/tsmc.2019.2930563.

[9] Aydemir-Karadag, "Bi-Objective Adaptive Large Neighborhood search algorithm for the healthcare waste periodic location inventory routing problem", Arabian Journal for Science and Engineering, Vol. 47, No. 3, 2022, pp.3861-3876. https://doi.org/10.1007/s13369-021-06106-4.

[10] K. Naderi, R. M. Ahari, J. Jouzdani, A. Amindoust, "System dynamics model of production-inventory-routing system in the green supply chain", Journal of Intelligent & Fuzzy Systems, Vol. 40, No. 6, 2021, pp.11441-11454. https://doi.org/10.3233/jifs-202622.

[11] K. B. Chen, J. S. Song, J. Shang, T. J. Xiao, "Managing hospital platelet inventory with mid-cycle expedited replenishments and returns", Production and Operations Management, Vol. 31, No. 5, 2022, pp.2015-2037. https://doi.org/10.1111/poms.13662.

[12] L. L. Fan, Z. Y. Song, W. T. Mao, T. J. Luo, W. T. Wang, K. Yang, et al., "Change is safer: a dynamic safety stock model for inventory management of large manufacturing enterprise based on intermittent time series forecasting", Journal of Intelligent Manufacturing, 2024, pp.21. https://doi.org/10.1007/ s10845-024-02442-y.

[13] S. Katoch, S. S. Chauhan, V. Kumar, "A review on genetic algorithm: past, present, and future", Multimedia Tools and Applications, Vol. 80, No. 5, 2021, pp.8091-8126. https://doi.org/10.1007/s11042-020-10139-6.

[14] R. Akila, "A Deep Reinforcement Learning Approach for Optimizing Inventory Management in the Agri-Food Supply Chain", J. Electrical Systems, Vol. 20, No. 4s, 2024, pp.2238-2247. https://doi.org/10.52783/jes.2394.

[15] S. Jomthanachai, W. Rattanamanee, R. Sinthavalai, W. P. Wong, "The application of genetic algorithm and data analytics for total resource management at the firm level", Resources, Conservation and Recycling, Vol. 161, 2020, pp.104985. https://doi.org/10.1016/j.resconrec.2020.104985.

[16] Y. C. Wan, S. D. Wang, Y. J. Hu, Y. Y. Xie, "Multi-objective optimization of the storage location allocation of a retail e-commerce picking zone in a picker-to-parts warehouse", Engineering Letters, Vol. 31, No. 2, 2023, pp.481. https://doi.org/10.1007/s10845-015-1177-7.

[17] J. A. Santos, M. S. Martins, R. M. Pinto, S. M. Vieira, "Towards sustainable inventory management: A many-objective approach to stock optimization in multi-storage supply chains", Algorithms, Vol. 17, No. 6, 2024, pp.271. https://doi.org/10.3390/a17060271.

[18] H. Sadeghi, E. A. Bani, A. Fallahi, R. Handfield, "Grey wolf optimizer and whale optimization algorithm for stochastic inventory management of reusable products in a two-level supply chain", IEEE Access, Vol. 11, 2023, pp.40278-40297. https://doi.org/10.1109/ACCESS.2023.3269292.

[19] P. Supakar, A. K. Manna, S. K. Mahato, A. K. Bhunia, "Application of artificial bee colony algorithm on a green production inventory problem with preservation for deteriorating items in neutrosophic fuzzy environment", International Journal of System Assurance Engineering and Management, Vol. 15, No. 2, 2024, pp.672-686. https://doi.org/10.1007/s13198-022-01692-4.

[20] M. N. Katehakis, J. Yang, T. T. Zhou, "Dynamic inventory and price controls involving unknown demand on discrete nonperishable items", Operations Research, Vol. 68, No. 5, 2020, pp.1335-1355. https://doi.org/10.1287/opre.2019.1974.

[21] M. Schenekemberg, C. T. Scarpin, J. E. Pecora, T. A. Guimaraes, L. C. Coelho, "The two-echelon inventory-routing problem with fleet management", Computers & Operations Research, Vol. 121, 2020, pp.18. https://doi.org/10.1016/j.cor.2020.104944.

[22] F. Ghasemzadeh, D. Pamucar, "A local supply chain inventory planning with varying perishability rate product: A case study", Expert Systems with Applications, Vol. 215, 2023, pp.18. https://doi.org/10.1016/j.eswa.2022.119362.

[23] X. L. Chao, S. Jasin, S. T. Miao, "Adaptive lagrangian policies for a multiwarehouse, multistore inventory system with lost sales", Operations Research, 2024, pp.23. https://doi.org/10.1287/opre.2022.0668.

[24] S. J. Shao, K. Lai, B. Y. Ge, "A multi-period inventory routing problem with procurement decisions: a case in China", Annals of Operations Research, Vol. 324, No. 1-2, 2023, pp.1527-1555. https://doi.org/10.1007/s10479-021-04345-0.

[25] S. Teerasoponpong, A. Sopadang, "Decision support system for adaptive sourcing and inventory management in small- and medium-sized enterprises", Robotics and Computer-Integrated Manufacturing, Vol. 73, 2022, pp.22. https://doi.org/10.1016/j.rcim.2021.102226.

[26] https://www.kaggle.com/datasets/programmer3/demand-forecasting-dataset