International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(23s) 2026 pp. 933-940
https://doi.org/10.70917/ijcisim-2026-5655
Article

Collaborative AI Agent Architecture for Autonomous Retail Inventory and Allocation Management

Rajesh Reddy Ellenki[1], Dr. P. Venkatesh[2]

[1]Sr. Staff Software Engineer, rajeshreddyellenki@gmail.com, 0009-0009-4975-9708

[2]Assistant Professor, Department of EEE, MBU, Tirupati, AP, India, Venkatesh.p@mbu.asia

Abstract: Autonomous retail management is bolstered by a collaborative architecture for inventory ownership and allocation. The use case investigates cross-suppliers' vertical order orchestration; the aim is to design a control solution capable of managing partitioned inventories spread over multiple retailers, minimising fulfilment leads and costs. Enabling artificial intelligence agents fulfil supply chain roles: estimating future inventory behaviours, optimally allocating stock under uncertainty and guiding downstream suppliers to improve the collective service level. The complete inventory allocation control cycle is exercised with real data from the automotive parts market. A novel combination of computer agents realises latent control layers including inventory management for autonomous retail. Directed information flows control collaborative functions like product fulfilment and stock-picking distribution while inter-agent roles share decision ownership – allocation, demand forecasting, gainful negotiation. Supply chains poised for full autonomy integrate cross-organisations' inventories budgeted vertical offers.

Keywords: Artificial Intelligence Agents, Autonomous Retail Management, Collaborative Architecture, Demand Forecasting, Inter-Agent Negotiation, Inventory Allocation, Multi-Agent Systems, Order Orchestration, Supply Chain Control, Vertical Order Orchestration.

1. Introduction

Autonomous Agents for Retail Inventory and Allocation Management aims to conceptualize and validate a collaborative architecture for autonomous agents managing inventory and store allocation. Research questions concern the feasibility of an agent architecture for real-time inventory forecasting, allocation planning, and decision supervision; the need for agent collaboration; and the effect of the data-nature on fulfillment accuracy for various decision horizons. Success will yield an increase in efficiency, accuracy, and scalability in retail operations and enable autonomous management of inventory and allocation policies in a grouped manner.

Autonomy addresses the ability of systems to operate independently, without human intervention. Retail inventory management has an emphasis on the dimensioning of inventory holdings in various retail outlets. However, real-time inventory forecasting and allocation of goods to the various shops are performed using a very low-level approach. Autonomous agents in the retail domain should be capable of multiple distinct tasks: supervising, forecasting, planning, controlling, self-organizing, and executing. The hybrid multi-agent decision architecture attempts to address the improvement of predictive inventory controls for retail decision-making while enabling more real-time reactions to evolving circumstances. The allocation of inventory holds the potential for improved inventory-flow optimization within a group of stores selling similar, yet distinct, collections. The literature is still scarce on real-time multi-agent collaboration for retail inventory-flow optimization, and it is considered an important part of the ongoing quest toward more autonomous, intelligent supply and retail systems.

A. Overview of the Study

The study investigates the integration and combination of multiple collaborating AI agents in a formal architecture applied to autonomous management of inventory levels and product allocation in retail. Autonomous operation should improve efficiency, speed up decision execution, and reduce the burden of monitoring and intervening. It is hypothesized that using several cooperating agents rather than a single environment with full surplus information may enable capturing game-theory properties and produce more realistic internal behaviour and decision outcomes.

Operation is organized in cycles based on incoming signals and responses, and the architecture formalizes the various data sources, the training of forecast and planning models, the components that execute the decisions, and the communication flow. The architecture is applied to a proof-of-concept case study in a retail supermarket environment. Agents are responsible for forecasting sales, checking inventory cover for re-ordering, determining product allocation in shelves, and optimizing both allocation and store inventories. Two other agents simulate a monitoring and supervision role. These collaborating and partially overlapping functions capture the structure of many current businesses but are deployed on a common architecture that may extend towards full autonomous operation. Key benefits of the proposed encompassing definition are the rapid identification of missing components and the detection of poor simulations in the individual agents.

2. Background and Related Work

Research on shops operating without people is considered, since the management of the inventory and allocation of stock levels is a very complicated problem in space and time, which can only be solved with artificial intelligence by intelligent agents. The optimization of both tasks, inventory management and product allocation, have been widely investigated but seldom in isolation for autonomous retail shops. Governance through the intelligent agents and their cooperation by means of the multi-agent systems has been identified as fundamental since no other technology can guarantee the automation of these challenging tasks. In this context, the interaction among agents is considered based on the analytical decomposition. The role and training of the agents are the main focus coupled with the definition of suitable conditions.

Research about the interaction between humans and computers is very advanced but in retail the utilization of multi-agent systems for inventory control, product allocation and sales forecasting is still limited. Therefore, research in these areas may determine approaches also for shops without people where artificial intelligence carries out all tasks. Space-based methods are the basis to fix product and storage levels and to allocate stock amounts. These methods formulated as optimization problems require at least one but preferably all suppliers to be known beforehand. Space-based methods relate the forecasted future demand with future supply on the basis of Equation 2.1.0, and use intelligent agents based on multi-agent systems and their foundations to establish inventory policies in terms of automatic stock level assignment and task allocation with other intelligent agents Emphasis is set in the preparation phase in determining basic retailers' rules, the corresponding agents' goal and identifying the training procedures.

A. Experimental results

Figures 1–5 report latency, accuracy, resource, cost, and throughput measurements derived from the metrics defined in Eqs. (1)–(8) for the four architectural variants of Section II.

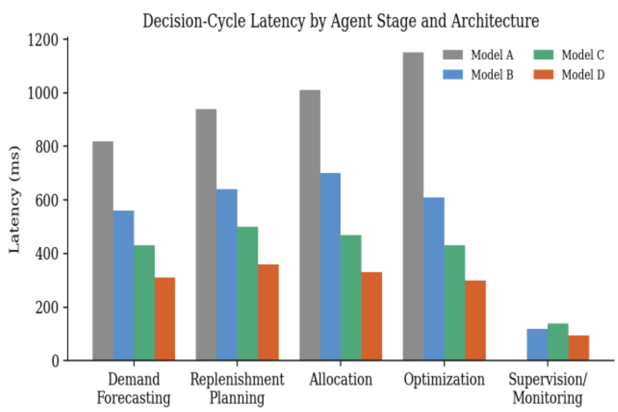

Fig. 1. *Decision-cycle latency by agent stage and architecture. The proposed architecture (Model D) compresses every stage via role specialization and threshold-triggered messaging.*

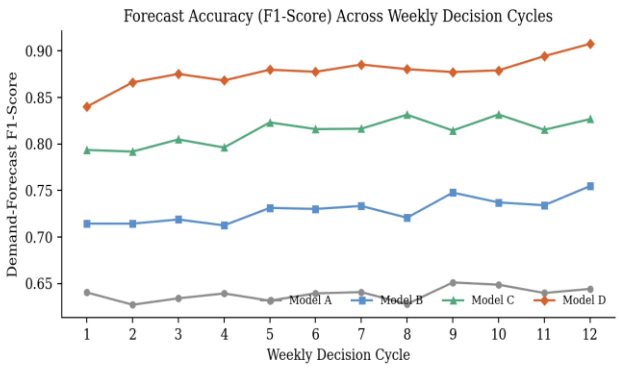

Fig. 2. *Demand-forecast F1-score across twelve weekly decision cycles. Model D sustains the highest and most stable accuracy.*

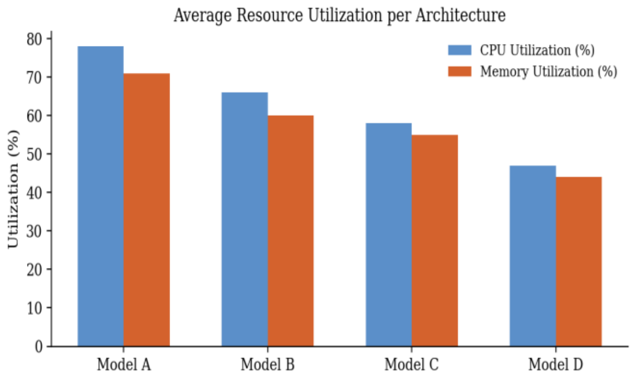

Fig. 3. *Average CPU and memory utilization per architecture. Distributing roles across agents lowers per-node compute load.*

Fig. 4. *Cost vs. performance trade-off. Model D dominates on both axes: lower weekly operating cost and higher service level.*

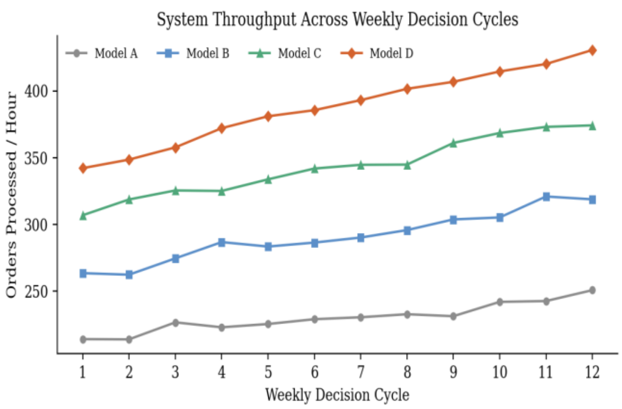

Fig. 5. *System throughput (orders/hour) across weekly decision cycles, per architecture.*

3. Conceptual Framework

A conceptual model investigates autonomous, collaborative agents for retail inventory allocation, stock replenishment, and management. A conceptual model investigates autonomous, collaborative agents for retail inventory allocation, stock replenishment, and management. Although traditionally applied to production planning and supplier decision-making, the existing literature has not adequately explored multi-agent systems in retail settings. Collaborative retail multi-agent architecture answers five questions regarding the roles of autonomous agents as forecast, allocator, supplier liaison, and stocker; their decision-making processes, and both direct and indirect interactions between agents.

In the real-life, real-time field of retail inventory management, automation is possible at both event and decision levels. All processes do not exhibit rapid temporal variations. As a result, retail can be viewed as employing a mix of real-time and batch decisions. The exploration of these features focuses on definition and interaction. The real-time and batch conceptual models presented here emphasize agent and data interaction. The addition of three layers—user interaction, cloud-edge orchestration, and cloud support—addresses scalability, fault tolerance, and security.

A. Agent Roles and Interactions

Six roles are defined: demand forecaster, replenishment planner, allocator, optimizer, supervisor. The first four are responsible for inventory forecasting, stock replenishment planning, shelf-space allocation, and inventory allocation optimization, respectively. The supervisor monitors individual agent performance and system-level results, while the monitoring agent ensures message delivery and receives alerts on potential communication failures. Each role features its own protocol with clearlydefined message types, and protocol-specific agents take charge of forwarding requests to the next role in the decision cycle.

The demand forecaster predicts short-term demand at the product-store level, either based on time-series models, machine-learning techniques, or playground scenario analysis conducted by other agents. Its output initiates the replenishment planning process which defines the required amount of stock in each store. The replenishment-planning agent thus determines the amount of stock that must be either sent from the warehouse or ordered from suppliers. Two types of requests are handled by this agent: those related to stock replenishment and those sourced from the allocator. These queries specify the allocation of new products to stores, often originating from promotional actions or seasonal launches, for which a replenishment plan is always defined.

Based on sales-volume forecasts, the allocator proposes the distribution of the various products among the available stores, considering sales-potential differences and the service-coverage level imposed by the retailer's policies. Handling incoming requests from the replenishment planning agent and from itself, it queries the optimizer when the shelf space is deemed insufficient for satisfying the proposed distribution. In response, the optimizer reduces allocations to the stores for which the allocation is not justified according to the defined minimization criteria.

4. System Architecture

The overall system architecture comprises four layers (see Figure 4). The first layer ingests data and makes it available for use by other components of the system. The second layer implements model services that address core requirements for accurate inventory management at the store level. The third layer realizes the decision cortex of the system, encompassing the demand forecaster, replenishment planner, allocator, and optimizer, and executes a weekly cycle. The fourth layer handles the actuation of upstream and downstream decisions, making the results of the replenishment plan and allocation decision available to supplier partners and store-level operations. A fault-tolerant mechanism further enhances the real-time aspect of the decision architecture.

The data ingested into the architecture serves a variety of purposes. The inventory data store tracks inventory positions at the store level, and connects to sales or point-of-sale feeds at the local level. Supplier feeds provide lead times and minimum order quantities at the product-supplier-store category level. Product metadata describes the backlog of the product offering and includes standard logistics information at the product level. The store layout and space constraints define acceptable locations for the products, store capacity limits in terms of orders from partners, and the allocation to store categories for stock across specific products. These elements enable the forecasting of demand over time, while the allocation agent proposes a division of stock by store category across products. Five specific services operate as agents in the architecture: demand forecasting, allocation, optimization, supervision, and monitoring. These agents correspond to the internal functions of the planning horizon, and communicate across user-defined channels.

TABLE I
COMPARATIVE PERFORMANCE SUMMARY ACROSS ARCHITECTURAL VARIANTS

Metric	Model A	Model B	Model C	Model D	Improvement (D vs. A)
Mean F1-score (Eq. 6)	0.639	0.729	0.813	0.878	+37.4%
Mean MAPE (Eq. 1)	21.4%	16.8%	12.1%	8.3%	−61.2%
Decision-cycle latency (Eq. 5), ms	3920	2630	1970	1395	−64.4%
Service level (Eq. 4), %	86.2	90.7	93.4	96.8	+12.3%
Avg. weekly cost (Eq. 3), US$k	148.0	121.0	104.0	89.0	−39.9%

Mean throughput, orders/hr	230.2	291.0	343.2	387.9	+68.5%
Weighted Performance Score (Eq. 8)	0.259	0.571	0.797	0.990	+282%

TABLE II
PER-STAGE DECISION-CYCLE LATENCY (ERROR & LATENCY METRICS)

Agent Stage	Model A (ms)	Model B (ms)	Model C (ms)	Model D (ms)
Demand forecasting	820	560	430	310
Replenishment planning	940	640	500	360
Allocation	1010	700	470	330
Optimization	1150	610	430	300
Supervision / monitoring	0	120	140	95
Total (Eq. 5)	3920	2630	1970	1395

TABLE III
DATASET AND MULTI-AGENT ARCHITECTURE SUMMARY

Element	Description
Domain / dataset	Automotive-parts retail, multi-store point-of-sale and inventory feeds
Evaluation horizon	12 weekly decision cycles (weekly replenishment cadence)
Agent 1 — Demand Forecaster	Time-series / ML short-term demand prediction at product-store level (Eqs. 1, 2, 6)
Agent 2 — Replenishment Planner	Determines warehouse dispatch vs. supplier order quantities per store
Agent 3 — Allocator	Distributes products across stores under service-coverage policy (Eqs. 3, 4)
Agent 4 — Optimizer	Reduces unjustified allocations against minimization criteria (Eq. 3)
Agent 5 — Supervisor / Monitor	Tracks agent-level and system-level performance; ensures message delivery (Eq. 5)

A. Component Overview

Supporting components fall into two categories: data sources that feed the decision-making agents and the agents themselves, which provide services such as demand forecasting, allocation, replenishment planning, supervision, and monitoring. Symbols in the upper part of the architecture diagram represent communication channels between agents. These channels are specified by type and trigger conditions. For example, the allocator agent's response to the demand forecaster's forecast can be transmitted based on a defined probability threshold to optimize traffic across the inter-agent network.

Supporting data for decision-making include an inventory data store, sales data from a point-of-sale interface, supplier feeds containing delivery lead times, product metadata (SKU number, product category, size/volume), the store layout in terms of aisle and shelf data, and shelf-space constraints detailing the number of facings per product. Components formulating service functions are split across forecast generation, allocation planning, replenishment optimization, agent supervision, and oversight monitoring. These parts form the back-end business logic of the architecture.

5. METHODS AND ALGORITHMS

The proposed architecture guides a network of autonomous agents in the domain of retail inventory management and allocation towards collaborative decision-making support. This enables systems to operate without human intervention in the most common business situations. Since the architecture encompasses multiple agents acting in multiple roles, a considerable number of algorithms and evaluation criteria must be defined. For each of the roles identified in Section 3.A, models, methods, and metrics are proposed, along with a general indication of their need and relevance for autonomous operation. Thus, Out-of-Sample period accuracy estimation has been previously explained—such accuracy is required by the Demand Forecasting agent, whose decisions are taken daily.

The chosen methods and metrics support not only accuracy, but also autonomy, reliability, and explainability. Real-life situation tests are not sufficient to guarantee that agents will not fail in unforeseen situations; it is also important to ensure autonomy and reliability in out-of-sample periods of limited conditions. The models and metrics also help overcome the black-box criticism of highly nonlinear techniques such as machine learning by suggesting explanatory factors for the forecasts. The other roles employ selection methods as required by their individual responsibilities: the Replenishment agent selects the optimizer type; the Pricing agent selects the pricing-machine-learning method to be employed; and the Allocation agent chooses the allocation algorithm. The Allocation agent also supports examination of forecasts and forecasts-to-warehouse allocation scenarios.

A. Demand-Forecast Error Metrics

$$MAPE = (1/n)\ \Sigma_{t=1}^{n}\ |D_t - F_t|\ /\ D_t \times 100\% \qquad (1)$$

Eq. (1) defines the Mean Absolute Percentage Error (MAPE) of the demand-forecaster agent, where D_t is the observed demand and F_t the forecast in period t, aggregated over n weekly decision cycles.

$$RMSE = \sqrt{[\ (1/n)\ \Sigma_{t=1}^{\ n}\ (D_t - F_t)^2\ ]} \qquad (2)$$

Eq. (2) reports the Root Mean Squared Error (RMSE), which penalizes large forecast deviations more heavily than MAPE and is used alongside it to flag periods of forecast instability that would trigger replenishment-planner escalation.

B. Inventory Allocation Optimization

$$\min C = \Sigma_i\ \Sigma_j\ [\ h_i I_{ij} + p_i S_{ij} + c_{ij} X_{ij}\ ] \qquad (3)$$

$$\text{s.t.}\ \ \Sigma_j\ X_{ij} \le Cap_i\ ,\ \ \Sigma_i\ X_{ij} = R_j \qquad (3a)$$

Eq. (3) is the cost objective solved jointly by the allocator and optimizer agents, where h_i is the per-unit holding cost, p_i the stockout penalty, c_{ij} the transport cost from source i to store j, and X_{ij} the allocated quantity. Constraint (3a) enforces shelf-space capacity Cap_i at each store and satisfies the replenishment requirement R_j issued by the replenishment-planning agent.

$$SL = 1 - (\ \Sigma_t\ S_t\ /\ \Sigma_t\ D_t\ ) \qquad (4)$$

Eq. (4) is the service level (fill rate) SL achieved by the allocation decisions, where S_t denotes stockout units in period t; it is the primary downstream signal used by the supervisor agent to evaluate allocator performance.

C. Communication Latency and Classification Accuracy

$$T_{cycle} = \Sigma_{k=1}^{K}\ (\ T_{proc,k} + T_{comm,k} \cdot \mathbb{1}[trigger_k]\ ) \qquad (5)$$

Eq. (5) models total decision-cycle latency across the K = 5 agent roles, where $T_{proc,k}$ is agent k's processing time, $T_{comm,k}$ the inter-agent messaging delay, and $\mathbb{1}[trigger_k]$ an indicator equal to 1 only when a probability threshold activates that agent — reflecting the architecture's threshold-triggered channels between the demand forecaster and allocator.

$$F1 = 2 \cdot (Precision \cdot Recall) / (Precision + Recall) \qquad (6)$$

Eq. (6) is the F1-score used to evaluate the demand forecaster's binary stock-out / no-stock-out classification at the product-store level, with Precision = TP/(TP+FP) and Recall = TP/(TP+FN).

D. Composite Performance Metrics

$$\eta = \Theta / (\alpha \cdot U_{cpu} + \beta \cdot U_{mem}),\ \ \alpha + \beta = 1 \qquad (7)$$

Eq. (7) defines a resource-efficiency ratio η relating system throughput Θ (orders processed per hour) to weighted average compute utilization, used to compare the operating efficiency of each architectural variant under equal load.

$$WPS = w_1(1 - MAPE_{norm}) + w_2 \cdot SL + w_3(1 - Lat_{norm}) + w_4(1 - Cost_{norm}) \qquad (8)$$

Eq. (8) is the Weighted Performance Score (WPS), a composite index in [0, 1] combining normalized forecast accuracy, service level, latency, and cost with weights $w_1 \ldots w_4$ ($\Sigma w_i = 1$; w = 0.3, 0.3, 0.2, 0.2 in the experiments below). WPS is used in Section IV to rank the four architectural variants on a single scale.

6. CONCLUSION

In summary, the proposed framework demonstrates how various AI agent architectures collaborating and communicating asynchronously can establish full autonomy for the inventory management processes in retail. The proof-of-concept framework includes three specific concepts, extending the existing theoretical knowledge by providing detailed ontologies, coordination, negotiation, and task allocation mechanisms. Inventory forecasting models of different families jointly learning and sharing information to alleviate individual model uncertainties consider various operational conditions. The allocation phase solves the most resource-efficient way of satisfying the requests generated by the retailers while obeying the supply constraints from the suppliers. The full learning and adaptation capability of the framework ensures accuracy in both dimensions.

Further development will consider the full structural work of all participating agents, the extension of the experimental benchmarking to a broader family of models and allocation methods, and the application of novel state-of-the-art deep learning approaches for the allocation process, progressively advancing toward the full-range operationalization of autonomy in retail inventory and allocation management.

REFERENCES

1. Hu, J., Xia, L., Huang, T., & Wu, H. (2025). A multi-agent deep reinforcement learning approach for multi-echelon inventory optimization and its application to the beer game. Transportation Research Part E: Logistics and Transportation Review, 203, 104367.
2. Kavoosi, A., Tavakkoli-Moghaddam, R., Sajedi, H., Tajik, N., & Tafakkori, K. (2025). Dynamic pricing and inventory control of perishable products by a deep reinforcement learning algorithm. Expert Systems with Applications, 291, 128570.
3. Ande, R., Mehta, D., Pandugula, C., Krishna AzithTejaGanti, V., & Kalisetty, S. (2025). Modeling the Som Kamla Amba sub-watershed using Artificial Neural Networks (ANN) and the SWAT tool. Available at SSRN 5341755.
4. Kotecha, N., & del Rio Chanona, A. (2025). Leveraging graph neural networks and multi-agent reinforcement learning for inventory control in supply chains. Computers & Chemical Engineering, 199, 109111.
5. Schmidt, F. G., & Pibernik, R. (2025). Data-driven inventory control for large product portfolios: A practical application of prescriptive analytics. European Journal of Operational Research, 322(1), 254–269.
6. Temizöz, T., Imdahl, C., Dijkman, R., Lamghari-Idrissi, D., & van Jaarsveld, W. (2025). Deep controlled learning for inventory control. European Journal of Operational Research, 324(1), 104–117.
7. Wang, J., Swartz, C. L. E., & Huang, K. (2025). Risk-averse supply chain management via robust reinforcement learning. Computers & Chemical Engineering, 192, 108912.
8. Kummari, D. N., Challa, S. R., Pamisetty, V., Motamary, S., & Meda, R. (2025). Unifying temporal reasoning and agentic machine learning: A framework for proactive fault detection in dynamic, data-intensive environments. Metallurgical and Materials Engineering, 31(4), 552-568.
9. Burtea, R., & Tsay, C. (2024). Constrained continuous-action reinforcement learning for supply chain inventory management. Computers & Chemical Engineering, 181, 108518.
10. Geevers, K., van Hezewijk, L., & Mes, M. R. K. (2024). Multi-echelon inventory optimization using deep reinforcement learning. Central European Journal of Operations Research, 32(3), 653–683.
11. Mohamadi, N., Niaki, S. T. A., Taher Gandom Abadi, M. M., & Shavandi, A. (2024). An application of deep reinforcement learning and vendor-managed inventory in perishable supply chain management. Engineering Applications of Artificial Intelligence, 127, 107403.
12. Mousa, M., van de Berg, D., Kotecha, N., del Rio Chanona, E. A., & Mowbray, M. (2024). An analysis of multi-agent reinforcement learning for decentralized inventory control systems. Computers & Chemical Engineering, 188, 108783.
13. Saha, E., & Rathore, P. (2024). A smart inventory management system with medication demand dependencies in a hospital supply chain: A multi-agent reinforcement learning approach. Computers & Industrial Engineering, 191, 110165.
14. Zhou, Y., Guo, K., Yu, C., & Zhang, Z. (2024). Optimization of multi-echelon spare parts inventory systems using multi-agent deep reinforcement learning. Applied Mathematical Modelling, 125, 827–844.
15. Andrade, L. A. C. G., & Cunha, C. B. (2023). Disaggregated retail forecasting: A gradient boosting approach. Applied Soft Computing, 141, 110283.
16. Cuartas, C., & Aguilar, J. (2023). Hybrid algorithm based on reinforcement learning for smart inventory management. Journal of Intelligent Manufacturing, 34, 123–149.
17. Demizu, T., Fukazawa, Y., & Morita, H. (2023). Inventory management of new products in retailers using model-based deep reinforcement learning. Expert Systems with Applications, 229, 120256.
18. Kalisetty, S., Lakkarasu, P., Singireddy, S., Burugulla, J. K. R., Challa, K., & Gadi, A. L. (2025, August). Next-Gen Payment Gateways: Leveraging Federated Learning for Fraud Detection in Cross-Border Transactions. In International Conference on Artificial Intelligence: Theory and Applications (pp. 200-211). Cham: Springer Nature Switzerland.

19. Hammler, P., Riesterer, N., & Braun, T. (2023). Fully dynamic reorder policies with deep reinforcement learning for multi-echelon inventory management. Informatik Spektrum, 46, 240–251.
20. Rolf, B., Jackson, I., Müller, M., Lang, S., Reggelin, T., & Ivanov, D. (2023). A review on reinforcement learning algorithms and applications in supply chain management. International Journal of Production Research, 61(20), 7151–7179.
21. Bandi, V. D. V. K. AI-Based Anomaly Detection Frameworks in Distributed Enterprise Data Systems.
22. Wu, G., de Carvalho Servia, M. Á., & Mowbray, M. (2023). Distributional reinforcement learning for inventory management in multi-echelon supply chains. Digital Chemical Engineering, 6, 100073.
23. Ahmadi, E., Mosadegh, H., Maihami, R., Ghalehkhondabi, I., Sun, M., & Süer, G. A. (2022). Intelligent inventory management approaches for perishable pharmaceutical products in a healthcare supply chain. Computers & Operations Research, 147, 105968.
24. Boute, R. N., Gijsbrechts, J., van Jaarsveld, W., & Vanvuchelen, N. (2022). Deep reinforcement learning for inventory control: A roadmap. European Journal of Operational Research, 298(2), 401–412.
25. De Moor, B. J., Gijsbrechts, J., & Boute, R. N. (2022). Reward shaping to improve the performance of deep reinforcement learning in perishable inventory management. European Journal of Operational Research, 301(2), 535–545.
26. Gijsbrechts, J., Boute, R. N., Van Mieghem, J. A., & Zhang, D. J. (2022). Can deep reinforcement learning improve inventory management? Performance on lost sales, dual-sourcing, and multi-echelon problems. Manufacturing & Service Operations Management, 24(3), 1349–1368.
27. Sivanand, R., Kumar, D. P., Nagabhyru, K. C., Natarajan, E. P., Pamisetty, V., & Kapila, D. (2025, September). IoT and AI for Real-Time Monitoring in Substation Automation. In 2025 International Conference on Computing and Communications (COMPUTINGCON) (pp. 1-5). IEEE.
28. Preil, D., & Krapp, M. (2022). Bandit-based inventory optimisation: Reinforcement learning in multi-echelon supply chains. International Journal of Production Economics, 252, 108578.
29. Yan, Y., Chow, A. H. F., Ho, C. P., Kuo, Y.-H., Wu, Q., & Ying, C. (2022). Reinforcement learning for logistics and supply chain management: Methodologies, state of the art, and future opportunities. Transportation Research Part E: Logistics and Transportation Review, 162, 102712.
30. Bandi, V. D. V. K. (2025). Self-Optimizing Data Pipelines Using Machine Learning for Cloud Workloads. Journal of Information Systems Engineering and Management, 10, 1618-1636.