Spektrum Industri
Vol. 24, No. 2, 2026, pp. 366–378
ISSN (print) 1693-6590 | ISSN (online) 2442-2630
https://journal3.uad.ac.id/index.php/spektrum/index

A Trend-Driven Framework for Integrated SKU-to-Pod Assignment and Replenishment Policy in Robotic Mobile Fulfillment Systems (RMFS)

Salsabila Herwening Mileniadewi [a,b*], Shuo-Yan Chou [a], Sinta Rahmawidya Sulistyo [b], Anna Maria Sri Asih [b]

[a] Department of Industrial Management, National Taiwan University of Science and Technology, Taipei 106335, Republic of China
[b] Department of Mechanical and Industrial Engineering, Universitas Gadjah Mada, Yogyakarta 55281, Indonesia
* Corresponding Author: salsamileniadewi@gmail.com

ARTICLE INFO

Article history
Received July 5, 2026
Revised September 1, 2026
Accepted September 24, 2026

Keywords
Agent based modelling;
Robotic mobile fulfillment system;
Replenishment policy;
SKU-to-pod assignment.

ABSTRACT

Robotic Mobile Fulfillment System (RMFS) performance is greatly affected by stock-keeping unit (SKU) assignment and replenishment decisions, especially with the changing e-commerce demand. However, current studies typically treat these decisions in isolation, without paying adequate attention to changing demand patterns. The aim of this study is to develop and evaluate a trend-driven framework that combines demand-based SKU clustering, SKU-to-pod assignment and dynamic replenishment strategies for improving the operational performance of RMFS. The proposed framework uses machine-learning based clustering to group SKUs according to their demand characteristics, and then implements capacity-constrained pod assignment and replenishment policies for each cluster. An agent-based simulation was conducted with ten replications to test the proposed strategy against a static baseline, employing order throughput, replenishment-to-picking ratio, and pod utilization as performance indicators. The results show that the proposed framework significantly increases the order throughput from 74.75% to 93.44% (an increase of 18.69 percentage points) and significantly decreases the replenishment-to-picking ratio from 38.98% to 34.56% (a decrease of 4.42 percentage points). Pod utilization remains unchanged, meanwhile. These results suggest that the combination of demand-driven assignment and adaptive replenishment can improve fulfillment efficiency while maintaining pod utilization. This paper proposes an integrated simulation-based decision-support framework for demand clustering, pod assignment and replenishment decisions to enable adaptive RMFS operations under volatile e-commerce demand.

©The Author(s)
This is an open-access article under the CC–BY-SA license.

1. Introduction

The rapid growth of e-commerce has significantly transformed warehouse management, requiring warehouses to handle volatile demand while maintaining high operational efficiency (Boysen et al., 2019). To address these challenges, innovative technological solutions have been introduced, one of which is the Robotic Mobile Fulfillment System (RMFS), which combines mobile

robots, picking stations, and portable storage pods to improve warehouse efficiency by reducing human travel and enabling faster order fulfillment (Azadeh et al., 2019; Benavides-Robles et al., 2024; Filipović & Rogić, 2025). However, the performance of RMFS depends heavily on key operational decisions, including SKU-to-pod assignment, order picking strategies, and replenishment policies. Since storage pods are limited and constrained by weight and dimensional capacities, SKUs must be strategically allocated to minimize pod visits, robot travel distance, and overall energy consumption (Kim et al., 2020). These decisions are further complicated by e-commerce demand that shifts with seasonality, promotions, and changing customer preferences (Lamballais et al., 2022).

Research on SKU-to-pod assignment has evolved through several categories of approach. Early methods rely on turnover-based policies, using rules such as random, class-based, and full-turnover that place frequently demanded items near picking stations and reduce retrieval time (Merschformann et al., 2019; Silva et al., 2020). These rules, however, evaluate products based on individual turnover and ignore product co-occurrence patterns in customer orders (Mirzaei et al., 2022). To capture this, affinity-based placement groups SKUs by co-occurrence in historical orders. Mirzaei et al. (2022) proposed an integrated cluster-allocation policy combining turnover and affinity, showing reduced retrieval time compared to turnover-only rules. Yuan et al. (2021) further considered product co-occurrence by co-locating frequently ordered-together products to reduce pod visits, while Zhuang et al. (2024) jointly optimized affinity-based assignment with downstream order and rack sequencing. In parallel, dispersed and scattered storage strategies have been proposed to distribute correlated inventory across pods and zones, thereby improving retrieval efficiency (Lamballais et al., 2022; Ma et al., 2023; Mirzaei et al., 2022). More recent research has extended this perspective by applying clustering and classification techniques to group SKUs by demand characteristics: order-correlation and time-series clustering capture dynamic demand for storage assignment (Kalkha et al., 2024; Keung et al., 2021; Kiabakht & Sajadieh, 2026; Li et al., 2020), while hybrid approaches combine ABC classification with dimensional clustering for pod assignment (Chou et al., 2023; Pratiwi et al., 2025). At the formulation level, the assignment problem is also commonly formulated as a bin-packing or container-loading problem, where items must be placed into capacity-limited containers while satisfying orientation and non-overlap constraints (Rampone et al., 2023; Tsao et al., 2024).

A more recent and less-developed line integrates replenishment into storage planning. Most SKU assignment studies optimize only order-picking effort and assume that inventory in the picking area is always sufficient, thereby overlooking replenishment decisions (Kiabakht & Sajadieh, 2026; Wang et al., 2025). When replenishment is considered, existing studies tend to model it from different operational perspectives: as a fixed-sequence rule supported by pod-selection heuristics such as the emptiest-pod policy Merschformann et al. (2019), as pod-level and warehouse-level inventory triggers Chou et al. (2023), as an SKU-level continuous review policy embedded within the storage model Wang et al. (2025), or as a reactive recourse action activated only after shortages occur (Kiabakht & Sajadieh, 2026). More recently, integrated approaches have begun to couple the two decisions directly, for example Teck et al. (2025) applied deep reinforcement learning to address real-time storage assignment and replenishment. A key insight from this literature is that SKU dispersion across multiple pods can improve picking efficiency, but may also increase replenishment effort. Therefore, storage assignment and replenishment should be evaluated jointly (Chou et al., 2023). However, these models are generally assuming stable demand conditions, making them less responsive to the volatility of modern e-commerce operations (Filipović & Rogić, 2025; Lamballais et al., 2022; Sulthana & Vasantha, 2019; Utama & Salsabilah, 2025).

Two key research gaps remain. First, SKU-to-pod assignment and replenishment are commonly treated as separate decision problems which limit the use of demand-pattern information across two processes. Second, although clustering is widely used as the basis for assignment, replenishment is rarely triggered at the cluster level, and demand trends are rarely incorporated to adjust storage and replenishment decisions over time. To address these gaps, this study proposes a data-driven and trend-driven framework that advances prior work in three main ways. First, it jointly integrates SKU-to-pod assignment and replenishment decisions rather than optimizing them separately.

Spektrum Industri
Vol. 24, No. 2, 2026, pp. 366–378

Second, it periodically re-clusters SKUs based on demand generated during operation, allowing both cluster assignments and replenishment thresholds to adapt to the latest observed demand patterns rather than relying on explicit demand forecasting. Third, it applies a cluster-specific dual-trigger replenishment policy, extended from Chou et al. (2023), so that replenishment decisions are aligned with the demand behavior of each SKU cluster.

The contribution of this research is an integrated, trend-driven decision framework for RMFS that combines SKU clustering based on demand and affinity, SKU-to-pod assignment, and cluster-differentiated replenishment within a single agent-based simulation environment. The scientific contributions of this study are threefold. First, it develops an SKU clustering method that incorporates demand volume, demand variability, order frequency, and product affinity. Second, it proposes a cluster-level dual-trigger replenishment policy, in which replenishment thresholds are defined for each cluster and periodically updated as SKUs are re-clustered during operation. This addresses the limited consideration of cluster-based replenishment in prior RMFS studies. Third, it develops an agent-based simulation framework that jointly evaluates SKU-to-pod assignment and replenishment under stochastic and dynamically changing demand, using order throughput, replenishment-to-pick ratio, and pod utilization as performance indicators. From a practical perspective, the proposed framework serves as an adaptive decision-support tool for warehouse managers. It improves order throughput and replenishment efficiency compared with a static baseline while maintaining pod utilization, supporting more responsive and less labor-intensive fulfillment operations in volatile e-commerce environments.

2. Method

This study integrates machine learning, implemented in Python, with an agent-based simulation built in NetLogo (Pasupuleti et al., 2024; Rizqi et al., 2025). Python performs SKU clustering and assignment, while NetLogo represents the interactions between robots, picking stations, and replenishment stations. NetLogo and Python are connected so that NetLogo can call Python routines and receive their outputs during the simulation run. Research framework is shown in Fig. 1.

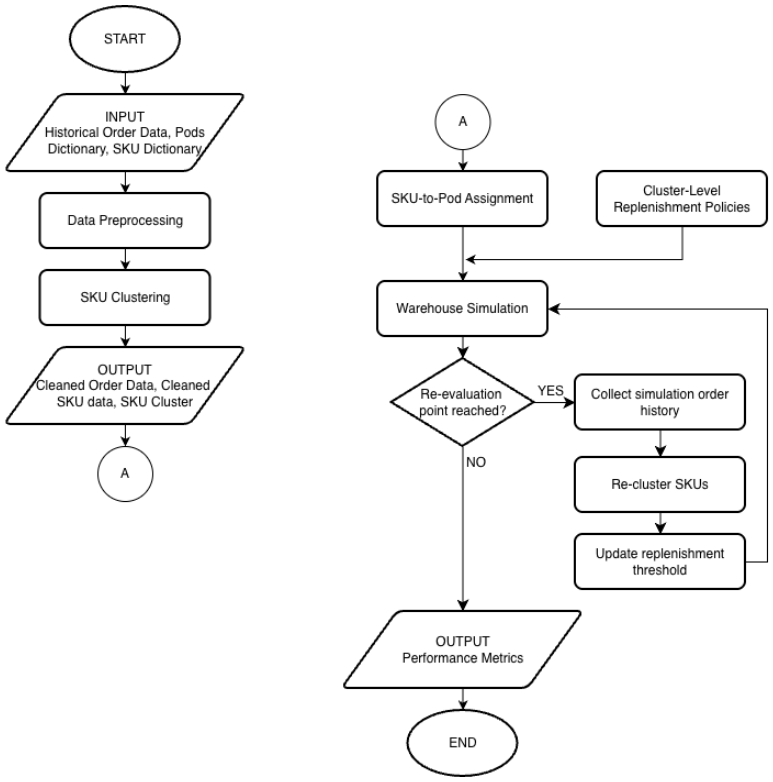


Fig. 1. Research Framework

The process starts with an initialization phase based on historical order data. After preprocessing, SKUs are clustered according to their demand characteristics, assigned to pods, and initialized with cluster-based replenishment thresholds. The simulation is then executed using the initial configuration. At a predefined re-evaluation point, the demand generated during the simulation is sent back to Python, where SKUs are re-clustered and replenishment threshold are updated before being returned to NetLogo. During the feedback loop, only SKU cluster and assignments and replenishment thresholds are updated, while the initial SKU-to-pod assignment remains fixed. In this study, re-evaluation is conducted once at hour 12 of the 24-hour simulation horizon, allowing sufficient operation demand to accumulate before re-clustering. This enables the replenishment policy to reflect the latest observed demand patterns rather than remain static (Abolghasem et al., 2025; Guven et al., 2026).

2.1. Data Collection and Preprocessing

This study uses a dataset from an e-commerce warehouse covering 21 days of historical orders. The original dataset contains 321.315 orders and 7.443 unique SKUs. A preprocessing step removed SKU records that did not match their corresponding order information, leaving a cleaned dataset of 308.944 orders and 7.440 SKUs. Using historical customer orders allows the study to identify meaningful demand patterns and product affinities, both of which are essential for characterizing SKU demand behavior and supporting the clustering and storage-assignment steps (Jain et al., 2024; Kalkha et al., 2024; Tian et al., 2024).

2.2. SKU Clustering

SKUs are clustered based on demand characteristics and product affinity. Clustering products using demand features and product affinity has been shown to improve warehousing efficiency by optimizing storage allocation and reducing retrieval times (Kalkha et al., 2024; Mirzaei et al., 2022). In this research, demand characteristics are represented using three features: demand volume, coefficient of variation, and order frequency (Mejía & Aguilar, 2024). Demand volume reflects the average quantity demanded for a product over a given period. The coefficient of variation (CV) calculated using Eq. (1), is the ratio of the standard deviation to the mean value and captures demand variability. Order frequency captures how often a SKU appears in customer orders.

$$CV = \frac{\sigma}{\mu} \tag{1}$$

In addition to demand characteristics, product affinity captures the relationship between products that are frequently ordered together. The affinity value between two products i and j is calculated using Eq. (2), measures the co-occurrence within the same customer orders (Mirzaei et al., 2022). For clustering, each SKU's maximum affinity, which represent its strongest single co-occurrence relationship, is used.

$$\rho_{ij} = \frac{\sum_{o \in O} r_{oi} \times r_{oj}}{|O|}, \quad i, j \in P, i \neq j \tag{2}$$

where $r_{oi} = 1$ if order $o \in O$ contains a request for product $i \in P$, and $r_{oi} = 0$ otherwise.

Because the four features differ widely in scale, all are rescaled to [0, 1] using min-max normalization before clustering. It prevents any single feature from dominating the distance computation. Clustering uses the K-means algorithm, chosen for its efficiency on large SKU sets and its demonstrated performance in classifying orders and improving operational efficiency in RMFS settings (Keung et al., 2021; Kim et al., 2020). The number of clusters k was determined using the elbow method on the within-cluster sum-of-squares (WCSS) together with the average silhouette coefficient. Both criteria support the selection of k = 5 for the analysis, as five clusters provide sufficient granularity to define meaningful demand tiers for the storage and replenishment decisions developed in this study. The detailed values and elbow plot are presented in Section 3.1.

Spektrum Industri
Vol. 24, No. 2, 2026, pp. 366–378

2.3. SKU-to-pod Assignment

SKU-to-pod assignment in RMFS resembles the bin-packing and container-loading problems, in which items are placed into capacity-limited containers under orientation and non-overlap constraints; the problem is NP-hard, so an exact optimum is generally computationally intractable (Kim et al., 2020; Tsao et al., 2024). In this study, a single pod type with five slot types is used, and each slot holds a single SKU type. Each SKU is characterized by volume, weight, and inventory quantity (Merschformann et al., 2019; Zhuang et al., 2024). Following common practice for three-dimensional item placement, each SKU box may be positioned in six orientations within a slot (Nguyen et al., 2021; Tsao et al., 2024). The assignment follows these rules:

- Each compartment type in a pod store only one specific SKU.
- Each compartment type has a fixed dimension, and a SKU may be placed in six orientations.
- Each product must be aligned with the compartment walls.
- Each product must fit entirely inside the compartment, with no overlap.
- The quantity of each product stored is determined by the reorder point in Eq. (3) and then converted into a corresponding number of boxes. where μ is mean demand, LT is lead time (set to 1), z is the service-level factor (90%), and σ is the standard deviation of demand.

$$ROP = \mu \times LT + z\sigma \times \sqrt{LT} \tag{3}$$

Product dispersion is applied to high-moving clusters by distributing frequently demanded SKUs across multiple pods. This strategy reduces stockout risk and improves throughput by increasing the availability of high-demand products in the system (Lamballais et al., 2022). As a result, a single pod may contain products from more than one cluster. During assignment, priority is given to items in high-moving clusters and items that require less storage space or can be placed efficiently within the available slot configuration. SKUs with box dimensions that exceed the available pod slot capacity are considered infeasible and are handled separately. The overall SKU-to-pod assignment procedure is summarized in Fig. 2 and consists of two linked stages. In the first stage, a feasible configuration is determined for each SKU by testing all the six possible box orientations and calculating the maximum number of boxes. In the second stage, SKUs are sorted by cluster priority and by the smallest compatible slot type and are then assigned to pods. For each SKU, the procedure will search the opened pods first to identify a compatible available slot. A new pod is opened only when no feasible slot is available in the existing opened pods. SKUs with initial inventory levels that exceed the capacity of the assigned slot are dispersed across additional pods, while each pod is constrained by a total weight capacity of 600 kg. This process is repeated until all feasible SKUs have been assigned, resulting in the final SKU-to-pod assignment.

Due to the limitation of the simulation environment, stratified sampling is applied to select SKUs that included in the simulation, following standard sampling practice (Cochran & Cochran, 1977). The sampling process is conducted in two stages: (1) by demand cluster, (2) by impact score quartile within each cluster. Each feasible SKU is assigned an impact score in the range of [0.1]. It calculated as the equal-weighted average of normalized demand and affinity characteristics using min-max normalization, as shown in Eq. (4).

$$x_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{4}$$

Within each cluster, SKUs are split into four impact quartiles and sampled at different rates, 40% for Q4, 25% for Q3, 10% for Q2, and 5% for Q1. This disproportionate allocation intentionally oversamples higher-impact SKUs because they contribute more strongly to system behavior and must be represented more accurately, consistent with SKU-classification approaches that prioritize items by operational importance (Van et al., 2012). Disproportionate stratified sampling is also a standard method for maintaining the influence of important strata within a sample (Cochran & Cochran, 1977).

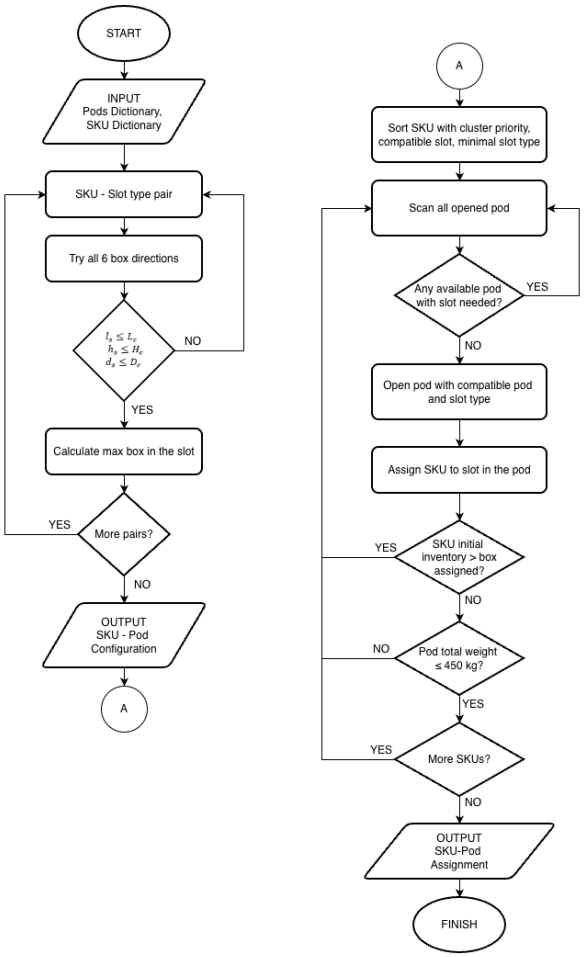


Fig. 2. SKU-to-Pod Assignment Flowchart

2.4. Cluster-Based Replenishment Policy

Replenishment directly affects system throughput, stockout risk, and operating cost in RMFS (Lamballais et al., 2022; Merschformann et al., 2019). This study adopts two-trigger approach by Chou et al. (2023), which consider both warehouse-level and pod-level inventory conditions. The pod-level threshold is set at 60% for all pods to keep pods adequately stocked and sustain throughput. While the warehouse-level threshold is differentiated by cluster so that replenishment responsiveness reflects each cluster's demand behavior.

A single warehouse-level threshold applied to every SKU would ignore differences in demand level and variability. It may either over-serve low-demand SKUs or insufficiently protect high-demand SKUs from stockouts. To avoid this, the warehouse-level threshold is assigned by cluster, following the class-based inventory principle of C. Wang & Ning (2026), who show that assigning higher service levels to more important and demand-stable classes can improve the overall fill rate with lower safety-stock cost. Since the clusters formed capture differences in demand level and variability, they provide a suitable basis for differentiating replenishment service levels. Therefore, the warehouse-level thresholds for Clusters 0 to 4 are set at 70%, 62%, 75%, 65%, and 80%, respectively, with the high-moving cluster assigned the most sensitive threshold to reduce stockout risk. In proposed scenario, these cluster-based warehouse thresholds are recomputed whenever SKU clusters are re-evaluated. This allows the replenishment policy to follow the latest observed demand patterns rather than remaining fixed throughout the simulation.

Spektrum Industri
Vol. 24, No. 2, 2026, pp. 366–378

2.5. Simulation Design

The simulation is implemented in NetLogo and Python and reproduces the interaction between robots, picking stations, and replenishment stations within the warehouse layout illustrated in Fig. 3. The blue blocks represent the storage pods where SKUs are stored, while the red figures indicate the mobile robots operating within the warehouse. On the left side of the layout, three picking stations are located where human operators retrieve items from pods delivered by robots. On the right side, two replenishment stations are positioned for restocking inventory into the pods.

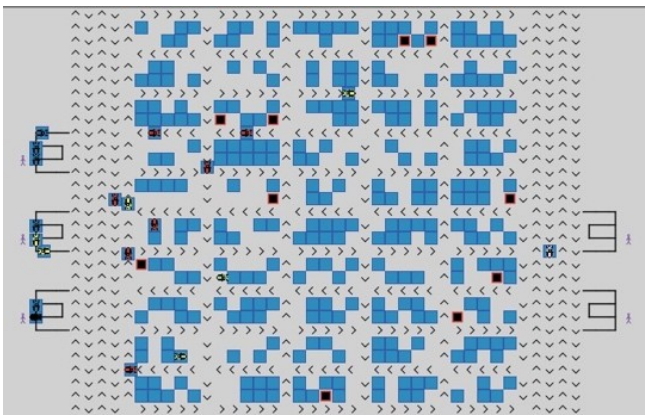


Fig. 3. Simulation Layout

Order arrivals are generated from a Poisson distribution at a rate of two orders per minute, with an initial backlog of 20 orders. The number of SKUs per order follows a geometric distribution with parameter $p = 0.3$. The order proportion for each cluster is modeled with a Markov-modulated Poisson process (MMPP) to reproduce real-world demand dynamics, capturing the alternation between high-demand and low-demand periods for each cluster over time (Lamballais et al., 2022). The full set of simulation parameters is listed in Table 1. Two scenarios are compared: a baseline and the proposed scenario. In the baseline, replenishment policies are fixed to predefined SKU clusters and remain unchanged throughout the run, representing a static policy. In the proposed scenario, SKU clusters are periodically re-evaluated from the demand generated during the run, allowing each cluster's replenishment policy to be updated to the latest demand pattern.

Table 1. Simulation Parameters

Parameter	Value	Unit
Run length	24	Hours
Replication	10	Replications
Inventory area	500	Locations
Inventory capacity	420	Pods
Empty storage area	10	Locations
Pod batch	2 x 5	Blocks
Aisles	12 verticals; 4 horizontals	Aisles
Stations	3 picking; 2 replenishments	Stations
Initial order	20	Orders
Order proportion	5 class created with Markov-modulated Poisson Process	
Order arrival rate	2 orders/min	Poisson dist.
Number of SKUs per order	3.33 SKUs/order	Geometric dist.
Pod capacity	34	Slots
Pod slot type	5	Types
Number of SKUs	1449	SKUs
Picking time (per SKU)	1.5	Second
Replenishment time	15	Second
Robot speed	1.5	Meter/Second
Time to lift and store pod	1.5	Second
Number of AGVs	20	Robots

3. Results and Discussion

3.1. SKU Clustering

The number of clusters was determined using the elbow method based on the within-cluster sum of squares (WCSS), complemented by the average silhouette coefficient. The WCSS decreases sharply from $k = 2$ to $k = 5$, after which the reduction becomes relatively small. The silhouette score was also used as a complementary criterion, with k = 5 showing a relatively high silhouette score of 0.538. Therefore, five clusters were selected for the analysis, as they provide sufficient granularity to define meaningful demand tiers this study. The resulting clusters and their characteristics are summarized in Table 2. The five clusters differ clearly in demand level, variability, and order frequency. Cluster 4 represents high-moving SKUs with the highest mean demand and order frequency and relatively stable demand, indicating products that are ordered often and co-picked with other items. Cluster 0 contains medium-to-high-demand SKUs, while Cluster 2 holds moderate-demand items with higher variability. Clusters 1 and 3 consist of low-demand SKUs that appear infrequently, with Cluster 3 showing the most intermittent, highly variable demand. The affinity number is near zero across all clusters, indicating that the co-occurrence of any two specific SKUs is rare given the large catalog and the 21-day observation window. Nevertheless, affinity is retained as a clustering feature because even weak co-occurrence patterns can help distinguish demand behavior across a large catalog and support the dispersion of high-moving SKUs across multiple pods.

Table 2. Clustering Result

Cluster	Number of SKUs	Mean demand	Mean CV	Mean frequency	Mean max affinity
3	1002	0.182	4.472	0.048	0.000
1	2497	0.655	2.700	0.164	0.000
2	1806	2.116	1.574	0.418	0.000
0	1784	22.608	0.978	0.818	0.000
4	351	176.029	0.637	0.985	0.001

3.2. SKU-to-Pod Assignment

Following the SKU-to-pod assignment procedure to the clustered SKUs, 1.628 SKUs were identified as infeasible because their box dimensions exceeded the available pod slot sizes. These SKUs were therefore handled separately. After stratified sampling based on the impact score, the final simulation sample consisted of 1.449 SKUs assigned across 158 active pods. In this study, product dispersion is applied to cluster 4 because it represents a high-moving items. Product dispersion is used to distribute frequently demanded products across multiple pods to reduce the risk of stockouts and improve throughput performance (Lamballais et al., 2022). To implement this strategy, SKUs in cluster 4 are allocated to more than one pod. As a result, each pod may contain products from two or more clusters. Product assignment is prioritized for items that require minimal storage space or fit efficiently within the available pod slot configuration.

3.3. Simulation Result

The simulation in this study was conducted using NetLogo to capture the interactions between robots, picking stations, and the proposed operational policies. The pod inventory threshold is set at 60% to maintain high throughput efficiency and ensure that pods remain sufficiently stocked during operations (Chou et al., 2023). Meanwhile, the warehouse inventory level is determined based on cluster classification. The warehouse inventory thresholds for clusters 0 to 4 are set at 70%, 62%, 75%, 65%, and 80%, respectively. Among these clusters, cluster 4, which represents high-moving SKUs, is assigned the highest warehouse inventory level to reduce the risk of stockouts and maintain operational efficiency. Each scenario was simulated for ten independent replications. Using ten replications provides a sufficient sample to estimate variability and to test whether the differences between scenarios are statistically reliable rather than the result of random fluctuation.

Statistical tests are used to evaluate the significance of the differences between the baseline and proposed scenario results. The normality of the paired differences is first assessed using the Shapiro–Wilk test. If the paired differences are normally distributed, a paired-samples t-test is applied. If the normality assumption is rejected, the nonparametric Wilcoxon signed-rank test is used instead. All tests are conducted at a significance level of $\alpha = 0.05$. Table 3 presents the descriptive statistics and paired significance tests for each performance indicators, while Fig. 4 illustrates the scenario comparisons for each performance indicators.

Table 3. Performance comparison between baseline and proposed scenarios

Metric	Baseline (Mean ± stdev)	Proposed (Mean ± stdev)	Test	p value	Significance
Order throughput (%)	74.75 ± 3.28	93.44 ± 2.25	Paired t	< 0.001	Yes
Replenishment-to-picking ratio (%)	38.98 ± 1.76	34.56 ± 0.95	Wilcoxon	0.002	Yes
Pod utilization (SKUs/visit)	6.59 ± 0.12	6.59 ± 0.10	Paired t	0.920	No

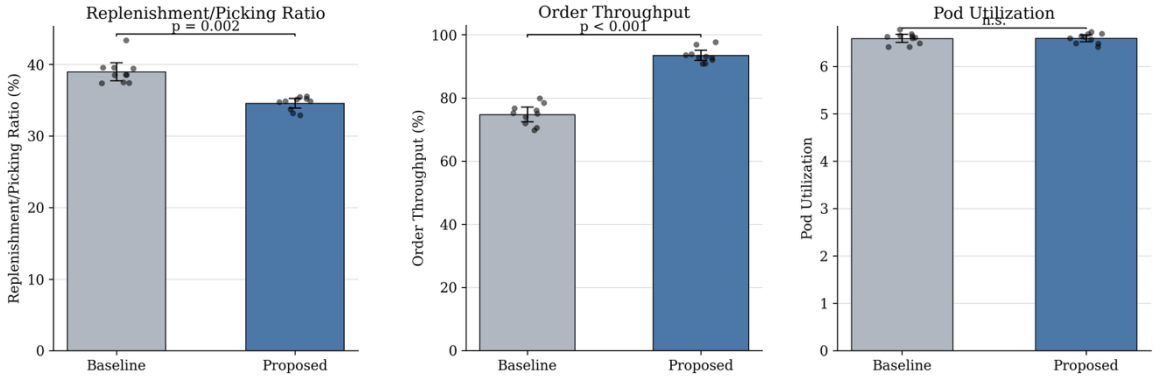


Fig. 4. Performance comparison between baseline and proposed scenarios

The proposed scenario achieved a mean order throughput of 93.44%, compared with 74.75% in the baseline scenario. The results showed that the improvement of 18.69 percentage points, equivalent to a 25% relative increase, was statistically significant ($p < 0.001$). The replenishment-to-picking ratio decreased from 38.98% in the baseline scenario to 34.56% in the proposed scenario. The reduction of 4.42 percentage points, corresponding to an 11% relative decrease, was statistically significant ($p = 0.002$), with improvements observed in all ten replications. While pod utilization metric showed no meaningful difference between the two scenarios, with both achieving a mean of 6.59 SKUs per pod visit (paired t-test, $p = 0.920$).

3.4. Discussion

This study demonstrates that integrating replenishment and SKU-to-pod assignment through clustering can improve RMFS performance under stochastic and dynamically changing demand. The increase in throughput and the reduction in the replenishment-to-picking ratio can be explained by the demand responsive structure of the proposed policy. By applying cluster-based replenishment thresholds, high-demand clusters are maintained at higher inventory levels. It can reduce the likelihood of stockouts during order fulfillment and help pods remain productive. With fewer stockouts, pods carrying high-demand SKUs remain available for picking rather than waiting to be replenished, thereby sustaining a higher order-completion rate. In parallel, dispersing high moving SKUs across multiple pods distributes picking demand more evenly and prevents bottlenecks and congestion.

The lower replenishment-to-picking ratio reflects the same mechanism for the replenishment. Cluster-based threshold triggered replenishment when its operationally needed rather than applied uniformly across all SKUs. This reduces unnecessary replenishment trips that would otherwise

compete with picking activities for robot and station capacity. These findings are consistent with Chou et al. (2023) who showed that a dual-trigger replenishment policy can improve throughput, and with Lamballais et al. (2022), who found that inventory dispersion across pods improves retrieval efficiency. This study extends these ideas by differentiating replenishment thresholds based on the cluster and updating the cluster's members as demand patterns change. Unlike the current assignment studies that optimize storage placement under relatively static demand conditions, the proposed framework integrates demand-based clustering with replenishment. The absence of a significant change in pod utilization is expected because this metric is primarily determined by the SKU-to-pod assignment structure, which remains the same across both scenarios. Because the metric measures the number of SKUs picked per pod visit, it is less directly affected by the replenishment policy that differentiates them.

For the affinity feature used in the cluster, as shown in Table 2, the mean maximum affinity is highest for the high-moving cluster. It is consistent with the expectation that frequently ordered items are also more likely to be co-ordered. To assess the contribution of affinity, clustering was repeated using only demand features. The resulting cluster structure differed substantially, indicating that affinity affects how SKUs are grouped rather than merely acting as noise. Because all features are standardized before clustering, the algorithm responds to relative differences among SKUs rather than absolute magnitudes. Therefore, even features with small absolute values can still contribute to meaningful cluster separation. From a theoretical perspective, this study contributes to the RMFS literature by proposing a data-driven and integrated framework that considers SKU-to-pod assignment and replenishment policy under stochastic demand. The proposed frameworks link both decisions through machine-learning-based demand clustering and periodically re-clusters SKUs based on demand observed during system operation. This improves the responsiveness of storage and replenishment decisions to changing demand patterns. By incorporating demand and affinity feature into the clustering process, the framework provides a more realistic representation of highly dynamic e-commerce demand than turnover-based approaches. It also connects SKU grouping directly to replenishment responsiveness, rather than focusing only on storage efficiency.

From a managerial perspective, the proposed framework provides an adaptive decision-support approach for automated warehouse. The lower replenishment-to-picking ratio indicates better coordination between replenishment and order fulfillment, which may reduce replenishment-related congestion and operational cost. In assignment side, distributing high-moving SKUs across multiple pods helps balance the picking workload and prevents individual pods from becoming bottlenecks. Periodic re-clustering also reduces the need for planners to manually adjust storage and replenishment settings. Together, these results suggest that the proposed framework can help warehouses maintain service levels and reduce stockout risk in volatile demand environments.

4. Conclusion

This study set out to address two gaps in RMFS decision-making, which are the common separation of SKU-to-pod assignment and replenishment and the limited use of demand trend to adapt these decisions over time. To address them, this study developed a data-driven, trend-driven framework that integrates SKU clustering, cluster-guided SKU-to-pod assignment, and cluster-differentiated replenishment within a single agent-based simulation, in which the clusters and replenishment thresholds are re-evaluated from the demand generated during operation. The proposed framework was evaluated over ten replications against a static baseline. The proposed framework increased order throughput from 74.75% to 93.44% and reduced the replenishment-to-picking ratio from 38.98% to 34.56%, with both improvements statistically significant, while pod utilization remained unchanged. These results indicate that adapting cluster-based replenishment to observed demand improves throughput and replenishment efficiency without reducing the effectiveness of each pod visit, directly addressing the two identified research gaps.

Spektrum Industri
Vol. 24, No. 2, 2026, pp. 366–378

Several limitations should be noted. First, the evaluation is based on a single e-commerce dataset and one warehouse layout. Second, due to the constraints of the simulation environment, only 1.449 SKUs were included through stratified sampling. Third, the simulation covers a relatively short horizon of 24 hours, with re-evaluation performed only once at the midpoint. This duration may be insufficient for demand trends to fully develop or for the periodic re-clustering mechanism to show its complete effect. A longer simulation horizon would allow demand pattens and would allow demand patterns and steady-state system behavior to become apparent more clearly. From this limitations, future research could: (1) run multi-day simulations with more frequent re-evaluation to capture longer-term demand behaviour; (2) validate the framework across multiple warehouses and larger-scale layouts; (3) incorporate dynamic SKU relocation, reassigning SKUs to pods as clusters evolve, and (4) test the framework against diverse demand patterns to assess its robustness.

Author Contribution: Salsabila Herwening Mileniadewi: conceptualization, methodology, software, formal analysis, investigation, writing, visualization, writing–original draft. Shuo-Yan Chou: conceptualization, methodology, supervision, writing–review & editing, resources. Sinta Rahmawidya Sulistyo: supervison, validation, writing–review & editing. Anna Maria Sri Asih: supervison, validation, writing–review & editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding

Acknowledgment: The authors would like to express their sincere gratitude to all individuals and institutions who contributed to the completion of this research. The authors also appreciate the support and assistance provided throughout the research process.

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

Declaration of AI-Tool Usage: The authors declare that AI-based tools, if used, were utilized solely to assist with language refinement and grammatical improvement. All research ideas, methodological decisions, data analysis, interpretations, and conclusions remain the responsibility of the authors. The authors have carefully reviewed and verified the manuscript and take full responsibility for its content.

References

Abolghasem, M., Sadeghi, K., Ghazaie, S. H., & Peng. W. (2025). Enhancing nuclear cogeneration efficiency using the low-grade waste heat recovery from nuclear hydrogen production system. *Nuclear Engineering and Design*. https://doi.org/10.1016/j.nucengdes.2025.114166

Azadeh, K., De Koster, R., & Roy, D. (2019). Robotized and Automated Warehouse Systems: Review and Recent Developments. *Transportation Science*, *53*(4), 917–945. https://doi.org/10.1287/trsc.2018.0873

Benavides-Robles, M. T., Valencia-Rivera, G. H., Cruz-Duarte, J. M., Amaya, I., & Ortiz-Bayliss, J. C. (2024). Robotic Mobile Fulfillment System: A Systematic Review. *IEEE Access*, *12*, 16767–16782. https://doi.org/10.1109/ACCESS.2024.3359434

Boysen, N., de Koster, R., & Weidinger, F. (2019). Warehousing in the e-commerce era: A survey. *European Journal of Operational Research*, *277*(2), 396–411. https://doi.org/10.1016/j.ejor.2018.08.023

Chou, S.-Y., Hendrawan, E., Dewabharata, A., & Zulvia, F. E. (2023). *Replenishment Policy and SKU Classification to Pod Assignment Design for Robotic Mobile Fulfillment System Performances*. https://doi.org/10.3233/ATDE230633

Cochran, W. G., & Cochran, W. G. (1977). *Sampling techniques*. academia.edu. https://www.academia.edu/download/53436713/Cochran_1977_Sampling_Techniques__Third_Edition.pdf

Filipovic, M., & Rogic, K. (2025). Robotic Mobile Fulfilment System: A Literature Review. *Transportation Research Procedia*, *91*, 465–472. https://doi.org/10.1016/j.trpro.2025.10.060

Guven, D., Kayalica, M. O., & Bozdag, C. E. (2026). Assessing policy interventions for electric vehicle adoption through an agent-based environmental framework. *Applied Energy*. https://doi.org/10.1016/j.apenergy.2025.127338

Jain, N., Vamshidhar, M., Kulshrestha, S., & Gupta, N. (2024). Integrating information science into financial management: A comprehensive review of emerging trends and practices. In *Library Progress International*. https://doi.org/10.48165/bapas.2024.44.2.1

Kalkha, H., Khiat, A., Bahnasse, A., & Ouajji, H. (2024). Enhancing Warehouse Efficiency With Time Series Clustering: A Hybrid Storage Location Assignment Strategy. *IEEE Access*, *12*, 52110–52126. https://doi.org/10.1109/ACCESS.2024.3386887

Keung, K. L., Lee, C. K. M., & Ji, P. (2021). Data-driven order correlation pattern and storage location assignment in robotic mobile fulfillment and process automation system. *Advanced Engineering Informatics*, *50*, 101369. https://doi.org/10.1016/j.aei.2021.101369

Kiabakht, H., & Sajadieh, M. S. (2026). Integrated approach for operations in robotic mobile fulfillment centers under order uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, *208*, 104683. https://doi.org/10.1016/j.tre.2026.104683

Kim, H.-J., Pais, C., & Shen, Z.-J. M. (2020). Item Assignment Problem in a Robotic Mobile Fulfillment System. *IEEE Transactions on Automation Science and Engineering*, *17*(4), 1854–1867. https://doi.org/10.1109/TASE.2020.2979897

Lamballais, T., Merschformann, M., Roy, D., de Koster, M. B. M., Azadeh, K., & Suhl, L. (2022). Dynamic policies for resource reallocation in a robotic mobile fulfillment system with time-varying demand. *European Journal of Operational Research*, *300*(3), 937–952. https://doi.org/10.1016/j.ejor.2021.09.001

Li, X., Hua, G., Huang, A., Sheu, J.-B., Cheng, T. C. E., & Huang, F. (2020). Storage assignment policy with awareness of energy consumption in the Kiva mobile fulfilment system. *Transportation Research Part E: Logistics and Transportation Review*, *144*, 102158. https://doi.org/10.1016/j.tre.2020.102158

Ma, Z., Wu, G., Ji, B., Wang, L., Luo, Q., & Chen, X. (2023). A Novel Scattered Storage Policy Considering Commodity Classification and Correlation in Robotic Mobile Fulfillment Systems. *IEEE Transactions on Automation Science and Engineering*, *20*(2), 1020–1033. https://doi.org/10.1109/TASE.2022.3178934

Mejía, S., & Aguilar, J. (2024). A demand forecasting system of product categories defined by their time series using a hybrid approach of ensemble learning with feature engineering. *Computing*, *106*(12), 3945–3965. https://doi.org/10.1007/s00607-024-01320-y

Merschformann, M., Lamballais, T., de Koster, M. B. M., & Suhl, L. (2019). Decision rules for robotic mobile fulfillment systems. *Operations Research Perspectives*, *6*, 100128. https://doi.org/10.1016/j.orp.2019.100128

Mirzaei, M., Zaerpour, N., & de Koster, R. B. M. (2022). How to benefit from order data: correlated dispersed storage assignment in robotic warehouses. *International Journal of Production Research*, *60*(2), 549–568. https://doi.org/10.1080/00207543.2021.1971787

Pasupuleti, V., Thuraka, B., Kodete, C. S., & Malisetty, S. (2024). Enhancing Supply Chain Agility and Sustainability through Machine Learning: Optimization Techniques for Logistics and Inventory Management. *Logistics*, *8*(3), 73. https://doi.org/10.3390/logistics8030073

Pratiwi, D. T., Chou, S.-Y., Suparno, & Kurniati, N. (2025). *Combination of SKU in POD Assignment in Robotic Mobile Fulfillment Systems* (pp. 185–195). https://doi.org/10.1007/978-981-97-6492-1_15

Rampone, F., Lecca, F., Giolito, P., & Romano, M. (2023). Blockchain in the agrifood sector: From storytelling to traceability fact-checking up to new economic models. *Economia Agro-Alimentare*, *25*(2), 97–114. https://doi.org/10.3280/ecag2023oa14958

Rizqi, Z. U., Chou, S.-Y., & Oscar, A. D. (2025). Towards energy-efficient Robotic Mobile Fulfillment System: Hybrid agent-based simulation with DEA-based surrogate machine learning. *Applied Soft Computing*, *176*, 113141. https://doi.org/10.1016/j.asoc.2025.113141

Silva, A., Coelho, L. C., Darvish, M., & Renaud, J. (2020). Integrating storage location and order picking problems in warehouse planning. *Transportation Research Part E: Logistics and Transportation Review*, *140*, 102003. https://doi.org/10.1016/j.tre.2020.102003

Sulthana, A. N., & Vasantha, S. (2019). Influence of electronic word of mouth eWOM on purchase intention. In *International Journal of Scientific.*

Teck, S., Pham, T. S., Rousseau, L.-M., & Vansteenwegen, P. (2025). Deep reinforcement learning for the real-time inventory rack storage assignment and replenishment problem. *European Journal of Operational Research*, *327*(2), 606–622. https://doi.org/10.1016/j.ejor.2025.05.008

Tian, M., Zhang, Q., Tian, D., Jin, L., Li, J., & Xiao, F. (2024). Pre-stability control for in-wheel-motor-driven electric vehicles with dynamic state prediction. *IEEE Transactions on Intelligent Vehicles*. https://doi.org/10.1109/TIV.2024.3368207

Tsao, Y.-C., Tai, J.-Y., Vu, T.-L., & Chen, T.-H. (2024). Multiple bin-size bin packing problem considering incompatible product categories. *Expert Systems with Applications*, *247*, 123340. https://doi.org/10.1016/j.eswa.2024.123340

Utama, D. M., & Salsabilah, A. L. (2025). Optimizing Green Vehicle Routing Problem for Halal and Non-Halal Products using Salp Swarm Algorithm. *Spektrum Industri*. https://doi.org/10.12928/si.v23i2.413

Van, K. T. J., Akkerman, R., & Pieter van Donk, D. (2012). SKU classification: a literature review and conceptual framework. *International Journal of Operations & Production Management*, *32*(7), 850–876. https://doi.org/10.1108/01443571211250112

Wang, M., Liu, X., Wang, L., Bian, Y., Fan, K., & Zhang, R.-Q. (2025). A data-driven optimization model for the scattered storage assignment with replenishment. *Computers & Industrial Engineering*, *200*, 110766. https://doi.org/10.1016/j.cie.2024.110766

Yuan, R., Li, J., Wang, W., Dou, J., & Pan, L. (2021). Storage Assignment Optimization in Robotic Mobile Fulfillment Systems. *Complexity*, *2021*(1). https://doi.org/10.1155/2021/4679739

Zhuang, Y., Zhou, Y., Hassini, E., Yuan, Y., & Hu, X. (2024). Improving order picking efficiency through storage assignment optimization in robotic mobile fulfillment systems. *European Journal of Operational Research*, *316*(2), 718–732. https://doi.org/10.1016/j.ejor.2024.02.025