Leveraging Transdisciplinary Engineering in a Changing and Connected World 393
P. Koomsap et al. (Eds.)

doi:10.3233/ATDE230633

Replenishment Policy and SKU Classification to Pod Assignment Design for Robotic Mobile Fulfillment System Performances

Shuo-Yan CHOU, Edwin HENDRAWAN, Anindhita DEWABHARATA[1], Ferani E. ZULVIA
[a] *Department of Industrial Management, National Taiwan University of Science and Technology, Taipei, Taiwan, ROC.*

Abstract. Robotic Mobile Fulfillment System (RMFS) is a well-known automated parts-to-picker system that is highly suitable in a fast-moving warehouse for handling critical challenges in the e-commerce industry. Implementing this system in the e-commerce industry has been shown to boost the throughput compared with the traditional picker-to-parts picking system. Nevertheless, there are still several ways to increase the efficiency of the warehouse. Therefore, this study proposes product-to-Pod or SKU-to-Pod assignment and replenishment policies that can increase warehouse efficiency using a simulation approach. There are three SKU-to-pod assignment policies evaluated in this study. They are Random Assignments, One Pod One Class, and Mix Class One Pod assignments. In addition, four replenishment policies, including the Emptiest Pod, Pod Inventory Level, Warehouse Inventory-SKU in Pod, and Stockout Probability, are also simulated. The simulation results show that the Mix Class One Pod assignments combined with Warehouse Inventory-SKU in Pod is the best policy. The SKU-to-pod policy can improve pod utilization by increasing pick units in each visit. Pod with more SKU types is likely to fulfill more orders. The replenishment policy has the role of maintaining the inventory of the warehouse and keeping the pod at a high service level. Other than that, replenishment triggers reduced visits to the picking station. A pod with insufficient capacity could not be assigned with the new order, although it already has the most order assigned.

Keywords. Robotic Mobile Fulfillment System, Pod Replenishment, SKU Classification, Pod Utilization.

Introduction

Warehouse design has a role in maintaining and improving efficiency and reducing warehouse distribution costs [1]. There are five main groups of warehouse systems "picker-to-parts," "pick-to-box," "pick-and-sort," "parts-to-picker," and "completely automated picking" [2]. In parts-to-picker warehouse integrated with the Internet of Things such as data management, sensors, robot, and many other things to synchronize all warehouse processes. Robotic Mobile Fulfillment System (RMFS) is one breakthrough in implementing e-commerce warehouses and is categorized as the parts-to-picker system. Kiva System, as one of the innovative approaches, uses hundreds of

---

[1] Corresponding Author, Mail: d10101801@mail.ntust.edu.tw.

custom-built mobile robots or automated guided vehicles (AGVs) that carry small shelving units (Pods) and deliver them to the assigned stations (Picking or Replenishment Station) [3]. With proper management or strategies, it can help to speed up the order-picking process. Therefore, this study aims to improve the operation strategies in RMFS.

Some activities directly related to the order picking process are SKU-to-Pod assignment and replenishment. To increase pod utilization, SKUs dimensions, cycles, characteristics, and popularity are considered factors in assigning SKUs to pods [4]. The replenishment policy is designed to avoid stock out of SKUs which can delay orders fulfillment and reduce pod utilization. In a traditional warehouse or "pickers-to-parts," replenishment is always triggered by each SKU's inventory level. If the inventory level of the SKU reaches a certain level, it will order a certain quantity [5, 6]. However, the RMFS's replenishment considered pod condition containing many SKU types. Pod condition is influenced based on the pod's location, free space, and frequency being picked, and the pod's inventory level of an SKU is considered as the trigger for the replenishment [7]. After the pod was triggered to be replenished, AGV delivered the pod to the replenishment station. AGVs could be assigned only for the picking/replenishment process [8] or both processes. This assignment also influenced the time for the pod to get replenishment.

This study aims to design a replenishment policy and SKU-to-pod assignment design, which can maximize pod utilization by considering the inventory level. There are some scenarios for SKU to pod assignment and the replenishment proposed in this study. This study applies a simulation to evaluate the proposed scenarios.

1. Literature Review

The literature review briefly discusses some basic concepts in RMFS, including the components, basic operations, and inventory in the RMFS. RMFS is one of the breakthroughs in warehouse systems to adapt to e-commerce disruption. AGVs, Pods, Replenishment Stations, Picking Stations, and Charging Stations support this system. AGVs bring the pod to picking or replenishment stations and then bring the pod back to the storage area. One of the advantages of this system is eliminating the need for human movement in the storage area [8, 9]. Pods have the role of SKU storage which is divided into a few compartments. AGV can carry a pod weighing around 450 – 1300 Kg, depending on the pod size [10].

The process of RMFS starts when there is an order enters the system. There are two types of orders: picking orders and replenishment orders. Picking order assigned to the pod which has the required SKU. AGV picked the pod from its location and delivered it to the picking station. After finishing the picking process, AGV returned the pod to the empty storage area. A replenishment order is triggered when the pod reaches a certain inventory level [7]. The assignment of SKU to pod should be considered to provide an efficient picking process.

SKUs to Pods assignment is described as SKUs distribution to all pods that are limited to single type SKU in one pod or multiple types SKU in one pod. This assignment influenced the number of units being picked from the shelf. There are three issues related to this assignment. They are a combination of SKU types in one pod, SKU quantity in one pod, and SKU distribution in other pods [11, 12]. After considering these factors, SKUs can be assigned as single SKUs in one pod and multiple in one pod (mixed shelves).

A single SKU in one pod is suitable when there are large quantity orders. However, mixed shelves are more suitable in e-commerce which has small quantity orders [13].

Multiple SKUs can be assigned randomly or following a policy to one pod. Many policies can be used, like SKU similarity [7, 14, 15], Association Rules, and ABC classification [16]. In SKUs similarity, the pod requires maximizing all SKUs similarity. The association rules approach has the same objective of SKU similarity, but it considers the most frequent combination of the SKUs. In ABC classification, the SKUs are classified by the popularity of the demands and SKU types.

Inventory management has an important role in the warehouse management system. The main purpose is to maintain product availability in the warehouse. A replenishment strategy which not designed properly can lead to overstock or stock out in the warehouse, and this can cause a bullwhip effect or disruption to the entire supply chain process [17]. There have been many studies discussing inventory management, especially replenishment policies such as [5, 6, 18-20]. However, the inventory policy needs to be adjusted in the RMFS warehouse using a pod that contains single or multiple types of SKUs. RMFS replenishment needs to consider the pod's condition, which has the most urgent condition that needs to be replenished.

2. Methodology

This study proposes the SKU-to-Pod assignment and replenishment policies to improve warehouse efficiency. The proposed policies are evaluated using a simulation approach.

2.1. Simulation Layout

The RMFS warehouse replicated in this simulation layout consists of three areas: a picking station area, a replenishment station area, and a storage area. The layout for this simulation is shown in Figure 1.

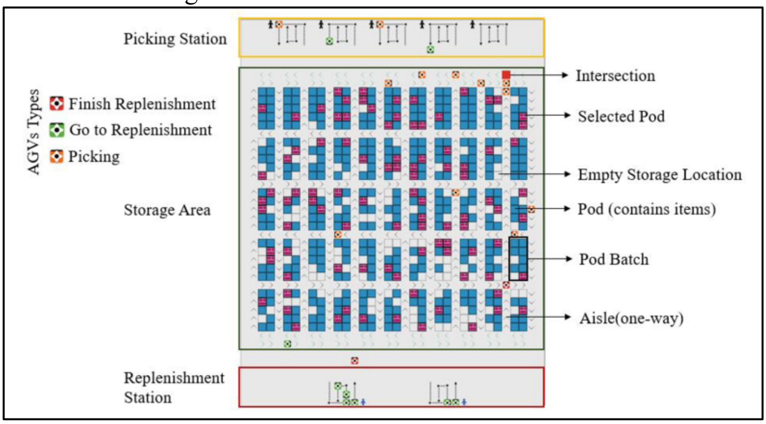

Figure 1. Layout Simulation.

2.2. Process Flow

RMFS process is divided into a few parts based on the relation of each resource. The RMFS process flow can be seen in Figure 2. There are six resources involved in the system: order, SKU, pod, robot (AGVs), picking station, and replenishment station. The interaction between each resource includes SKU to Pod assignment, order arrival, order to pod assignment, pod sequencing, robot to pod assignment, robot routing, robot to stations (Picking station and replenishment station), robot to storage assignment, pod to replenish, and SKU to replenish.

1. ***SKU to Pod Assignment.*** This study applies three scenarios for SKU to Pod assignment. They are a Random assignment, One Pod One Class, and Mix Class One Pod assignment. Random assignment randomly assigns the SKU to the pod based on the SKU limit in the pod. It is used as the baseline. One class one pod uses an ABC rule to classify the number of pods based on 60%, 25%, and 15% pod of SKU A, B, and C, respectively. In the Mix class one pod assignment, a pod consists of several types of SKUs. The composition of SKU types is defined based on the ABC rules.
2. ***Order Arrival***. This study generates the arrival time following a certain distribution. Each arrival time consists of one or more orders with one type of SKU. The total amount of orders generated is between one and two[21]. The type of SKU is based on the following ABC rule where 10% of the SKU is 60% of the order, 30% of the SKU is 25% of the order, and 60% of the SKU is 15% of the order.
3. ***Order to Pod Assignment.*** After generated, each order is assigned to a pod which has the earliest due-date order and has enough capacity. SKU with sufficient capacity picked as the order to pod result. Otherwise, it checked another pod to assign. It means that one pod could serve more than one order.
4. ***Pod Sequencing.*** After the order-to-pod assignment, the system sorts the pod based on the earliest due date of the order. This rule can determine the urgency of the order in the pod.
5. ***Robot to Pod Assignment.*** After the order-to-pod assignment, the pod is labeled as a selected pod. An available AGV is assigned to pick the selected pod with the earliest due date and nearest distance. The total pod that is being considered is double the number of available AGVs. This assignment used the Manhattan distance to calculate the distance [22] and the Hungarian algorithm [23] to pair the AGVs with the pods. Distance calculation considered all the possibilities of the selected pod with the starting and ending intersections[24]. This distance is used by the Hungarian algorithm to find the nearest AGV.
6. ***Robot Routing.*** In this simulation, the routing policy follows previous research using a Simple Routing with Traffic Policy[24]. The AGVs routing is based on the direction in the aisle. There is an exception where the AGV can move beneath the pod horizontally to the destination in the same pod batch. The traffic policy used is deadlock and collision prevention. Also, in each intersection AGV horizontal aisle will be prioritized.
7. ***Robot to Stations Assignment.*** An AGV that has been assigned to the selected pod should move to the station based on the order. The AGV is assigned to the station with the least AGV waiting in picking or replenishment order. Assign AGV to the station with more AGV increased the queuing time in the station.
8. ***Robot to Storage Assignment.*** After the picking or replenishing, the AGV must bring the pod back to the storage area. First, list all the possible empty locations in the storage area. This study utilizes the Hungarian algorithm to determine the nearest open location from the station.
9. ***Pod to Replenish.*** After a pod finishes a picking process, the system evaluates the SKU quantity in the pod. AGV directly delivers the pod to the replenishment station

if the pod needs to be replenished. This study defines four policies that trigger replenishment. They are the Emptiest Pod, Pod Inventory Level, Warehouse Inventory-SKU in Pod, and Stockout Probability. As a basic rule, this study defines that it should be replenished if the Pod Inventory Level is 60% of the total capacity. When a pod goes to the replenishment station, all SKUs are refilled to the maximum capacity.

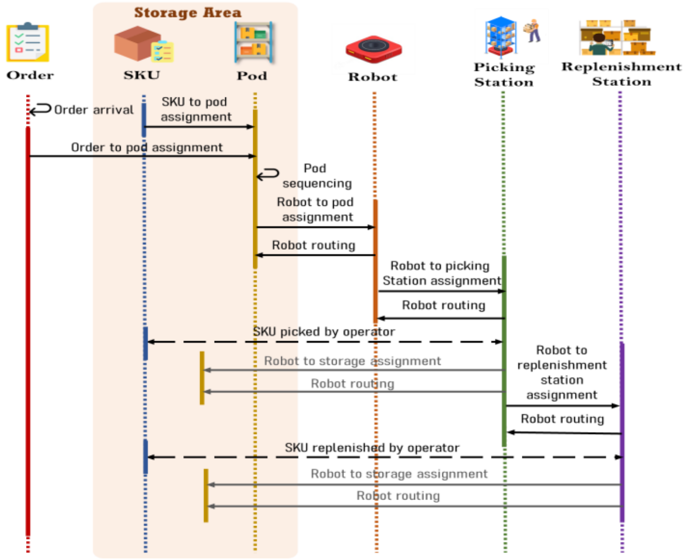

Figure 2. RMFS Process Flow and The Relation Between Each Resource.

This study proposed four replenishment policies. These policies are designed to balance the inventory level of the warehouse. The pod urgency triggers the replenishment after finishing the picking process. These are the replenishment policies used in this study:

a) **Emptiest.** This policy sorts all the pods in the system and labels the pod which has the lowest inventory level. However, if the lowest pod has not been assigned to the picking station, it will stay idle until an AGV is assigned.

b) **Pod Inventory Level**. This policy is considered the inventory level of a pod without considering the SKU types. First, calculate the index of the pod-j inventory level $P_j = \sum_i^{n_j} S_{ij} / C_j, i \in J$, where S_{ij} is the quantity of SKU-i in Pod-j, n_j is total SKU types in pod-j, and C_j is the total unit capacity of pod-j. The replenishment trigger of pod j, $R_j = 1$ if $P_j < IL$, where IL is pod inventory level. Otherwise, $R_j = 0$. If $R_j = 1$ the pod is labeled to get replenished. After labeling a pod as "get replenished", AGV brought the pod directly to the replenishment station. In the replenishment station, SKUs that have already been decreased got replenished, and the total service time of the replenishments is dependent on the total of SKUs that need to replenish.

c) **Stockout Probability.** This policy has the same method as the pod inventory level but considers the stockout probability. The stockout probability can be calculated by considering the order estimation of SKU in the pod. Order generated based on the exponential distribution. The total estimation of orders in one day is 36,000 orders.

The picking time uses the alpha of the gamma distribution parameter. Then, it is generated based on the ABC rule proportion for seven days and summarized in descriptive analysis. The descriptive analysis is shown in Table 1.

Table 1. The Orders Estimation in Each Class.

Class	Average	Standard Deviation	Order Estimation	Avg Pod Distribution/ SKU	Order Estimation/ SKU/ Pod
Class A	43.06	6.19	49	6	8
Class B	7.22	2.57	10	2	5
Class C	1.21	1.04	2	1	2

This order estimation is used in the scenario of replenishment policy. The Stockout Probability of pod-j, (P_j) considers the order estimation in each pod. This policy uses the index of pod stockout level $P_j = \left(\sum_i^{n_j} 1 - S_{ij}/E_{ij}\right)/n_j\, , i \in J$, where E_{ij} is the estimation order of SKU-i in Pod-j. After the stockout level for each SKU already being calculated. Then add the stockout probability of all SKUs to get the pod stockout level (P_j). This showed that the bigger the stockout probability, the more likely to get replenished. Then, the replenishment trigger of the pod inventory level $R_j = 1$ if $P_j > IL$, otherwise $R_j = 0$.

d) **Warehouse Inventory – SKU in Pod.** This policy is conducted by checking the SKU inventory level in the warehouse and the total SKU in one pod that needs to get replenished. First, calculate all SKU-i inventory levels (U_i) in the warehouse $U_i = S_i/X_i\, , i \,\forall\, I$, where S_i is quantity of SKU-I and X_i is maximum quantity of SKU-i. After calculating the index of SKU inventory in the warehouse, it is labeled as an SKU that needs to be replenished (W_i). The $W_i = 1$ if $U_i < UL$, otherwise $W_i = 0$. If $W_i = 1$ the SKU is labeled as the SKU that needs to be replenished. Otherwise, it didn't label to be replenished. Then, calculate the index of the pod-j based on the SKU level (Q_j) that needs to replenish using $Q_j = \sum_1^{n_j} U_{ij}/n_j\; , i \in J$, where U_{ij} is the index of SKU-i in pod-j. The $R_j = 0$ if $Q_j \leq KL$ where KL is SKU inventory level in the pod, otherwise $R_j = 1$. If $R_j = 1$ the pod is labeled to get replenished. Otherwise, it does not get replenished. After a pod is labeled as to get replenished, AGV brings the pod directly to the replenishment station. In the replenishment station, the SKU labeled to get replenished will be replenished to the maximum quantity of the SKU. The total service time of the replenishments depends on the total SKUs that need replenishment.

2.3. Parameter and Performance Analysis

Table 2 lists the assumptions used for all parameters used in the simulation. This study runs the simulation on NetLogo and Python. This study analyzes the proposed scenarios' performance based on pod utilization and inventory analysis. The pod utilization (PR_t) is defined as the number of items picked in each visit to a picking station. It is defined as $PR_t = R_t/Z_t$, where R_t is the total number of items picked in period t and Z_t is the frequency of pods visiting the picking station during the period t. The inventory analysis is defined as $RR_t = B_t/R_t$, where B_t is the total replenishment in period-t, and R_t is the pick unit in period-t. It indicates the capability of the replenishment policy. The scenario shows an unstable system if the ratio is lower than 1. Other than that, the minimum inventory level can also be called the average total inventory.

Table 2. Simulation Parameters

Parameter	Value	
Run Length	8	Hours
Replication	5	Replication
Inventory area	550	Location
Inventory capacity	467	Pods
Empty Storage Area	83	Location
Pod Batch	2 x 5	Blocks
Aisles	12 Vertical; 4 Horizontal	Aisles
Stations	5 Picking; 2 replenishment	Stations
Initial Order	100	Orders
Orders Proportion	A = 60%; B = 30%; C = 10%	Orders
Order Arrival Time	Mean =1.6	Exp. Dist
Order Arrive	1 - 2	Orders
Pod's Capacity	100	Units
Number of SKUs	5000: 10% Class A, 30% Class B, 60% Class C	SKUs
Queuing	5	AGVs
Picking-Time (per picked unit)	Alpha = 12; Beta = 1.5	Gamma. Dist
Queuing Replenishment	5	AGVs
Replenishment-Time	Alpha = 19; Beta = 0.8	Gamma. Dist
Robot speed	1	m/s
Time to lift and store pod	1	Seconds
Number of AGVs	25	AGVs

3. Results and Discussion

Table 3 summarizes simulation results for the emptiest policy. The Random assignment with 60% Pod Inventory Level as the baseline has 92.80% of throughput efficiency. The result shows that the mixed class one pod has the highest throughput efficiency.

Table 3. Performance of The Emptiest.

SKU to Pod Scenario	Throughput Efficiency	Rep/ Pick Ratio	Average pick visit	Average Pick units/visit	Average inventory (%)
Random	94.91%	0.32	808.38	1.23	56.35
One Class One Pod	95.59%	0.38	807.75	1.24	55.80
Mixed Class One Pod	96.36%	0.27	805.65	1.26	55.10

Table 3 also shows that the mixed class one pod has the lowest average pick visit. It indicates that the Mixed Class One Pod can pick more units on each visit. The average inventory of the warehouse shows that all scenarios have an average inventory lower than 59%. This condition showed that replenishment rarely happens. It is because the emptiest pod has not been picked at the picking station. The average replenishment visit for this policy is only 3.5 replenishment/ hour. Other than the unstable system, the performance of this policy is also worse than the baseline. The best result of this policy is compared with the baseline, which increases 125.29% of pick visits and reduces 54.51% of pick units/visit.

The simulation is conducted based on all scenarios of SKU to Pod assignment and replenishment policy with the Pod Inventory Level policy. There are Random, One Class One Pod, and Mixed Class One Pod in SKU to Pod assignments. These SKU to Pod assignments got combined with the Pod Inventory Level with different replenishment levels (40%, 60%, and 80%). The result of the throughput efficiency of the Pod Inventory Level policy combined with all SKU to Pod scenarios is shown in Table 4. The Random assignment with 60% Pod Inventory Level has 92.80% of throughput efficiency. The Mixed Class One Pod for 40% and 80% inventory levels show instability because the replenish/pick ratio is lower than one. The best throughput efficiency is the Mixed Class One Pod, with a 60% inventory level. The best scenario with the lowest scenario in each inventory level is the Mixed Class One Pod. Furthermore, the result shows that the Mixed Class One Pod can have more units being picked on each visit. It shows that lower inventory levels can reduce pod utilization. In 40% inventory level for One Class, One Pod, and Random has lower pick units/ visit than 1.6 units. The inventory condition of the warehouse causes this reduction. Furthermore, the result indicates that the inventory policy needs to be maintained above 59% to have a stable warehouse.

Table 5 shows the simulation results of the stockout probability policy. The Random assignment with 60% Pod Inventory Level as the baseline has 92.80% of throughput efficiency. The result shows that many scenarios have lower throughput efficiency results than the baseline. It indicates that the lower ratio can increase the throughput. Assigning SKU with the same class in one pod (One Class One Pod) has the best result compared with other SKUs to Pod assignments in each inventory level. More popular SKUs assigned in the same pod can increase the stockout probability. Higher stockout probability triggered replenishment more frequently. This result is also verified by the pod utilization. The pod utilization rate shows that One Class One Pod, with an 80% inventory level, has the best average pick units/visits with 2.94 units/visit.

Table 4. Performance of The Pod Inventory Level.

SKU to Pod Scenario	Inventory Level (%)	Throughput Efficiency (%)	Rep/ Pick Ratio	Average pick visit	Average pick unit/visit	Average Inventory (%)
Mixed Class One Pod	40	93.00	0.82	589.03	1.79	56.36
One Class One Pod	40	93.73	1.27	619.98	1.59	59.49
Random	40	95.83	1.19	667.68	0.3	59.85
Mixed Class One Pod	60	92.81	1.02	342.98	2.85	59
One Class One Pod	60	91.71	1.35	357.6	2.77	61.07
Random-Baseline	60	92.80	1.48	373.53	2.65	62.28
Mixed Class One Pod	80	92.50	0.99	346.18	2.84	58.83
One Class One Pod	80	91.35	1.01	355.85	2.75	59.34
Random	80	91.71	1.5	366.45	2.67	62.22

Table 5. Performance of The Stockout Probability

SKU to Pod Scenario	Inventory Level (%)	Throughput Efficiency (%)	Rep/ Pick Ratio	Average pick visit	Average Pick Unit/ Visit	Average Inventory (%)
Mix Class One Pod	40	92.48	0.97	354.90	2.80	58.89
One Class One Pod	40	91.87	1.22	340.20	2.88	60.46
Random	40	92.27	1.35	348.05	2.88	62.06
Mix Class One Pod	60	92.39	0.96	351.95	2.83	58.66
One Class One Pod	60	92.15	1.29	356.27	2.76	60.49
Random	60	92.37	1.50	362.02	2.72	62.23
Mix Class One Pod	80	92.14	0.94	354.85	2.79	58.69
One Class One Pod	80	91.00	1.47	330.10	2.94	61.73
Random	80	96.09	0.00	832.42	1.20	55.26

For warehouse inventory - SKU in Pod performance scenario, there are 27 scenarios with different inventory levels and pod levels. From all scenarios, only ten of them have stable systems shown by the rep/pic ratio, which is greater than 1.00 (see Table 6). The results also show that in terms of throughput efficiency, pick visit, and pick unit/visit, the proposed scenarios are better than the baseline (random scenario). The stockout throughput efficiency is reduced to 1.80%, the pick visit is reduced to 11.63%, and the pick unit/ visit increases to 10.94% from the baseline. The best result is Mixed Class One Pod (60%/60%) with Warehouse Inventory SKU in Pod. Although there is only a slight increase of throughput efficiency, the pick visit is reduced to 14.75% pick visit, and the pick unit/ visit increases to 17.83% pick unit/ visits.

Table 6. Performance of The Stockout Probability

SKU to Pod Scenario	Inventory level	Pod Level	Throughput Efficiency	Rep/ Pick Ratio
One Class One Pod	60	40	91.85%	1.04
		60	93.36%	1.07
		80	92.75%	1.05
Mixed Class One Pod	60	40	90.18%	1.03
		60	91.08%	1.24
		80	90.51%	1.21
Random	60	40	93.68%	1.47
	80	40	93.14%	1.2
		60	92.22%	1.2

4. Conclusion

The RMFS needs to have a proper design to increase the efficiency of the order-picking process. The implementation of the scenarios of SKU to Pod assignment and replenishment policy influenced the performance of the system. SKU to pod has the role of improving the pod utilization by increasing pick unit in each visit. Pod with more SKU types is likely to fulfill more orders. The replenishment policy has the role of maintaining the inventory of the warehouse and keeping the pod at a high service level.

This study proposes three scenarios in the SKU to Pod assignments: Random, divided as One Class One Pod, and Mixed Class One Pod, and four replenishment policies: the Emptiest, Pod Inventory Level, Stockout Probability, and Warehouse Inventory – SKU in Pod. The baseline of this study is Random-Pod Inventory Level with a 60% inventory level. This baseline is compared with other combined scenarios to find the best result. The best result is Mixed Class One Pod combined with the Warehouse Inventory SKU in Pod. There is a slight increase of 0.56% in throughput efficiency. Other than that, the pod utilization increased by 17.83% compared to the baseline. These results can be achieved by maintaining an average of 59.26% warehouse inventory.

Further study should try to implement a more flexible replenishment policy of SKU to the pod. Replenishing the pod with a new set of SKUs should be considered. Replacing less popular SKUs with more popular SKUs might increase pod utilization.

Acknowledgment

This research was partially supported by the National Science and Technology Council of the Republic of China (Taiwan) under grant NSTC 110-2221-E-011-136-MY3 and the Center for Smart Manufacturing Innovation from the Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan.

References

[1] Y. Yu, X. Wang, R.Y. Zhong, and G.Q. Huang, E-commerce logistics in supply chain management: Implementations and future perspective in furniture industry, *Industrial Management & Data Systems,* 2017, Vol. 117, No. 10, pp. 2263-2286.

[2] F. Dallari, G. Marchet, and M. Melacini, Design of order picking system, *The international journal of advanced manufacturing technology*, 2009, Vol. 42, pp. 1-12.

[3] J.J. Enright and P.R. Wurman, Optimization and coordinated autonomy in mobile fulfillment systems, *Workshops at the twenty-fifth AAAI conference on artificial intelligence*, 2011, pp. 33-38.

[4] N. Borin, P.W. Farris, and J.R. Freeland, A model for determining retail product category assortment and shelf space allocation, *Decision sciences*, 1994, Vol. 25, no. 3, pp. 359-384.

[5] M.M. Tanrikulu, A. Şen and O. Alp, A joint replenishment policy with individual control and constant size orders, *International Journal of Production Research*, 2010, Vol. 48, no. 14, pp. 4253-4271.

[6] H. De Vries, R. Carrasco-Gallego, T. Farenhorst-Yuan and R. Dekker, Prioritizing replenishments of the piece picking area, *European Journal of Operational Research*, 2014, vol. 236, no. 1, pp. 126-134.

[7] T. Lamballais Tessensohn, D. Roy, and R.B. De Koster, Inventory allocation in robotic mobile fulfillment systems, *IISE transactions*, 2020, Vol. 52, no. 1, pp. 1-17.

[8] M. Merschformann, T. Lamballais, M. De Koster, and L. Suhl, Decision rules for robotic mobile fulfillment systems, *Operations Research Perspectives*, 2019, vol. 6, 100128.

[9] M. Merschformann, L. Xie, and H. Li, RAWSim-O: A simulation framework for robotic mobile fulfillment systems, *arXiv preprint arXiv:1710.04726,* 2017.

[10] K. Azadeh, R. De Koster, and D. Roy, Robotized and automated warehouse systems: Review and recent developments, *Transportation Science*, 2019, Vol. 53, no. 4, pp. 917-945.

[11] R. D. Meller and J. A. Pazour, A heuristic for SKU assignment and allocation in an A-frame system, *IIE Annual Conference. Proceedings*, 2008, p. 770.

[12] F. Weidinger, N. Boysen, and M. Schneider, Picker routing in the mixed-shelves warehouses of e-commerce retailers, *European Journal of Operational Research*, 2019, 274, no. 2, pp. 501-515.

[13] N. Boysen, R. De Koster, and F. Weidinger, Warehousing in the e-commerce era: A survey, *European Journal of Operational Research*, 2019, Vol. 277, no. 2, pp. 396-411.

[14] H.-J. Kim, C. Pais, and Z.-J. M. Shen, Item assignment problem in a robotic mobile fulfillment system, *IEEE Transactions on Automation Science and Engineering*, 2020, Vol. 17, no. 4, pp. 1854-1867.

[15] R. Yuan, H. Wang, and J. Li, The pod assignment model and algorithm in robotic mobile fulfillment systems, *2019 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, 2019, pp. 99-103.

[16] K.-M. Tsai, M.-h. Chen, P. Bremer, and T.-Y. Chen, Implementing Association Rules for Rack Replenishment in Kiva Systems, *Proceedings of the 24th International Symposium on Logistics "Supply Chain Networks vs Platforms: Innovations, Challenges and Opportunities",* Würzburg, 14th-17th July, 2019, pp. 555-559.

[17] H. Lee and J. Wu, A study on inventory replenishment policies in a two-echelon supply chain system, *Computers & Industrial Engineering*, 2006, Vol. 51, no. 2, pp. 257-263.

[18] G. Buxey, Reconstructing inventory management theory, *International Journal of Operations & Production Management,* 2006, Vol. 26, No. 9, pp. 996-1012.

[19] J. Acimovic and S.C. Graves, Mitigating spillover in online retailing via replenishment, *Manufacturing & Service Operations Management*, 2017, Vol. 19, no. 3, pp. 419-436.

[20] K.H. Van Donselaar, V. Gaur, T. Van Woensel, R.A. Broekmeulen, and J.C. Fransoo, Ordering behavior in retail stores and implications for automated replenishment, *Management Science*, 2010, Vol. 56, no. 5, pp. 766-784.

[21] T.L. Tessensohn, D. Roy, and D.K. René B.M., Inventory Allocation in Robotic Mobile Fulfillment Systems, *IISE Transactions*, 2020, Vol. 52, no. 1, pp. 1-17.

[22] M.D. Malkauthekar, Analysis of Euclidean Distance and Manhattan Distance Measure in Face Recognition, *Computational Intelligence and Information Technology*, Mumbai, 2013, vol. 3, pp. 503-507.

[23] G.A. Mills-Tettey, A. Stentz, and M.B. Dias, *The Dynamic Hungarian Algorithm for the Assignment Problem with Changing Costs*, 2007, https://www.ri.cmu.edu/pub_files/pub4/mills_tettey_g_ayorkor_2007_3/mills_tettey_g_ayorkor_2007_3.pdf, accessed July 6 2023.

[24] A. Y. Kosasih, *Traffic Control and Coordination for Robot Motion Optimization on Robotic Mobile Fulfillment System*, NTUST, Taipei, Master Thesis 2020.