A NOVEL APPROACH TO ANALYZE INVENTORY ALLOCATION DECISIONS IN ROBOTIC MOBILE FULFILLMENT SYSTEMS

T. Lamballais
Rotterdam School of Management, Erasmus University, Rotterdam

D. Roy
Indian Institute of Management, Ahmedabad, Ahmedabad

M.B.M. de Koster
Rotterdam School of Management, Erasmus University, Rotterdam

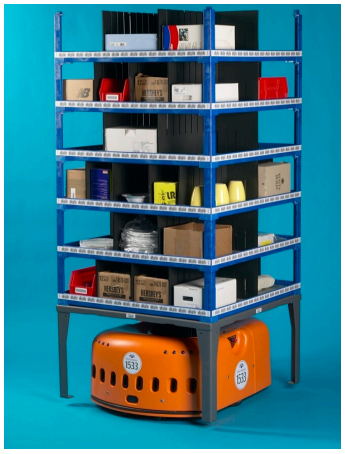
Abstract

The Robotic Mobile Fulfillment System is a newly developed automated, parts-to-picker material handling system. Storage shelves, also known as inventory pods, are moved by robots between the storage area and the workstations, which means that they can be continually repositioned during operations. This paper develops a queueing model for optimizing three key decision variables: (1) the number of pods per product (2) the ratio of the number of pick to the number of replenishment stations, and (3) the replenishment level per pod. We show that too few or too many pods per product leads to unnecessarily long order throughput times, that the ratio of the number of pick to the number of replenishment stations can be optimized for order throughput time, and that waiting to replenish until a pod is completely empty can severely decrease throughput performance.

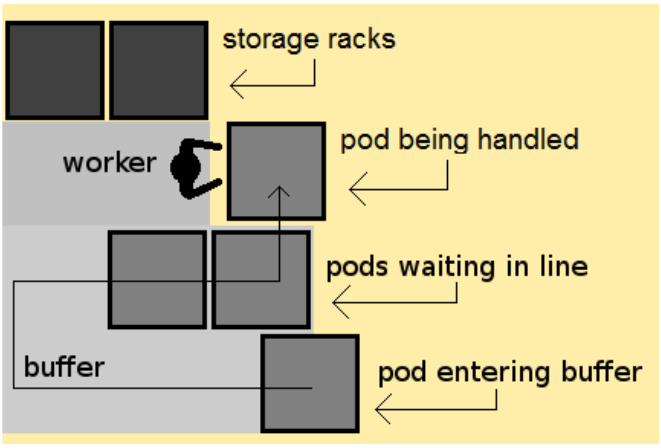
1. Introduction

E-commerce order fulfillment can be quite challenging for warehouses. Assortments tend to be large, orders are typically single-line orders and the order frequency of products can fluctuate strongly. Robotic Mobile Fulfillment Systems (RMFS) are a new category of automated storage and part-to-picker order picking systems developed specifically to fulfill e-commerce orders. These have been brought to the market by companies such as Amazon Robotics (previously known as Kiva Systems, see [3]), Swisslog, Interlink, GreyOrange, Scallog, and Mobile Industrial Robots. Implementations so far suggest that picking rates may double compared to traditional picker-to-parts systems (Wulfraat [17]).

The core innovation of an RMFS are robots that transport the pods, i.e. shelves containing products, to workstations. At a workstation, the pods queue while a worker either picks items from, or replenishes items on, the pod directly in front of him, see Figure 1.



Robot carrying a pod, [19]



Top view of a workstation

Figure 1: Illustration of an inventory pod and close-up of a workstation

An RMFS is flexible in operations, because pods do not need to have a fixed position in the storage area but can instead be repositioned continually throughout the day, see also Wurman and Enright [19]. Inventory can thus be positioned close to the workstations as needed.

In addition, replenishment of a product can happen across multiple pods that can be positioned independently from each other. However, across how many pods should a given amount of a product's inventory be spread? In an e-commerce warehouse, one of the main performance metrics is the order throughput time. If all inventory is allocated to one pod, then there is the risk of temporary unavailability of that product when the pod needs to go for replenishment. If inventory is allocated to multiple pods, however, replenishment happens more frequently and it also becomes less likely that a large order can be fulfilled with inventory from a single pod. In both cases, orders for that product will be delayed and order throughput time increases.

The extent to which this would happen also depends on the replenishment level. A higher replenishment level means that replenishment happens more frequently and may therefore cause additional robot travel time and additional queueing at the workstations. However, it also means that the average inventory on a pod is higher and hence means that orders which require many units have to wait less.

The queueing at the workstations is also influenced by the ratio of the number of pick stations to replenishment stations. A higher replenishment level does not necessary lead to more queueing if the number of replenishment stations is also higher. If the number of pods per SKU and the replenishment level are not optimized, long and unnecessary delays may occur that can have a large impact on the order throughput time. If the ratio of the number of pick to the number of replenishment stations is not optimized, pick stations may have unacceptably low utilization while too much queueing occurs at the replenishment stations, or vice versa.

This paper studies how to minimize the order throughput time by optimizing three

decision variables: (1) the number of pods per product, (2) the ratio of the number of pick to the number of replenishment stations, and (3) the replenishment level per pod.

Section 2 discusses the literature and motivates why a queueing model is suitable for analyzing these decision variables. Section 3 details how the queueing network is constructed, Section 4 provides the results and Section 5 the conclusions and future outlook.

2. Literature

Queueing networks have been used extensively for analyzing the performance of autonomous vehicle storage and retrieval systems (AVS/RS) and automated storage and retrieval systems (AS/RS). These networks can optimize key decision variables, because the low computation time allows evaluation of a large set of parameters. For example, Kuo et al. [7] use queueing models to predict the vehicle utilization and the service, waiting and cycle times while varying five key design variables, namely the number of aisles, the number of storage columns per aisle, the number of storage tiers in the system, the number of vehicles in the system, and the number of lifts in AVS/RS. As another example, Fukunari and Malmborg [5] estimate the expected utilization of resources in an AVS/RS machine using a queueing model that incorporates both single and dual command cycles.

In addition, queueing networks can incorporate the stochasticity of vehicle traveling and the worker speed and can capture the resulting congestion effects, see Tappia et al. [16], Marchet et al. [9], Roy et al. [12], Roy et al. [13], Roy et al. [14] and Roy et al. [11].

Networks where orders arrive and depart from the system can be divided into two broad categories: Open Queueing Networks (OQN) (Heragu et al., [6]) and Semi-Open Queueing Networks (SOQN). SOQNs can capture the matching of different kinds of resources and can therefore include the time an order has to wait before being matched with a vehicle. For example, Roy et al. [10] use a multi-class semi-open queueing network to analyze the performance impact of system parameters such as the number of vehicles and lifts, the depth-to-width ratio and the number of zones. They also study the impact of operational decisions such as vehicle assignment rules on vehicle utilization and order cycle time.

A disadvantage of SOQNs is that they do not have product form solutions and therefore only approximations rather than exact solutions exist. Ekren et al. [4] apply the matrix-geometric method to analyze a SOQN for an AVS/RS and obtain quite accurate performance measures. Roy et al. [10] develop a decomposition approach to evaluate system performance.

Lamballais et al. [8] and Roy et al. [15] develop SOQN for estimating the performance of picking operations in an RMFS. Lamballais et al. [8] optimize the layout of an RMFS warehouse by estimating the expected order cycle time, workstation utilization and robot utilization for a given layout and determining the optimal dimensioning of the storage area, the optimal placement of the workstations. This paper extends the work by Lamballais et al. [8] by considering both pick and replenishment

operations and analyzing inventory allocation decisions.

3. Model

From the perspective of the robot, three movements happen: (a) moving empty or idle to a storage location, then (b) lifting the pod and bringing it to a workstation, and finally (c) moving the pod to another storage location and storing it, after which this cycle repeats Lamballais et al. [8]. Since the workstations can be either pick or replenishment stations and pods may need to wait for an order to arrive, the complete picture from the pod's perspective is more complicated with 8 processes rather than 3 moves.

The pod (1) waits to be matched with an order, (2) waits for a robot to come to its storage location, (3) moves to the pick station, (4) queues for its turn and then has items picked from it, (5) returns to the storage area if its inventory is not below the inventory level, (6) otherwise is brought to a replenishment station, (7) queues for its turn and then is replenished at the replenishment station, and (8) returns to the storage area, see also Figure 2.

Figure 2: Illustration of pod movement

Each of these processes can be modeled as a queue, where the distribution of the travel times in a situation becomes the distribution of the service time of the corresponding queue. The queueing network is shown in Figure 3. It is a Semi-Open Queueing Network to capture the matching of an order to a pod. The numbers in Figure 2 and Figure 3 show which situation corresponds with which queue.

Since RMFSs were designed specifically for e-commerce situations, all orders are

assumed to be single-line orders. If every line requires only one unit of a product, then the queueing model can be solved using the methods in Buitenhek et al. [2] and Bolch et al. [1]. If a line requires more than one unit, then the behavior of the queueing network becomes more complicated, because a pod with only one remaining unit cannot fulfill an order line that needs multiple units. In that case, the queueing network can be analyzed using the corresponding Markov Chain.

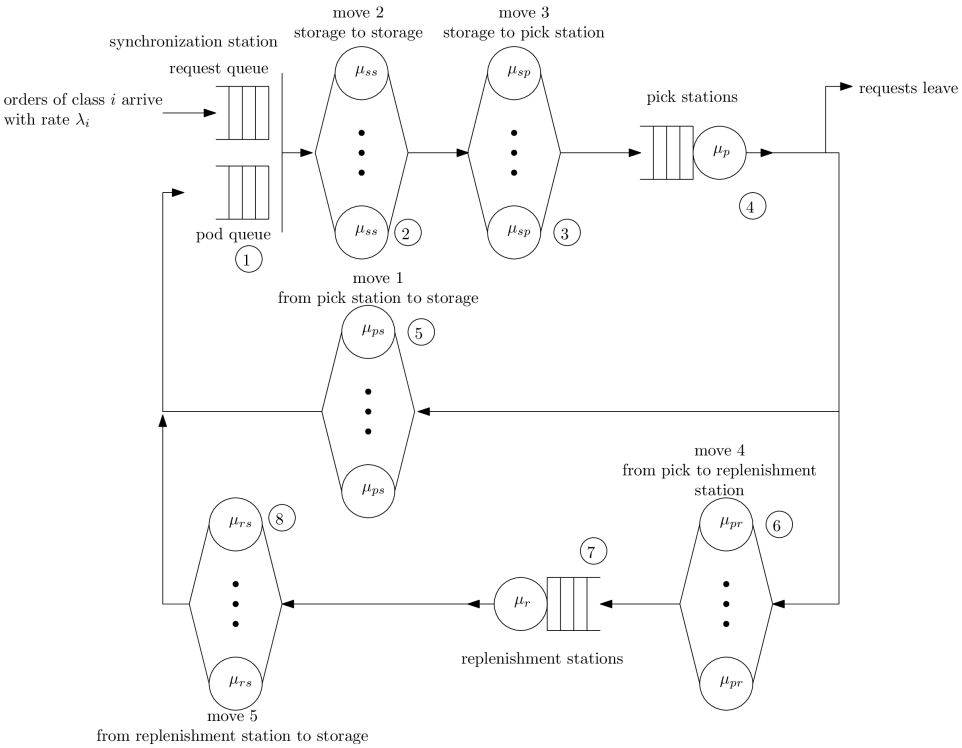


Figure 3: Queueing model of pod movements in the RMFS

Calculating the probabilities of all the states in the Markov Chain will allow the derivation of the performance metrics. Let $M\hat{}s$ be the number of pods for an SKU s, so that the total number of pods equals $N = \sum\_s M\hat{}s$, let π_ϕ be the stationary probability for the state ϕ, let $n\hat{}\phi$ be the number of pods in use in state ϕ, let $o\hat{}\phi$ be the number of orders in the system in state ϕ and let λ be the order arrival rate. Then the order throughput time, t_ot, measured in seconds, and the pod utilization, ρ_pod, can be calculated as:

$$t\_ot = \sum\_\phi \pi\_\phi \ o\hat{}\phi/\lambda$$
$$\rho\_pod = \sum\_\phi \pi\_\phi \ n\hat{}\phi/N$$

Here the formula for t_ot is simply Little's Law, weighted by the state probabilities.

Pod utilization measures the percentage of pods being transported to and from workstations and being handled by workers. In other words, pod utilization measures the percentage of pods carried by robots.

4. Results

Tables 1 shows the results from the experiments. Let U be the number of units on a pod directly after replenishment, let r be the ratio of the number of pick stations to the number of replenishment stations, and let ξ be the replenishment level.

Table 1: Results experiment, t_ot in seconds and ρ_pod in percentages

			$\xi = 0\%$		$\xi = 50\%$		$\xi = 100\%$	
$M\^s$	U	r	t_ot	ρ_pod	t_ot	ρ_pod	t_ot	ρ_pod
1	36	(1, 5)	414.4	11.1	168.3	11.2	171.0	13.2
1	36	(2, 4)	330.2	7.4	92.7	7.5	97.2	9.7
1	36	(3, 3)	332.7	7.2	89.3	7.3	93.8	9.6
1	36	(4, 2)	328.5	7.1	88.6	7.3	93.2	9.7
1	36	(5, 1)	323.7	7.1	88.9	7.3	103.9	13.1
2	18	(1, 5)	172.0	5.7	148.3	5.6	146.3	6.5
2	18	(2, 4)	89.7	3.5	77.0	3.6	77.2	4.6
2	18	(3, 3)	85.6	3.4	73.7	3.5	74.0	4.5
2	18	(4, 2)	85.4	3.4	73.0	3.5	73.3	4.6
2	18	(5, 1)	86.3	3.4	73.3	3.5	74.0	6.5
3	12	(1, 5)	147.3	3.6	141.4	3.7	147.2	4.4
3	12	(2, 4)	74.5	2.3	72.2	2.4	72.2	3.0
3	12	(3, 3)	72.2	2.2	69.1	2.3	69.2	3.0
3	12	(4, 2)	71.5	2.2	68.6	2.3	68.6	3.0
3	12	(5, 1)	70.7	2.2	68.7	2.4	68.8	4.4
4	9	(1, 5)	158.2	2.8	140.0	2.8	140.3	3.2
4	9	(2, 4)	87.5	1.7	69.5	1.8	69.3	2.2
4	9	(3, 3)	81.1	1.7	66.4	1.7	66.3	2.2
4	9	(4, 2)	83.4	1.7	65.7	1.7	65.7	2.2
4	9	(5, 1)	84.8	1.7	66.0	1.8	65.9	3.2
6	6	(1, 5)	139.9	1.8	137.0	1.9	144.5	2.2
6	6	(2, 4)	66.0	1.2	65.9	1.2	65.9	1.4
6	6	(3, 3)	63.3	1.1	63.1	1.2	62.9	1.4
6	6	(4, 2)	62.6	1.1	62.4	1.2	62.5	1.4
6	6	(5, 1)	63.1	1.1	62.6	1.2	62.6	2.1

The number of pods is the same for all SKUs, and varies from $M\^s = 1$ to $M\^s = 6$ in the experiments. The maximum possible inventory in the system per SKU is kept constant at 36 and therefore U varies so that $M\^s\ U = 36$ everywhere. The total number of workstations is 6 and $r\_n = (i, j)$ indicates that i pick stations and j replenishment

stations were present. The total number of SKUs is 100 and per hour two orders arrived per SKU, so $\lambda = 200$.

Table 1 shows that allocating all inventory of an SKU on just a single pod leads to relatively high order throughput times. Generally speaking, the lowest order throughput times seem to be achieved when $M\hat{}s = 6$, so in other words when the inventory of an SKU is spread across as many pods as possible. Table 1 also shows, especially in the case of $M\hat{}s = 1$, that $\xi = 0\%$ leads to suboptimal results. In other words, waiting to replenish a pod until it is empty appears to lead to relatively high order throughput times. Replenishing a pod after every pick operation (the case of $\xi = 100\%$), may not be efficient, but the order throughput times are not much higher than in the case that $\xi = 50\%$, i.e. replenishing a pod when it is half full. The pod utilization does seem to be affected and is clearly higher for $\xi = 100\%$ than for $\xi = 50\%$. In addition, it seems that skewing r too much in favor of the replenishment stations leads to strong increases in the order throughput times. The optimal r in terms of lowest order throughput times depends on both $M\hat{}s$ and ξ. Similar patterns can be observed for pod utilization, which indicates that increased order throughput times are mainly due to longer queueing times at the workstations.

5. Conclusions and Future Work

The results show three main findings. First of all, the number of pods can be optimized, and having only a single pod per SKU results in large increases in order throughput time. Even if all units of a product fit on one pod, it is beneficial to spread the units across multiple pods. A disadvantage of spreading inventory would be that this could results in additional work at the replenishment stations.

Secondly, the optimal ratio of the number of pick to the number of replenishment stations depends on both the number of pods per SKU and on the replenishment level. It also appears that having just a single pick station strongly increases the order throughput times as compared to having more than a single pick station.

Lastly, the replenishment level itself can also be optimized. It seems that waiting to replenish a pod until it is empty severely decreases the performance of the system. However, the effect of replenishing a pod after every pick operation does not seem to have as strong an effect on the order throughput times as may have been expected.

This paper focused on several important tactical decisions, but there are many promising directions for future research, especially with regard to operational decisions. For example, an RMFS is flexible in capacity as robots can be added quickly and workstations can be opened and closed as needed. Another interesting feature is the high degree to which the system's decisions can be decentralized. Robot movement and collision detection was already decentralized in the earliest implementation by Kiva Systems, but other elements such as route planning, task scheduling, and resource allocation can also be decentralized, see Wurman et al. [18].

References

[1] Bolch, G., Greiner, S., de Meer, H., and Trivedi, K. S. (2006). Queueing Networks and Markov Chains. Wiley Publishing Inc.

[2] Buitenhek, R., Van Houtum, G.-J., and Zijm, H. (2000). AMVA-based solution procedures for open queueing networks with population constraints. Annals of Operations Research, 93: 15-40.

[3] Business Wire (2015). Amazon unveils its eighth generation fulfillment center. www.businesswire.com/multimedia/home/20141130005031/en.

[4] Ekren, B. Y., Heragu, S. S., Krishnamurthy, A., and Malmborg, C. J. (2014). Matrix-geometric solution for semi-open queuing network model of autonomous vehicle storage and retrieval system. Computers & Industrial Engineering, 68: 78-86.

[5] Fukunari, M. and Malmborg, C. J. (2009). A network queuing approach for evaluation of performance measures in autonomous vehicle storage and retrieval systems. European Journal of Operational Research, 193: 152-167.

[6] Heragu, S. S., Cai, X., Krishnamurthy, A., and Malmborg, C. J. (2011). Analytical models for analysis of automated warehouse material handling systems. International Journal of Production Research, 49(22): 6833-6861.

[7] Kuo, P.-H., Krishnamurthy, A., and Malmborg, C. J. (2007). Design models for unit load storage and retrieval systems using autonomous vehicle technology and resource conserving storage and dwell point policies. Applied Mathematical Modelling, 31: 2332-2346.

[8] Lamballais, T., Roy, D., and de Koster, M. B. M. (2016). Estimating performance in a robotic mobile fulfillment system. European Journal of Operations Research (EJOR).

[9] Marchet, G., Melacini, M., Perotti, S., and Tappia, E. (2013). Development of a framework for the design of autonomous vehicle storage and retrieval systems. International Journal of Production Research, 51(14): 4365-4387.

[10] Roy, D., Krishnamurthy, A., Heragu, S. S., and Malmborg, C. J. (2012). Performance analysis and design trade-offs in warehouses with autonomous vehicle technology. IIE Transactions, 44: 1045-1060.

[11] Roy, D., Krishnamurthy, A., Heragu, S. S., and Malmborg, C. J. (2013). Blocking effects in warehouse systems with autonomous vehicles. IEEE Transactions on Automation Science and Engineering, 99: 1-13.

[12] Roy, D., Krishnamurthy, A., Heragu, S. S., and Malmborg, C. J. (2015a). Queuing models to analyze dwell-point and cross-aisle location in autonomous vehicle-based warehouse systems. European Journal of Operational Research, 242: 72-87.

[13] Roy, D., Krishnamurthy, A., Heragu, S. S., and Malmborg, C. J. (2015b). Stochastic models for unit-load operations in warehouse systems with autonomous vehicles. Annals of Operations Research, 231: 129-155.

[14] Roy, D., Krishnamurthy, A., Heragu, S. S., and Malmborg, C. J. (2016). A simulation framework for studying blocking effects in warehouse systems with autonomous vehicles. European Journal of Industrial Engineering, 10(1): 51-80.

[15] Roy, D., Nigam, S., Adan, I. J. B. F., de Koster, M. B. M., and Resing, J. (2014). Mobile fulfillment systems: Model and design insights. Working paper.

[16] Tappia, E., Roy, D., De Koster, M. B. M., and Melacini, M. (2016). Modeling, analysis, and design insights for shuttle-based compact storage systems. Forthcoming in Transportation Science.

[17] Wulfraat, M. (2012). Is Kiva systems a good fit for your distribution center? An unbiased distribution consultant evaluation. http://www.mwpvl.com/html/kiva_systems.html.

[18] Wurman, P. R., D'Andrea, R., and Mountz, M. (2008). Coordinating hundreds of cooperative, autonomous vehicles in warehouses. AI Magazine, 29(1): 9-19.

[19] Wurman, P. R. and Enright, J. J. (2011). Optimization and coordinated autonomy in mobile fulfillment systems. Working paper.