



Innovative Applications of O.R.

Estimating performance in a Robotic Mobile Fulfillment System

T. Lamballais^{a,*}, D. Roy^b, M. B. M. De Koster^a^aRotterdam School of Management, Erasmus University, PO Box 1738, Rotterdam 3000 DR, The Netherlands^bProduction and Quantitative Methods Area, Indian Institute of Management, Ahmedabad, Gujarat 380015, India

ARTICLE INFO

Article history:

Received 18 May 2015

Accepted 28 June 2016

Available online 1 July 2016

Keywords:

Facilities planning and design

Queueing

Robots

Mobile fulfillment

Material handling

ABSTRACT

This paper models Robotic Mobile Fulfillment Systems and analyzes their performance. A Robotic Mobile Fulfillment System is an automated, parts-to-picker storage system where robots bring pods with products to a workstation. It is especially suited for e-commerce distribution centers with large assortments of small products, and with strong demand fluctuations. Its most important feature is the ability to automatically sort inventory and to adapt the warehouse layout in a short period of time. Queueing network models are developed for both single-line and multi-line orders, to analytically estimate maximum order throughput, average order cycle time, and robot utilization. These models can be used to quickly evaluate different warehouse layouts, or robot zoning strategies. Two main contributions are that the models include accurate driving behavior of robots and multi-line orders. The results show that: (1) the analytical models accurately estimate robot utilization, workstation utilization, and order cycle time, (2) maximum order throughput is quite insensitive to the length-to-width ratio of the storage area and (3) maximum order throughput is affected by the location of the workstations around the storage area.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

A Robotic Mobile Fulfillment System (RMFS) is a new type of automated storage and part-to-picker order picking system, brought to the market by companies such as Kiva Systems, Swisslog, Interlink, GreyOrange and Mobile Industrial Robots. It is particularly suited for e-commerce distribution centers that handle strong demand fluctuations and large assortments of small products. For example, Amazon bought Kiva systems in 2012 and deployed RMFSs in ten of its warehouses in November 2014 (Business Wire, 2015). Other examples, such as Staples, suggest that the picking rates can double compared to traditional picker-to-parts systems (Wulfraat, 2012). However, installing an RMFS typically requires a multi-million dollar investment, most of which is spent on the robots that carry the pods (see Fig. 1). Therefore, understanding how order cycle time and robot utilization are influenced by warehouse layout and operating policies is important for practice.

This paper develops several models for estimating performance and robot utilization in an RMFS. These models address the most important process in an RMFS, namely the picking process. It is the most important, because it is responsible for picking the

customer orders before their due time. One of the main benefits of an RMFS is that pick rates can reach between 200 and 300 lines per picker per hour (Wulfraat, 2012; Wurman, D'Andrea, & Mountz, 2008). The picking process works as follows. An order arrives and waits until it can be assigned to one of the workstations where the orders are picked (see Fig. 2). Once the order is assigned to a workstation, robots can fetch products for it. Products are stored on inventory pods (i.e., movable shelf racks). A robot moves underneath a pod, lifts it, and brings the pod to a workstation, using the aisles and cross-aisles. The robot enters the workstation buffer and queues for its turn. Each workstation has one picker and once the picker has retrieved the required products from the pod, the robot transports the pod to a storage location and stores it there. The robot then drives to the next pod. As it is moving without a load it does not need to use the aisles but can move underneath the pods. Once all the required products of an order are collected, that order leaves the system and another order can be assigned to the workstation. For a complete description of an RMFS see Enright and Wurman (2011) and Wurman et al. (2008).

A top view of a typical warehouse layout with this system is shown in Fig. 2. The pods are stored in blocks in the storage area with total width W and length L . The dark gray squares represent the pods and the light gray squares represent unoccupied storage locations. The workstations are situated at three sides of the warehouse in Fig. 2. The aisles in the storage area all have a single travel direction to prevent deadlock and reduce aisle congestion.

* Corresponding author. Tel.: +31 104082403.

E-mail address: lamballaistessensohn@rsm.nl (T. Lamballais).

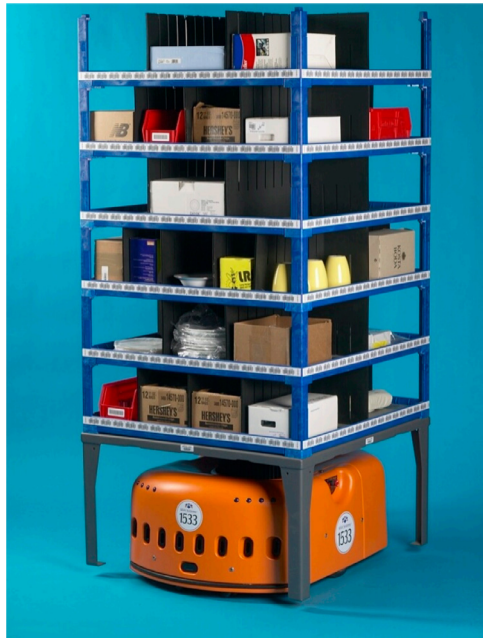


Fig. 1. Robot carrying a pod (Enright & Wurman, 2011).

For the same reason, movement in the space between the workstations and storage area is also single directional. Fig. 3 gives a close-up view of the system. The worker picks products from the pod in front of him or her and then adds them to the order totes on the left.

During the day, a pod is not required to maintain a fixed position, but can continually be repositioned. By changing the locations of the pods, the system can automatically sort inventory during operations and adapt to varying demand in the short run. The advantage is that the most popular products are usually located close to the workers, even during periods of strong demand fluctuation. Another advantage is that the layout of the warehouse can be rearranged relatively quickly. The number of workstations and their positions can easily be adapted to the changing numbers of workers in each shift. In addition, if storage

capacity becomes insufficient, the layout can be adjusted by adding more pods and storage locations. In other words, the layout is not static, but can be changed to suit changing circumstances relatively quickly.

Typically, the storage area is quite compact, because it only contains products needed within the next few days. With enough robots, workers can be kept busy continuously. So far, few analytical models have been developed to estimate the performance or robot utilization of an RMFS. This paper develops four queueing network models to estimate performance and robot utilization under different system parameters, warehouse layouts, and control policies. All models focus on the performance of a workstation in isolation, but they differ in whether they allow only single-line orders or also multi-line orders, and in whether they divide the storage areas into zones or not. These analytical models require very little computation time and can therefore be used to rapidly optimize the warehouse design, which is not easily possible using simulation models. In addition, the development time needed to adapt these models to analyze a specific warehouse setting will be less than what is needed for a simulation. The queueing models can incorporate the stochasticity in the travel times of the robots and the time that orders have to wait before they can be released to the system, so that the robot utilization and performance metrics such as order throughput and order cycle time can be estimated. By measuring order throughput, order cycle time, and robot utilization, these models enable warehouse managers and system developers to predict performance and optimize warehouse design. These models also enable researchers to rapidly compare the performance of the RMFS to other automation systems.

This paper will answer the following design-related research questions. How many orders can be completed per hour given a certain number of robots and workstations? How does the length-to-width ratio of the storage area affect maximum order throughput performance? How does the location of the workstations in the storage area affect maximum order throughput performance? How many robots are needed to achieve a certain desired throughput level and order cycle time?

The remainder of this paper is as follows: Section 2 reviews the literature, Section 3 explains the models, Section 4 describes the data and results, and Section 5 draws conclusions and provides directions for future research.

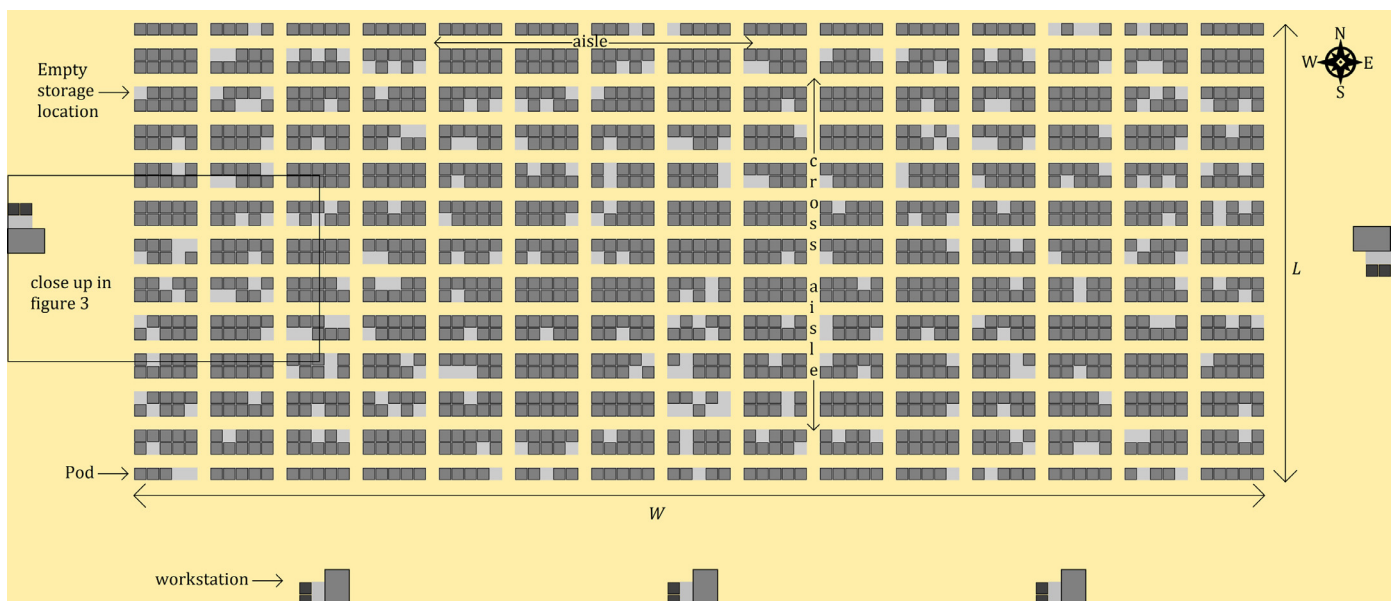


Fig. 2. Top view of an RMFS with workstations on three sides.

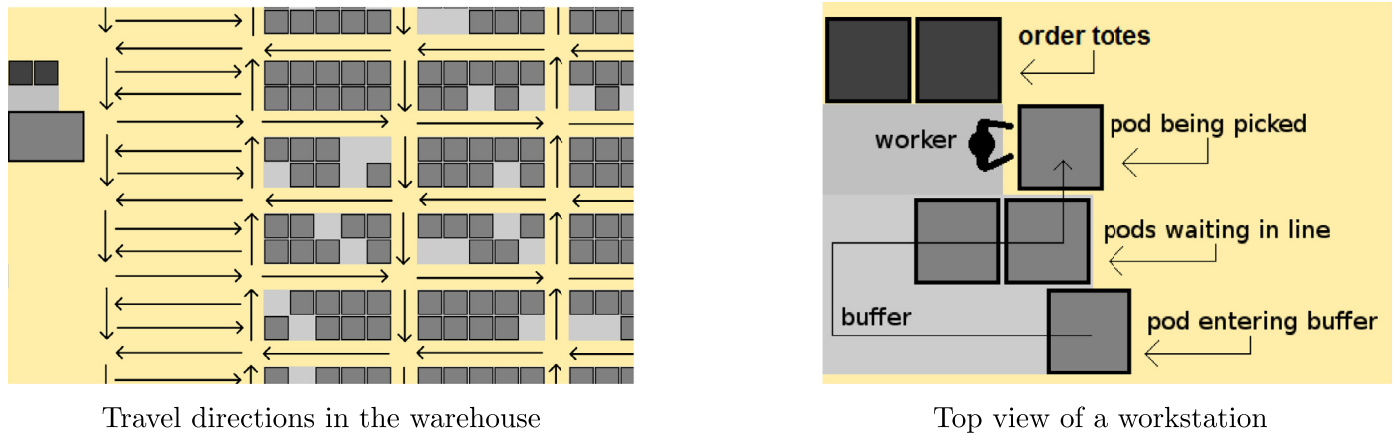


Fig. 3. Close-ups of parts of the RMFS.

2. Literature

Lu, McFarlane, Giannikas, and Zhang (2016) proposed dynamic order-picking strategies that allow for changes of pick-lists during a pick cycle, which have attracted attention with increase in e-commerce orders. Several modeling and performance analysis studies were also carried out on unit-load vehicle-based storage and retrieval systems. However, these studies consider strict rectilinear travel for storage and retrieval with one load/unload point only (see Marchet, Melacini, Perotti, & Tappia, 2013; Roy, Krishnamurthy, Heragu, & Malmberg, 2015a, b, 2016; Tappia, Roy, De Koster, & Melacini, 2016). Queueing models are popular for analyzing automated warehouse systems, because they can incorporate the stochasticity in the travel times of vehicles and in the speed of the workers, and can establish the effect on performance. Queueing networks have been developed for warehouse automation systems such as autonomous vehicle storage and retrieval systems (AVS/RS) and automated storage and retrieval systems (AS/RS). Kuo, Krishnamurthy, and Malmberg (2007) use queueing models to focus on five key design variables in AVS/RS systems for predicting vehicle utilization and service, waiting and cycle times. These five key design variables are 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 the system. Their main conclusion is that these models are computationally effective for exploring the effect of these key variables. Fukunari and Malmberg (2009) develop a queueing model that can estimate the expected utilization of resources in an AVS/RS machine and that can incorporate both single and dual command cycles. Schleyer and Gue (2012) develop a queueing model to estimate the distribution of the order throughput time. This queueing model can handle both single-line and multi-line orders, and the model is based on discrete time to better capture arrival rates from empirical data. Heragu, Cai, Krishnamurthy, and Malmberg (2011) model variants of both AVS/RS and AS/RS as Open Queueing Networks (OQN) and analyze the OQNs using a tool called the manufacturing system performance analyzer. Their conclusion is that this approach works better than simulation for rapidly evaluating different designs.

Besides OQNs, a number of papers use semi-open queueing networks (SOQN) for modeling a system because they can include the time an order has to wait before being processed. Roy, Krishnamurthy, Heragu, and Malmberg (2012) conduct a performance analysis for AVS/RS using a multi-class semi-open queueing network. This work explores the impact of system parameters, for example, the number of zones, the depth-to-width ratio, the number of vehicles and lifts, and the impact of operational decisions such

as vehicle assignment rules on performance measures such as cycle times and vehicle utilization. As SOQNs do not have closed form expressions, they develop a decomposition approach to evaluate system performance. Roy, Krishnamurthy, Heragu, and Malmberg (2013) study blocking in AVS/RS and the effect on transaction cycle times and system throughput. They use a semi-open queueing model and vary system parameters to study the effect of blocking delays within the AVS/RS. Ekren, Heragu, Krishnamurthy, and Malmberg (2014) use a SOQN to analyze an AVS/RS and apply the matrix-geometric method to solve the model, and obtain quite accurate performance measures.

To the best of our knowledge, Roy, Nigam, Adan, de Koster, and Resing (2014) is the only paper which develops queueing networks for an RMFS. However, they estimate order throughput time only for single-line orders and do not include zoning. This paper builds on this work by developing a queueing model that includes storage zoning and multi-line orders. In addition, it models robot travel underneath the pods and assumes a layout that is more realistic with multiple cross-aisles.

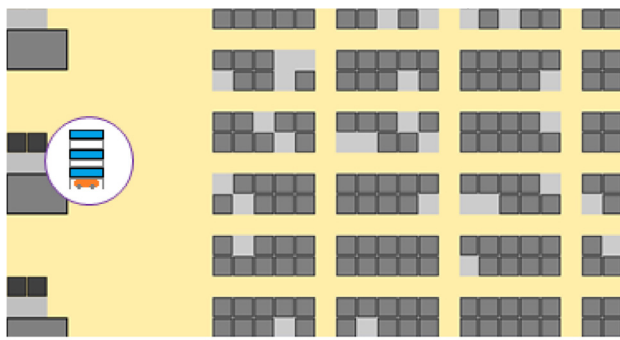
3. Models

3.1. Approach

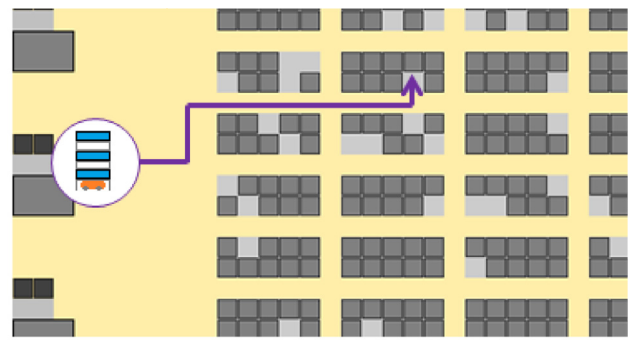
The aim of this paper is to construct an analytical model to study system performance. Performance is measured using three metrics, namely order throughput, average order cycle time, and robot utilization. Order throughput is the rate of orders leaving the system, the average order cycle time is the average time between order arrival at and departure from the warehouse, and robot utilization is the percentage of time that a robot is assigned to an order, averaged over all robots. This network should accurately estimate the three metrics, given system parameters such as the number of pods, robots and workers, and given different warehouse designs and different workstation locations. The network analyzes the performance of one workstation in isolation. The first, basic network assumes that all orders are single-line orders. The first extension to this model is to include storage zones. This means that the storage area is divided into multiple, non-overlapping regions called storage zones, where products are assigned to a storage zone depending on their demand frequency. Other forms of zoning are absent, so robots can work at any location and are not restricted to certain zones. The second extension can also handle multi-line orders.

This results in the following four models:

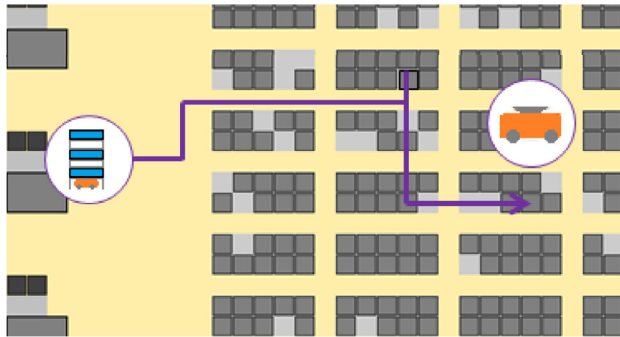
- Model M_1 : single-line without storage zones.



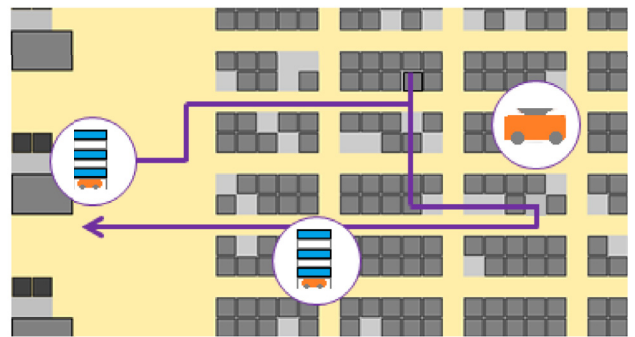
A robot with a pod at a workstation



Move 1: from workstation to pod storage



Move 2: from pod storage to pod retrieval



Move 3: from pod retrieval to workstation

Fig. 4. Robot movements.

- Model M_2 : single-line with storage zones.
- Model M_3 : multi-line without storage zones.
- Model M_4 : multi-line with storage zones.

The main assumptions are the following: (1) Storage and retrieval occurs at a random location. If storage zones are present, this location is random within the appropriate zone and otherwise it is a random location within the entire storage area. (2) Robots are dedicated to a workstation and are not used by another workstation if they are idle. (3) Aisles have single directional travel everywhere. (4) Delays at aisle intersections do not occur and neither do battery recharges or robot downtime. (5) Robot velocity is constant. (6) Robot congestion or blocking in aisles does not occur. This assumption is close to reality, since aisles are single directional and hence deadlock rarely occurs. (7) The storage area always contains a pod with sufficient units of a product to satisfy any incoming order line. (8) The Point Of Service Completion (POSC) is the dwell point policy for robots, which means that robots do not have to travel to a predetermined dwell point after a service completion. (9) Picking time is stochastic rather than deterministic, because the number of units needed to satisfy an order line vary. (10) The picking time follows a general distribution with mean $\frac{1}{\lambda}$. (11) The order lines of an order are picked sequentially. (12) The order arrival process follows an exponential distribution.

The subsections below explain the four queueing networks. This is followed by an explanation of calculating the travel times and thereafter by the analysis of the queueing networks.

3.2. Model M_1 : single-line without storage zones

The first queueing network describes a single workstation. It is based on three basic robot movements as depicted in Fig. 4. Suppose that the picker has completed picking products from a pod.

This means that an order line was filled using a product stored on that pod. The robot moves the pod from the workstation to a storage location and stores it. This is move 1. When it is matched with a request to retrieve a product for another order line, the robot will move from the storage location and move to a pod that contains that product. It then lifts this pod and takes it to the workstation. This is move 2. Upon arrival, it lifts the new pod and brings it to the workstation. This is move 3. Fig. 5 shows the queueing model that corresponds to this process.

At the workstation, robots queue until it is their turn, and the worker picks first-come first-serve with an average rate of λ lines per time unit. Each workstation has exactly one worker, therefore workstations are modeled as single server stations. After the picker has finished with the pod, the order line leaves the system. The robot with pod then enters a station that models move 1, namely traveling to a storage location and storing the pod. Once the pod is stored, the robot is unloaded and must be matched with a new order line at the synchronization station. The dwell policy is POSC, so the robot waits under the pod. Order lines arrive at the synchronization station and are synchronized with idle robots. After the synchronization station, two Infinite Server (IS) stations model the time it takes to execute move 2 and move 3, respectively.

Move 1 models travel from the workstation to the storage location where the pod needs to be stored. The storage policy is random storage, so the robot goes to any of the storage locations with equal probability. It is possible to obtain a distribution for the time a robot needs for move 1 by calculating the travel times between the workstation and each storage location (see Section 3.6) and weighing those travel times with these probabilities. The service rate is μ_1 , as depicted in Fig. 5. Service time distributions for move 2 and move 3 can be constructed in a similar way, where μ_2 and μ_3 are the service rates of the IS stations for move 2 and 3, respectively.

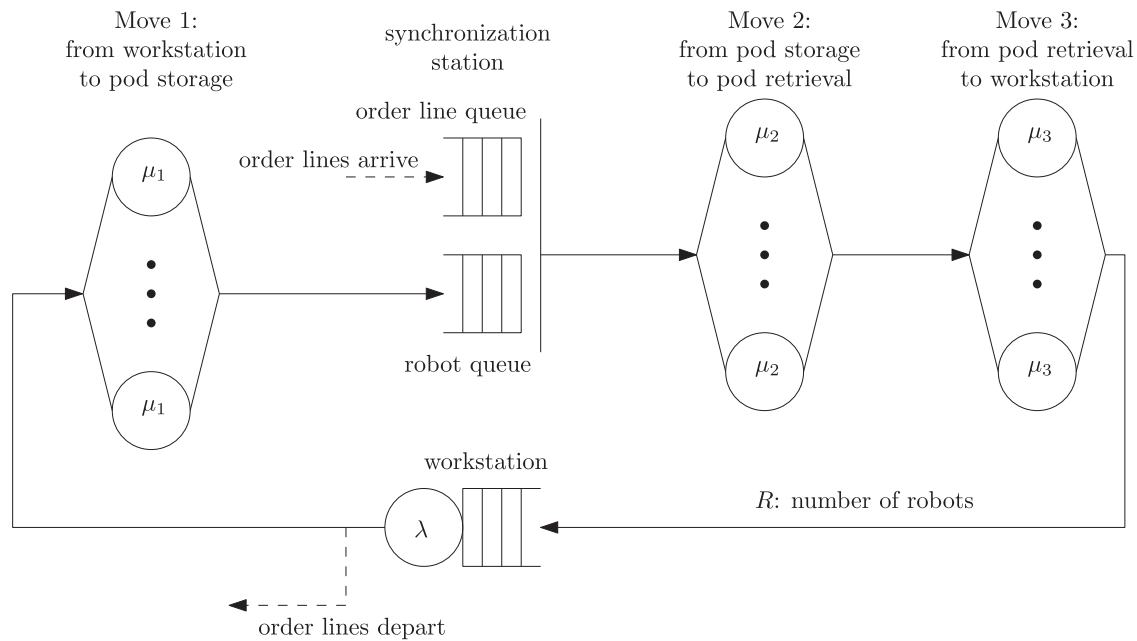


Fig. 5. Model M_1 : single-line without storage zones.

All the queues modeling travel are IS stations, because robots do not need to queue to begin traveling. The network is modeled as a Semi-Open Queueing Network (SOQN) to capture the time that the order lines have to wait before being matched with a robot. Without the time needed for synchronization, the model would estimate the maximum order throughput possible rather than the actual throughput for a given order line arrival rate.

3.3. Model M_2 : single-line with storage zones

In an RMFS, pods with popular products tend to be stored near workstations and those with less popular products tend to be stored further away from workstations. The main purpose is to reduce travel time. The idea of storage zones incorporates this aspect into the model. In a zoned storage system, each storage zone corresponds to a particular part of the storage area and products are assigned to storage zones based on their demand frequency. The probability that an order line needs products on a pod belonging to a zone z is denoted by p_z . Robots are dedicated to workstations but not to storage zones; each robot can visit each storage zone. Storage zones do not overlap, so all storage zones together cover the entire storage area and multiple robots can be in the same storage zone simultaneously.

The model contains a total of Z zones. This means that the model contains Z stations modeling move 1, one station for each of the zones. Move 3 is modeled in a similar way, using Z stations. Move 2 is modeled using $Z \times Z$ stations, because the robot can be in any of the Z zones after storing a pod and may need to go to any of the Z zones to retrieve the next pod.

This model is shown in Fig. 6. Storage and retrieval within the zones are random. Here μ_{z1}^{-1} is the average travel time from the workstation to a random storage location in zone z , with subscript 1 referring to move 1. In other words, μ_{z1}^{-1} is the service rate of the IS station for zone z and move 1. μ_{ij2}^{-1} is the average travel time from a random storage location in zone i to a random storage location in zone j , with subscript 2 referring to move 2. μ_{z3}^{-1} denote the average travel time from a random storage in zone z to the workstation, with subscript 3 referring to move 3.

The routing probabilities shown in Fig. 6 are based on the probabilities p_z . For example, consider a scenario for move 2, where the

robot stores a pod in zone 1 and needs to retrieve a pod in zone Z . The probability of this scenario occurring is $p_1 \times p_Z$, because the probability that the pod that was stored belongs to zone 1 is p_1 and the probability that the pod that needs to be retrieved belongs to zone Z is p_Z .

The division of the storage area into zones is workstation dependent, see the examples in Fig. 7. In these examples, the number of storage zones Z equals three, and zone 1 covers about 20 percent of the storage area, zone 2 about 30 percent, and zone 3 about 50 percent. For workstations that are located west or south of the storage area, the division is as indicated in Fig. 7. The zoning is assumed to be workstation dependent. This implies that when the layout has one workstation located west of the storage area and another one east, then a storage location close to the one located west would be in zone A in the analysis of that workstation, but when analyzing the workstation located east, it would be a zone C location. The zones indicate the likelihood that a pod is retrieved from that area to the workstation (see Fig. 7). This concurs with practice, since copies of fast movers can be stored on multiple pods and the system continues to reconfigure to keep the most popular products near the workstations (Wurman et al., 2008).

3.4. Models M_3 and M_4 : multi-line without and with storage zones

This section extends both models 1 and 2 to multi-line orders. It assumes that the number of lines in an order follows a geometric distribution with parameter p . The average number of order lines is therefore $\frac{1}{p}$. Model 3 extends model 1 with multi-line orders and is shown in Fig. 8.

During move 1, the robot transports the pod to a storage location and stores it. With probability $1 - p$, the order that was assigned to the robot needs more order lines and the robot goes to the IS station, modeling move 2. With probability p , the order that was assigned to the robot needs no more order lines, the order leaves the system and the robot goes to the synchronization station to wait for a new order. Model 2 can be extended in a similar way to arrive at model 4, a model with multi-line orders and storage zones.

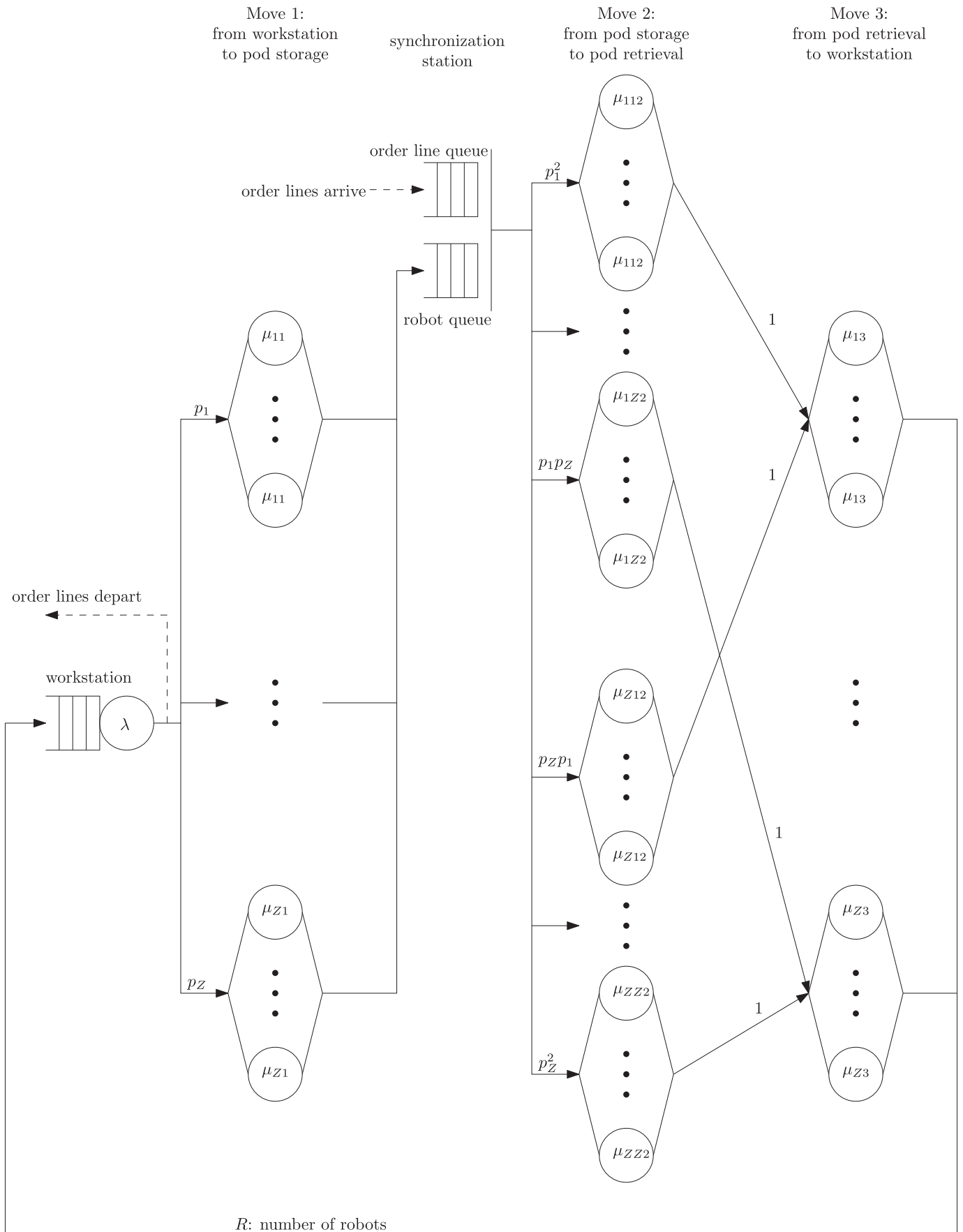
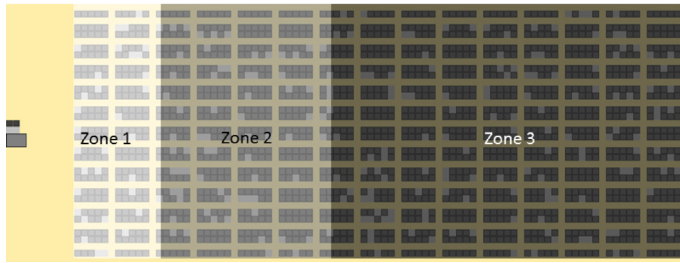
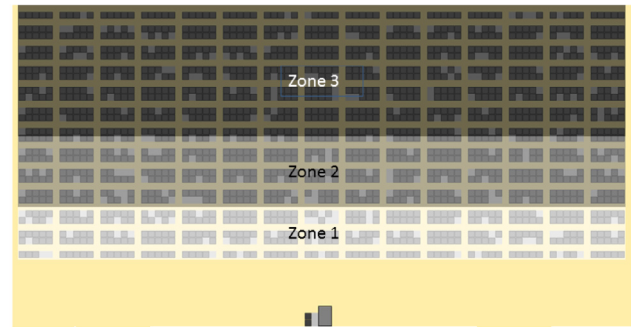


Fig. 6. Model M_2 : single-line with storage zones.



Division for a workstation located west of storage area



Division for a workstation located south of storage area

Fig. 7. Division of the storage area in three storage zones.

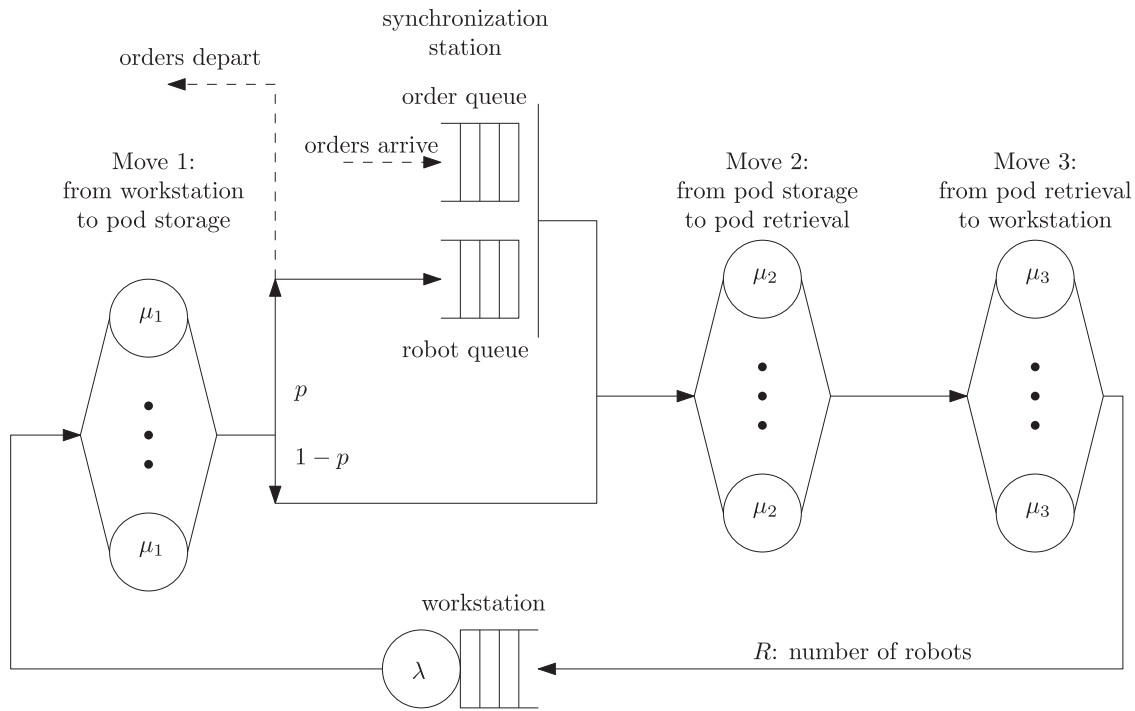


Fig. 8. Model M_3 : multi-line without storage zones.

3.5. Analysis

The queueing networks of models 1–4 are analyzed as single class Semi-Open Queueing Networks (SOQN) and solved using the solution procedure from Section 2.2 of Buitenhek, Van Houtum, and Zijm (2000). This procedure to solve a SOQN follows 3 steps. Step 1: A Closed Queueing Network (CQN) is created by removing the synchronization station from the SOQN. This CQN is analyzed with an Approximate Mean Value Analysis (AMVA) (see Appendix A). The AMVA yields τ_{CQN1} , the throughput of the CQN.

Step 2: A second CQN is created by replacing the synchronization station in the SOQN with a load-dependent exponential station. This station is denoted as station $S + 1$, with S the number of stations in the first CQN. Station $S + 1$ has service rate $\nu(r) = a$ for $r > 1$, when r robots are at the station. Here a denotes the arrival rate of the orders. The network is only stable if $a < \tau_{CQN1}$. For $r = 1$ the service rate is $\nu(1) = (1 - \frac{a}{\tau_{CQN1}})a$. The same AMVA algorithm can then be used to analyze this second CQN, yielding the throughput τ_{CQN2} . This AMVA algorithm also calculates $L_s(r)$ the

queue length at station s when r robots are present. Step 3: the solution procedure analyzes station $S + 1$ in isolation to calculate L_o , the mean length of the external queue of orders.

The other measures of interest are ρ_r , the utilization of the robots, t_{oc} , the order cycle time, and ρ_{ws} , the utilization of the workstation. Let L_r be the expected length of the robot queue at station $S + 1$, as found by the AMVA algorithm for the second CQN. Then $\rho_r = 1 - \frac{L_r}{R}$, where R denotes the total number of robots in the system. Let L_i be the sum of the expected queue lengths at the other stations, so $L_i = \sum_s L_s(R)$. Then the average order cycle time is as depicted in Eq. (1),

$$t_{oc} = \frac{L_o + L_i}{a} \tag{1}$$

The workstation utilization is $\rho_{ws} = \tau_{CQN2} v_{ws} E S_{ws}$, with v_{ws} the visit ratio of the workstation and $E S_{ws}$ the mean service time at the workstation, see also Appendix A. This method allows each station in the network to have c_s parallel servers. The Infinite Server stations are modeled by setting c_s equals to the number of robots R . The AMVA is an approximation as it uses the first and second

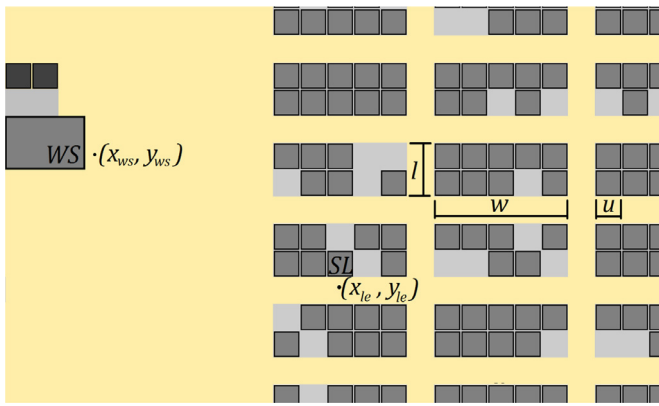


Fig. 9. Notation explained graphically.

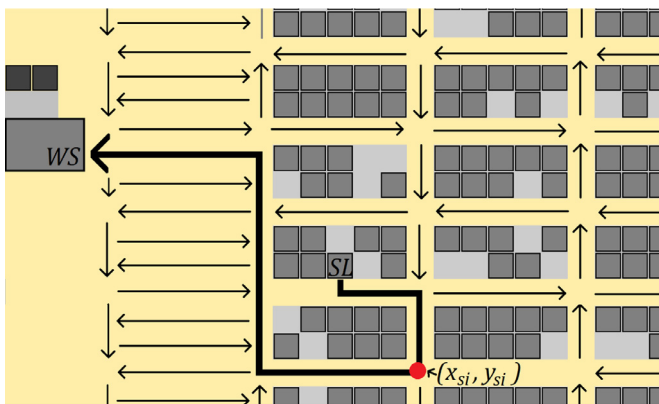


Fig. 10. Shortest route from SL to WS.

moments of the service time distributions as input, allowing the service times to follow a general distribution.

3.6. Travel times

The service time of an IS station in model 1 depends on the travel times that this IS station models. This section explains how to calculate the travel times for each move. During move 1 and 3, the robot is carrying a pod (loaded travel) but in move 2, the robot is not carrying a pod (unloaded travel). The travel distance for unloaded travel is simply the Manhattan distance, but calculating the travel distance for loaded travel is more complicated. Therefore, this section will mostly focus on calculating travel distance for loaded travel.

Once the travel distances are known, they must be divided by the speed of the robot. For move 1, the time needed to store the pod is added and for move 3, the time needed for lifting is added. Storing time, lifting time, and robot speed are assumed to be constants and the robot does not need to accelerate or decelerate. The resulting travel times fully describe the service times of the IS stations in each of the models.

3.6.1. Move 1 and 3: loaded travel

Loaded travel is rectilinear and each aisle has only one travel direction. This means that the travel time to obtain queue length distributions in model 1 can be calculated using closed form expressions. The location entrance of a storage location is the point located in the aisle in front of that storage location. The robot uses the location entrance to enter the storage location. The workstation entrance is the point in the hall from where the robot can enter the buffer of a workstation. The start intersection is a more complicated concept. Suppose that a robot travels from a storage

location that has a location entrance in an aisle with easterly travel direction and travels to a workstation that is located west of the storage area. It then first has to move in easterly direction before it can move west to the workstation, so initially the distance between the robot and its destination will increase. The start intersection is the first point on the robot’s route where the distance decreases. This point is always at an intersection of an aisle and a cross-aisle. More formally: on the shortest route between a storage location SL and a workstation WS, the start intersection is the first intersection with an outgoing arc that points towards the hall in which WS is located. For example, if SL is situated at an aisle with easterly travel direction while WS is located west of the storage area, the start intersection is the first intersection with direction west on the shortest route between SL and WS. Let the length of a block be denoted by l , the width by w , the unit distance by u , let the location entrance (abbreviated as le) of storage location SL be given by (x_{le}, y_{le}) , the entrance of the buffer of workstation WS by (x_{ws}, y_{ws}) , and the start intersection by (x_{si}, y_{si}) . The unit distance is the width of one storage location and in the standard layout (see Fig. 2) $l = 2u$ and $w = 5u$. The aisles are u wide. The distance $d_{ca, le}$ is the distance between (x_{le}, y_{le}) and the first cross-aisle while following the direction of the aisle in which the location entrance is situated. The distance $d_{le, si}$ is the distance between the location entrance and the start intersection. Aisles and cross-aisles can only have one direction, see also Figs. 9 and 10. In Fig. 10, the start intersection is depicted as a big dot.

A shortest route from a storage location SL to a workstation WS can be divided into four parts. The first is the distance between SL and its location entrance, which is equal to u , since both storage locations and aisles are u wide. The second part is the distance between the location entrance and the start intersection. The third part is the Manhattan distance between the start intersection and the buffer entrance of WS, which equals $|x_{si} - x_{ws}| + |y_{si} - y_{ws}|$. The fourth part is a detour $\Delta_{le, ws}$ that may be necessary because of travel directions in the hall between WS and the storage area. This detour $\Delta_{le, ws}$ is either $2u$ or 0 , depending on the location of the buffer entrance of WS. The conditions under which $\Delta_{le, ws} = 2$ are straightforward and simple, but too numerous to list here.

The distance of the shortest route for all storage locations and workstations can be derived from four fundamental cases:

- Case 1: the workstation is located west (east) of the storage area and the location entrance is situated in an aisle with travel direction west (east).
- Case 2: the workstation is located west (east) of the storage area and the location entrance is situated in an aisle with travel direction east (west).
- Case 3: the workstation is located north (south) of the storage area and the first cross-aisle encountered has travel direction north (south).
- Case 4: the workstation is located north (south) of the storage area and the first cross-aisle encountered has travel direction south (north).

In the first case, (x_{le}, y_{le}) is located in an aisle whose direction is towards WS. For example, (x_{le}, y_{le}) is located in an aisle with a westerly travel direction and WS is located west of the storage area. The distance D is as expressed in Eq. (2),

$$D = u + |x_{le} - x_{ws}| + |y_{le} - y_{ws}| + \Delta_{le, ws}. \tag{2}$$

In the second case, (x_{le}, y_{le}) is located in an aisle whose direction is not towards WS. For example (x_{le}, y_{le}) is located in an aisle with a westerly travel direction and WS is located east of the storage area. In this example, the start intersection is the first intersection with travel direction east on the shortest route. If the first cross-aisle west of (x_{le}, y_{le}) has travel direction north, then the start intersection is the intersection to the northwest and if

Table 1
Parameters used in the experiments.

Parameter	Value
Number of aisles	12
Number of cross-aisles	14
Number of storage locations	1800
Number of zones	3: A, B, C zone
Number of storage location per zone	A: 20 percent, B: 30 percent, C: 50 percent
Probability pod comes from	A: 70 percent, B: 25 percent, C: 5 percent
Robot speed	1.3 (meters per second)
R, number of robots	2, 8, 14
Time for pod lifting and storing	1 (seconds)
Distribution picker time	C_k , mean λ is 15 (seconds), cv^2 is 1.0

the cross-aisle has travel direction south, then the start intersection is the intersection to the southwest. In both cases $d_{le,si} = d_{ca,le} + l + 2u + w$. The distance D is as expressed in Eq. (3),

$$D = u + d_{le,si} + |x_{si} - x_{ws}| + |y_{si} - y_{ws}| + \Delta_{le,ws}. \tag{3}$$

In the third case, WS lies to the north or south and the first cross-aisle has a direction towards WS . For example, suppose that WS lies north of SL and that (x_{le}, y_{le}) is located in an aisle with travel direction west, then the first cross-aisle west of SL has travel direction north and the start intersection is the intersection directly west of (x_{le}, y_{le}) . For this case in general, $d_{le,si} = d_{ca,le}$. The distance D is as expressed in Eq. (4),

$$D = u + d_{le,si} + |x_{si} - x_{ws}| + |y_{si} - y_{ws}| + \Delta_{le,ws}. \tag{4}$$

In the fourth case, WS lies to the north or south and the first cross-aisle does not have a direction towards WS . For example, WS lies north of SL and (x_{le}, y_{le}) is located in an aisle with travel direction west, then the first cross-aisle west of SL has travel direction south. In general, one can choose between two possible start intersections, the first cross-aisle to the west with direction towards WS and the first to the east. Following the example, for the first option, the distance D_1 is as expressed in Eq. (5),

$$D_1 = u + d_{ca,le} + w + u + |(x_{le} - d_{ca,le} - w - u) - x_{ws}| + |y_{le} - y_{ws}| + \Delta_{le,ws}. \tag{5}$$

For the second option, the distance D_2 is as expressed in Eq. (6),

$$D_2 = u + d_{ca,le} + 2l + w + 3u + |(x_{le} - d_{ca,le} + w + u) - x_{ws}| + |y_{le} - y_{ws}| + \Delta_{le,ws}. \tag{6}$$

The distance of the shortest route is simply $D = \min(D_1, D_2)$. The first option is not available if the location entrance is situated in one of the westernmost blocks; the second option is not possible if it is situated in one of the easternmost blocks. The northernmost and southernmost blocks have only half the normal length and therefore, in the second and the fourth case, the term l becomes $\frac{1}{2}l$ if the path from location entrance to start intersection goes past these blocks.

All of the preceding formulas have been validated by comparing the results with the shortest routes found by the Dijkstra algorithm for a standard layout. These formulas are for routes from a storage location SL to a workstation WS , but are similar for routes from WS to SL . The formulas above therefore capture the routes for move 1 and 3.

4. Results

The results in this section come from three experiments. The first experiment shows the results for all four models and serves as validation. This experiment uses the standard layout as shown in Fig. 2. The number of robots R , and the average number of order lines per order p are varied to understand the effect on the order cycle time, robot utilization, and workstation utilization. In the second experiment, the effect of the storage area’s length-to-width ratio on maximum order throughput is explored by changing the number of aisles and cross-aisles while keeping the number of storage locations within 4 percent of 1800. In the third experiment, the effect that the locations of workstations along the sides of the storage area has on maximum order throughput is studied.

The traditional ABC categorization is used for zoning. This means that the storage area is divided into an A, B, and a C zone. According to Wulfrat (2012), robot speed is about three miles per hour, which is about 1.3 meters per second, and the average time for picking an order line is six seconds. This excludes the time needed to move the pod in front of the picker, for which no average length is mentioned. Additional experiments in Appendix B show how to calculate the average robot speed given a maximum speed and an acceleration. These experiments indicate that 1.3 meters per second corresponds to a maximum robot speed of 1.5 meters per second and an acceleration of 0.75 meters per second square. According to Wulfrat (2012) and Wurman et al. (2008), pick rates are above 200 lines per hour, therefore the average time for picking is set at 15 seconds in total or 240 lines maximum per hour. The distribution of the picker time is a C_k distribution, which is a Cox- k distribution as described in Bolch, Greiner, de Meer, and Trivedi (2006). The parameters are shown in Table 1, where cv^2 denotes the squared coefficient of variation.

4.1. Experiment 1: a single workstation

In each of the four models a single workstation is analyzed. Since the robots are dedicated per workstation, the results can be calculated by analyzing each of the workstations separately and then taking the average across the workstations. In the tables below, R denotes the number of robots dedicated to the workstation, a denotes the order arrival rate in orders per hour, ρ_r denotes the robot utilization, L_o is the mean length of the external order queue, t_{oc} denotes the average order cycle time in seconds, and ρ_{ws} denotes the utilization of the workstation. For the models with multi-line orders, models M_3 and M_4 , the number of order lines in an order is geometrically distributed, $\sim \text{Geom}(p = k)$, where p refers to the parameter of the Geometric distribution and

Table 2
Results experiment 1: 2 robots and high arrival rate.

Model	$R = 2$ $a(h^{-1})$	Analytical model				Simulation model			
		ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)
M_1	31.68	64.6	0.95	255.2	13.2	64.7	0.52	206.1	13.2
M_2	31.68	49.1	0.34	150.5	13.2	49.1	0.20	133.7	13.2
$M_3(p = 0.5)$	15.84	64.6	0.95	510.4	13.2	64.7	0.74	460.5	13.2
$M_4(p = 0.5)$	15.84	49.1	0.34	300.9	13.2	49.1	0.27	283.8	13.2
$M_3(p = 0.2)$	6.34	64.6	0.95	1276.1	13.2	65.0	0.89	1232.0	13.3
$M_4(p = 0.2)$	6.34	49.1	0.34	752.3	13.2	49.2	0.31	734.8	13.2

Table 3
Results experiment 1: 2 robots and low arrival rate.

$R = 2$		Analytical model				Simulation model			
Model	$a(h^{-1})$	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)
M_1	14.40	29.3	0.06	160.4	6.0	29.3	0.03	154.4	6.0
M_2	14.40	22.2	0.02	117.1	6.0	22.2	0.02	114.6	6.0
$M_3(p = 0.5)$	7.20	29.3	0.06	320.7	6.0	29.4	0.04	314.2	6.0
$M_4(p = 0.5)$	7.20	22.2	0.02	234.2	6.0	22.3	0.02	232.0	6.0
$M_3(p = 0.2)$	2.88	29.3	0.06	801.8	6.0	29.6	0.05	799.4	6.1
$M_4(p = 0.2)$	2.88	22.2	0.02	585.4	6.0	22.4	0.02	582.4	6.1

Table 4
Results experiment 1: 8 robots and high arrival rate.

$R = 8$		Analytical model				Simulation model			
Model	$a(h^{-1})$	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)
M_1	126.72	70.0	0.80	181.8	52.8	69.9	0.51	173.3	52.8
M_2	126.72	54.9	0.22	130.9	52.8	54.8	0.16	129.1	52.8
$M_3(p = 0.5)$	63.36	70.0	0.80	363.7	52.8	70.0	0.64	354.1	52.8
$M_4(p = 0.5)$	63.36	54.9	0.22	261.9	52.8	54.9	0.19	260.0	52.8
$M_3(p = 0.2)$	25.34	70.0	0.80	909.2	52.8	70.0	0.73	898.2	52.8
$M_4(p = 0.2)$	25.34	54.9	0.22	654.6	52.8	54.8	0.21	652.4	52.7

Table 5
Results experiment 1: 8 robots and low arrival rate.

$R = 8$		Analytical model				Simulation model			
Model	$a(h^{-1})$	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)
M_1	57.60	30.1	0.00	150.5	24.0	30.1	0.00	150.4	24.0
M_2	57.60	23.0	0.00	114.9	24.0	23.0	0.00	114.9	24.0
$M_3(p = 0.5)$	28.80	30.1	0.00	301.0	24.0	30.1	0.00	301.0	24.0
$M_4(p = 0.5)$	28.80	23.0	0.00	229.9	24.0	23.0	0.00	229.9	24.1
$M_3(p = 0.2)$	11.52	30.1	0.00	752.6	24.0	30.1	0.00	753.7	24.0
$M_4(p = 0.2)$	11.52	23.0	0.00	574.7	24.0	23.0	0.00	575.1	24.0

Table 6
Results experiment 1: 14 robots and high arrival rate.

$R = 14$		Analytical model				Simulation model			
Model	$a(h^{-1})$	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)
M_1	221.76	91.5	14.80	448.1	92.4	91.3	12.63	412.7	92.4
M_2	221.76	82.6	7.25	305.4	92.4	82.7	7.19	304.5	92.4
$M_3(p = 0.5)$	110.88	91.5	14.80	896.2	92.4	91.4	13.49	852.2	92.4
$M_4(p = 0.5)$	110.88	82.6	7.25	610.9	92.4	82.7	7.35	614.1	92.4
$M_3(p = 0.2)$	44.35	91.5	14.80	2240.6	92.4	91.5	13.88	2159.8	92.4
$M_4(p = 0.2)$	44.35	82.6	7.25	1527.2	92.4	82.5	7.21	1520.7	92.3

Table 7
Results experiment 1: 14 robots and low arrival rate.

$R = 14$		Analytical model				Simulation model			
Model	$a(h^{-1})$	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)
M_1	100.80	31.3	0.00	156.5	42.0	31.4	0.00	156.6	42.1
M_2	100.80	24.2	0.00	121.0	42.0	24.2	0.00	121.1	42.0
$M_3(p = 0.5)$	50.40	31.3	0.00	313.1	42.0	31.3	0.00	312.9	42.0
$M_4(p = 0.5)$	50.40	24.2	0.00	242.1	42.0	24.2	0.00	242.0	42.0
$M_3(p = 0.2)$	20.16	31.3	0.00	782.6	42.0	31.3	0.00	782.5	42.0
$M_4(p = 0.2)$	20.16	24.2	0.00	605.2	42.0	24.2	0.00	604.4	42.0

k is some number between zero and one. The parameter used is indicated by changing the notation to $M_3(p = k)$ and $M_4(p = k)$. Results are shown for R equal to 2, 8 and 14 and for a high and a low arrival rate, leading to six tables in total (Tables 2–7). The arrival rates for 8 robots are 4 times the arrival rates for 2 robots and the arrival rates for 14 robots are 7 times the arrival rates for 2 robots. A discrete event simulation model was built to validate the results. It was built from scratch in Java and simulates the

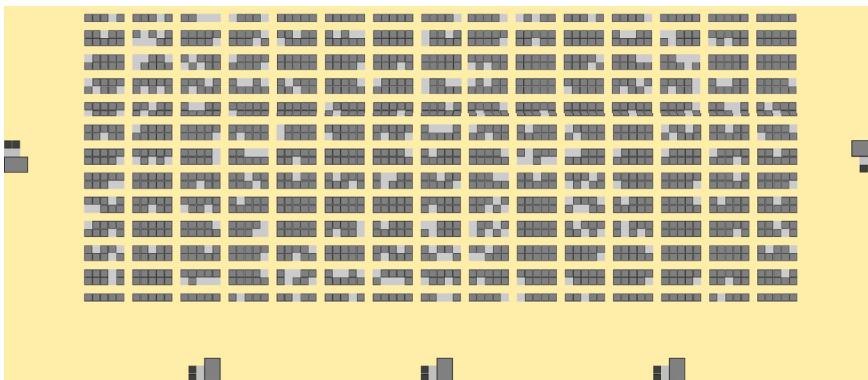
queueing models. The numbers in the tables are averaged over a hundred runs, where each run simulated the network for 168 hours, a full week. The width of the 95 percent-confidence intervals is usually less than 1 per cent of the number itself and at most a few per cent. For $p = 0.5$, the multi-line orders have an average of two order lines and they have an average of five order lines for $p = 0.2$. The order arrival rate was divided by 2 and 5 respectively to ease the comparison with the single-line models. The utilization

Table 8
Results experiment 1: 14 robots, high arrival rate and $cv^2 = 0.6$.

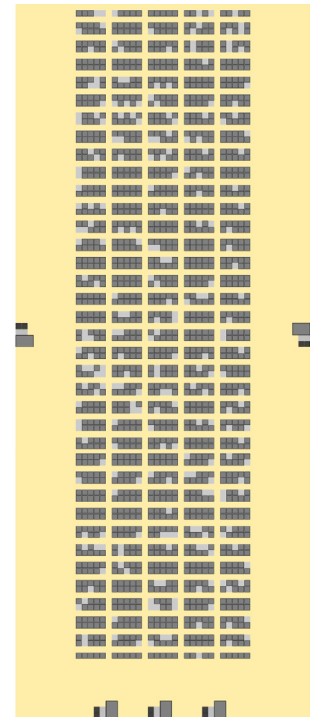
R = 14		Analytical model				Simulation model				
Model	$a(h^{-1})$	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	L_o	t_{oc} (seconds)	ρ_{ws} (percent)	
M_1	221.76	90.7	13.11	418.9	92.4	88.5	7.11	316.5	92.4	
M_2	221.76	82.0	6.95	299.1	92.4	80.1	4.80	259.8	92.5	
$M_3(p = 0.5)$	110.88	90.7	13.11	837.8	92.4	88.7	8.50	678.7	92.3	
$M_4(p = 0.5)$	110.88	82.0	6.95	598.2	92.4	80.6	5.56	546.4	92.4	
$M_3(p = 0.2)$	44.35	90.7	13.11	2094.6	92.4	88.7	9.14	1750.0	92.3	
$M_4(p = 0.2)$	44.35	82.0	6.95	1495.5	92.4	80.7	5.80	1386.2	92.4	

Table 9
Experiment 2, maximum throughput per hour.

Variant	# locations	R = 2		R = 8		R = 14	
		No zones	Zones	No zones	Zones	No zones	Zones
2 by 88 aisles	1780	73.5	105.3	291.6	412.5	504.3	693.8
4 by 44 aisles	1800	130.6	181.4	508.2	676.7	840.4	1014.6
6 by 30 aisles	1860	170.9	233.0	649.3	827.0	1011.2	1127.3
8 by 22 aisles	1840	204.2	274.4	755.9	928.3	1103.5	1172.4
12 by 14 aisles	1800	245.1	325.3	871.6	1029.0	1165.2	1193.9
14 by 12 aisles	1820	253.5	336.2	893.2	1048.9	1172.5	1196.3
16 by 10 aisles	1760	262.8	348.5	915.4	1068.1	1178.6	1197.7
20 by 8 aisles	1800	262.5	350.5	914.4	1072.5	1177.9	1198.1
26 by 6 aisles	1820	253.7	345.3	891.8	1066.4	1170.3	1197.9
36 by 4 aisles	1800	229.5	321.5	826.7	1029.9	1141.2	1195.4
60 by 2 aisles	1800	174.6	257.3	659.3	898.0	1013.0	1169.0



12 aisles by 14 cross-aisles



36 aisles by 4 cross-aisles

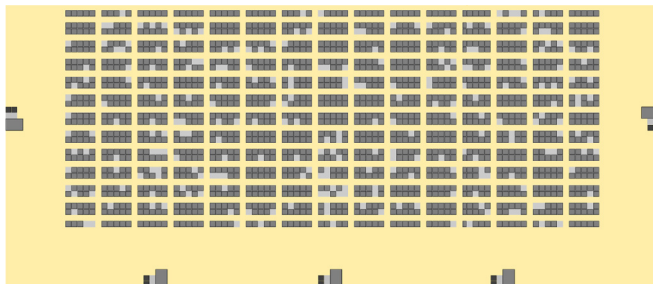
Fig. 11. Variants with different length-to-width ratios.

of the robots and the workstation from the analytical method is nearly the same as for the simulation. The mean length of the external order queue does differ between the analytical method and the simulation and this affects the estimates of the order cycle time. The average order cycle time depends on the mean length of the external order queue and therefore also differs between the

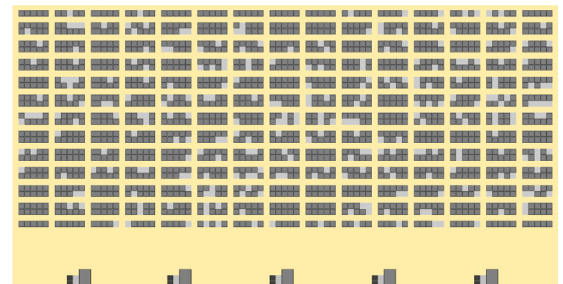
analytical method and the simulation. However, the differences are relatively small for the average order cycle time. The estimates of the analytical method typically stay below 10 percent of the estimates of the simulation, except for high arrival rate when $R = 2$. As is evident from the tables, using zones lowers the robot utilization and order cycle time.

Table 10
Experiment 3, maximum throughput per hour.

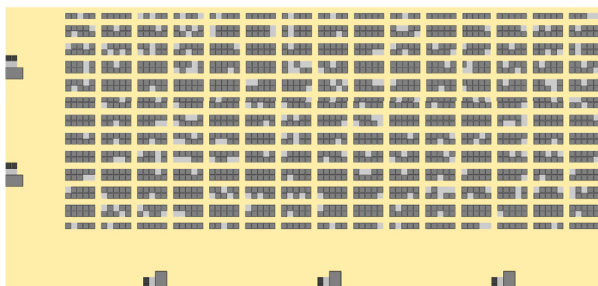
Variant	R = 2		R = 8		n = 14	
	No zones	Zones	No zones	Zones	No zones	Zones
Configuration 0, 0, 3, 2	252.0	296.3	889.2	986.1	1171.2	1191.1
Configuration 0, 0, 4, 1	254.2	299.5	893.8	990.6	1171.6	1191.0
Configuration 0, 0, 5, 0	251.1	295.3	886.6	983.0	1169.6	1190.2
Configuration 1, 0, 2, 2	249.8	312.5	884.0	1011.7	1169.8	1193.4
Configuration 1, 0, 3, 1	251.7	315.2	887.9	1015.3	1170.1	1193.3
Configuration 1, 0, 4, 0	247.6	309.6	877.6	1004.3	1166.7	1191.7
Configuration 1, 1, 2, 1	249.5	331.4	882.7	1040.9	1168.7	1195.6
Configuration 1, 1, 3, 0	245.1	325.3	871.6	1029.0	1165.2	1193.9
Configuration 2, 0, 2, 1	247.7	326.9	877.8	1035.9	1166.7	1195.4
Configuration 2, 0, 3, 0	243.3	320.9	866.6	1023.9	1163.1	1193.8
Configuration 2, 1, 1, 1	247.2	345.5	875.9	1064.5	1165.4	1197.5
Configuration 2, 1, 2, 0	241.1	337.0	861.5	1049.5	1161.8	1196.1
Configuration 2, 2, 1, 0	238.8	351.2	854.7	1073.1	1158.4	1198.0
Configuration 3, 0, 1, 1	245.4	341.4	871.0	1059.3	1163.2	1197.3
Configuration 3, 0, 2, 0	239.3	332.8	856.6	1044.3	1159.5	1195.9
Configuration 3, 1, 1, 0	238.8	351.5	854.8	1072.9	1158.3	1198.0
Configuration 3, 2, 0, 0	230.4	357.1	833.6	1081.5	1151.3	1198.5
Configuration 4, 0, 1, 0	237.6	348.6	851.5	1069.5	1156.8	1197.9
Configuration 4, 1, 0, 0	231.1	358.7	835.3	1083.1	1151.9	1198.5
Configuration 5, 0, 0, 0	229.9	355.9	832.0	1079.7	1150.5	1198.4



Configuration 1,1,3,0



Configuration 0,0,5,0



Configuration 2,0,3,0



Configuration 3,2,0,0

Fig. 12. Workstation configurations.

For the picker time, the model can also handle Cox- k distributions with cv^2 lower than 1, but then discrepancies arise between the analytical and the simulation results, even for robot utilization. Table 8 shows the results for 14 robots with a high arrival rate, where $cv^2 = 0.6$. In other words, choosing a cv^2 different from 1 is possible, but the analytical results will no longer be fully reliable. The models M_3 and M_4 assume that processing of multi-line orders happens sequentially, whereas in practice this may happen simultaneously. In Appendix C simulations are shown where sequential and simultaneous processing of multi-line orders are compared.

4.2. Experiment 2: varying the length-to-width ratio of the storage area

Table 9 shows the maximum throughput in orders per hour of all the workstations together. In a system that achieves the maximum throughput, robots do not have to wait for an order and hence a synchronization queue is not needed. For the situation without zones, a CQN is created by removing the synchronization queue from model M_1 and similarly for the situation with zones, a CQN was created based on model M_2 . The throughput per workstation was calculated by applying the single class AMVA method

in Appendix A to these CQNs. The number of aisles and of cross-aisles has to be even in each of the variants, because each (cross-)aisle has a single travel direction. For each aisle going west there must also be one going east and for each cross-aisle going north there must be one going south. The variants were chosen such that the number of storage locations was never more than 4 percent from 1800. As can be seen from Table 9, the result is relatively insensitive for the length-to-width ratio unless the ratio becomes unbalanced by a factor of more than 3 or 4 (Fig. 11). Using zones increases the maximum throughput by almost 50 percent. The length-to-width ratio can be measured in aisles or in meters. Since a block of storage locations measures 2 storage locations by 5 storage locations, a layout that has x aisles by y cross aisles has a storage area of $3x$ by $6y + 5$ meters.

4.3. Experiment 3: varying the location of workstations

Table 10 shows the maximum throughput in orders per hour of all the workstations together. The throughput per workstation was calculated by applying the single class AMVA method described in Appendix A to the same CQNs as in experiment 2. The configurations are named as “Configuration x_1, x_2, x_3, x_4 ” where x_1 denotes the number of workstations located west of the storage area, x_2 the number of workstations located east of the storage area, x_3 the number of workstations located south of the storage area and x_4 the number of workstations located north of the storage area, see also Fig. 12 for some examples. As can be seen from Table 10, the maximum throughput is sensitive to the location of the workstations. Interestingly, the results with zones are very different from the results without zones. Without zones, the maximum throughput tends to be higher if the workstations are located north and south of the storage area but if zones are present the maximum throughput tends to be higher if workstation are located west and east of the storage area. The difference between the best and worst workstation configuration is also higher if zones are present. The explanation is that without zones the average travel distance is shorter for workstations north and south of the storage area, whereas with zones the average travel distance is shorter for zones west and east of the storage area.

5. Conclusions

The main contribution of this paper is that it is one of the first to model Robotic Mobile Fulfillment System, and includes accurate driving behavior of robots, and multi-line orders. This paper develops queueing models to analyze an RMFS with and without zones and with single-line and multi-line orders, and it shows how to derive analytical expressions for distributions of the robot travel times. The aim was to gain insights for system design by measuring maximum order throughput, robot utilization, and order cycle time. The first experiment shows that the analytical method accurately estimates robot utilization, workstation utilization, and average order cycle time. The second experiment indicates that the maximum throughput is insensitive to the length-to-width ratio of the storage area, except if this ratio becomes strongly skewed. The last experiment shows that the location of workstations around the storage area matters. If storage zones are used, maximum throughput tends to be higher if the workstations are located west and east of the storage area, whereas without storage, it tends to be higher if workstations are located north and south of the storage area.

Two limitations of this study are that congestion and robot switching between workstations have not been included in the model. In practice however, congestion should only have a small effect on the order throughput and workstation utilization, since the system is designed with the aim that the picker is kept busy

to achieve high pick rates. Congestion may then cause a robot to enter the queue at the workstation a little later but the bottleneck in the system is the picker and not transportation. Also, congestion is unlikely to happen, as the number of robots is typically small relative to the space in which they drive and robots can travel underneath the racks when they are not carrying a pod. This makes it unlikely that the robots run into multiple other robots in the adjacent space around them. Robot switching could be beneficial if the workstations have a low utilization. However, this paper focuses on design aspects, whereas robot switching is connected with operational decisions. For example, the robot may have stored the pod at a particular location, because it is then close to another pod that needs to go to that same workstation. The robot may also have stored the pod close to a workstation that needs products from it soon. In other words, robot switching may bring operational benefits and would be interesting to explore in future research on operational decisions. In addition, Enright and Wurman (2011) mention several operational problems that have not yet been solved for an RMFS. As they show, Robotic Mobile Fulfillment Systems still pose many challenging problems and contain interesting, unexplored avenues.

Acknowledgments

The authors are thankful to the editor and the reviewers for their detailed suggestions, which helped to improve the exposition of this paper.

Appendix A. AMVA algorithm

This appendix shows the single class AMVA algorithm used for evaluating a CQN. It is the AMVA algorithm from Appendix A.2 in Buitenhek et al. (2000), and it has not been adapted except for step 3g where the queue lengths are calculated including the robots in service. The Infinite Servers stations are modeled by setting the number of servers c_s equal to the number of robots R . The notation is explained in Table A.11. Visit ratios are calculated as explained in Bolch et al. (2006).

Step 1: Initialize:

$$p_s(0 | 0) = 1, \quad s = 1, \dots, S \tag{A.1}$$

$$Q_s(0) = 0, \quad s = 1, \dots, S \tag{A.2}$$

$$L_s(0) = 0, \quad s = 1, \dots, S \tag{A.3}$$

Table A1
Notation used in the AMVA.

Symbol	Meaning
S	The total number of stations
R	The total number of robots
$ES_{rem,s}$	The expected time remaining until the first departure at station s
ES_s	The first moment of the service time of station s
ES_s^2	The second moment of the service time of station s
$L_s(r)$	The expected robot queue length including robots in service at station s when the system contains r robots
$\tilde{L}_s(r)$	The expected queue length excluding robots in service at station s when the system contains r robots
$Q_s(r)$	The probability that all servers are busy at station s when the system contains r robots
$p_s(i r)$	The probability that there are i robots at station s when the system contains r robots
$\tau(r)$	The throughput when the system contains r robots
$ET_s(r)$	The lead time at station s when the system contains r robots
c_s	The number of servers at station s
ν_s	The visit ratio of station s

Table B1

Average robot speed for various layouts, with acceleration 0.75 meters per second square and maximum speed 1.5 meters per second.

# aisles	# cross-aisles	Average robot speed
2	88	1.427
4	44	1.380
6	30	1.362
8	22	1.336
12	14	1.316
14	12	1.321
16	10	1.310
20	8	1.314
26	6	1.330
36	4	1.332
60	2	1.365

$$\tilde{L}_s(0) = 0, \quad s = 1, \dots, S \tag{A.4}$$

Step 2: Preprocessing. For $s = 1, \dots, S$

$$ES_{rem,s} = \frac{c_s - 1}{c_s + 1} \frac{ES_s}{c_s} + \frac{2}{c_s + 1} \frac{1}{c_s} \frac{ES_s^2}{2ES_s} \tag{A.5}$$

Step 3: Iteration. For $r = 1, \dots, R$

(a) For $s = 1, \dots, S$

$$ET_s(r) = Q_s(r-1)ES_{rem,s} + \tilde{L}_s(r-1) \frac{ES_s}{c_s} + ES_s \tag{A.6}$$

(b)

$$\tau(r) = \frac{r}{\sum_{s=1}^S v_s ET_s(r)} \tag{A.7}$$

(c) For $s = 1, \dots, S$ and for $b = 1, \dots, \min(c_s - 1, r)$

$$p_s(b | r) = \frac{ES_s}{b} v_s \tau(r) p_s(b-1 | r-1) \tag{A.8}$$

(d) For $s = 1, \dots, S$, if $r < c_s$, $Q_s(r) = 0$, otherwise,

$$Q_s(r) = \frac{ES_s}{c_s} v_s \tau(r) [Q_s(r-1) + p_s(c_s - 1 | r-1)] \tag{A.9}$$

(e) For $s = 1, \dots, S$

$$p_s(0 | r) = 1 - \sum_{b=1}^{\min(c_s-1,r)} p_s(b | r) - Q_s(r) \tag{A.10}$$

(f) For $s = 1, \dots, S$, if $r < c_s$, $\tilde{L}_s(r) = 0$, otherwise,

$$\tilde{L}_s(r) = \frac{ES_s}{c_s} v_s \tau(r) [\tilde{L}_s(r-1) + Q_s(r-1)] \tag{A.11}$$

(g) For $s = 1, \dots, S$

$$L_s(r) = \tau(r) v_s ET_s(r) \tag{A.12}$$

Table C1

experiment with sequential and simultaneous processing of multi-line orders.

	$a(h^{-1})$	No zones			Zones		
		ρ_r (percent)	t_{oc} (seconds)	ρ_{ws} (percent)	ρ_r (percent)	t_{oc} (seconds)	ρ_{ws} (percent)
seq. $R = 14, p = 0.5$	110.88	0.914	838.370	0.923	0.825	595.950	0.923
sim. $R = 14, p = 0.5$	110.88	0.933	787.952	0.927	0.864	502.530	0.923
seq. $R = 14, p = 0.2$	44.35	0.910	2006.742	0.921	0.821	1420.379	0.923
sim. $R = 14, p = 0.2$	44.35	0.935	1564.767	0.918	0.895	1079.375	0.922
seq. $R = 14, p = 0.5$	50.40	0.314	313.167	0.421	0.241	242.153	0.418
sim. $R = 14, p = 0.5$	50.40	0.340	183.844	0.421	0.274	147.588	0.421
seq. $R = 14, p = 0.2$	20.16	0.314	779.882	0.421	0.244	608.381	0.423
sim. $R = 14, p = 0.2$	20.16	0.386	261.163	0.423	0.338	222.834	0.426

Appendix B. Robot speed

Given an acceleration and maximum speed, the average robot speed for a layout can be calculated as follows. Each route consists of straight linear segments that are connected by angles of 90 degrees. Each time a robot turns it starts with a speed of zero and increases speed until it hits the maximum speed or until it is halfway. Then the robot decreases the speed until it is zero again and it turns to go on the next segment. The overall average robot speed is then calculated by averaging across all routes. For the standard layout it was found that an average robot speed of 1.3 meters per second, as mentioned in [Wulfraat \(2012\)](#), corresponds to an acceleration of 0.75 meters per second square and a maximum speed of 1.5 meters per second, which seems realistic. In [Table B1](#) the average robot speed is shown for layouts with varying number of aisles and cross-aisles, using this acceleration and maximum speed. As can be seen, the average robot speed stays roughly between 1.3 and 1.4 meters per second, even as the length-width ratio changes.

Appendix C. Multi-line orders

[Table C1](#) shows the results for the standard layout when multi-line orders are processed sequentially or simultaneously. Sequential processing means that only one robot retrieves all the pods for all the order lines of an order and is indicated in [Table C1](#) with “seq.”. Simultaneous, indicated with “sim.”, means an order can be processed in parallel and that multiple robots can fetch pods for the same order. The parameter p is the parameter for the geometric distribution of the number of order lines, i.e. $p = 0.2$ means that orders have on average 5 order lines. Sequential processing can be modeled analytically by models M_3 and M_4 , but for simultaneous processing there is no analytical model, therefore the results were generated by simulation. The results indicate that robot utilization is higher and order cycle time lower with simultaneous processing. However, the workstation utilization remains almost the same under both forms of processing.

References

Bolch, G., Greiner, S., de Meer, H., & Trivedi, K. S. (2006). *Queueing networks and Markov chains*. Wiley Publishing Inc.

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

Business Wire (2015). *Amazon unveils its eighth generation fulfillment center*. www.businesswire.com/multimedia/home/20141130005031/en. Accessed 14.07.16.

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

Enright, J. J., & Wurman, P. R. (2011). Optimization and Coordinated Autonomy in Mobile Fulfillment Systems. In *Automated action planning for autonomous mobile robots* (pp. 33–38).

Fukunari, M., & Malmborg, C. J. (2009). A network queueing approach for evaluation of performance measures in autonomous vehicle storage and retrieval systems. *European Journal of Operational Research*, 193, 152–167.

- Heragu, S. S., Cai, X., Krishnamurthy, A., & Malmberg, C. J. (2011). Analytical models for analysis of automated warehouse material handling systems. *International Journal of Production Research*, 49(22), 6833–6861.
- Kuo, P.-H., Krishnamurthy, A., & Malmberg, 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.
- Lu, W., McFarlane, D., Giannikas, V., & Zhang, Q. (2016). An algorithm for dynamic order-picking in warehouse operations. *European Journal of Operational Research*, 248, 107–122.
- Marchet, G., Melacini, M., Perotti, S., & 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.
- Roy, D., Krishnamurthy, A., Heragu, S. S., & Malmberg, C. J. (2012). Performance analysis and design trade-offs in warehouses with autonomous vehicle technology. *IIE Transactions*, 44, 1045–1060.
- Roy, D., Krishnamurthy, A., Heragu, S. S., & Malmberg, C. J. (2013). Blocking effects in warehouse systems with autonomous vehicles. *IEEE Transactions on Automation Science and Engineering*, 99, 1–13.
- Roy, D., Krishnamurthy, A., Heragu, S. S., & Malmberg, 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.
- Roy, D., Krishnamurthy, A., Heragu, S. S., & Malmberg, C. J. (2015b). Stochastic models for unit-load operations in warehouse systems with autonomous vehicles. *Annals of Operations Research*, 231, 129–155.
- Roy, D., Krishnamurthy, A., Heragu, S. S., & Malmberg, 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.
- Roy, D., Nigam, S., Adan, I. J. B. F., de Koster, M. B. M., & Resing, J. (2014). Mobile fulfillment systems: Model and design insights. Working paper.
- Schleyer, M., & Gue, K. R. (2012). Throughput time distribution analysis for a one-block warehouse. *Transportation Research Part E*, 48, 652–666.
- Tappia, E., Roy, D., De Koster, M. B. M., & Melacini, M. (2016). Modeling, analysis, and designinsights for shuttle-based compact storage systems. *Transportation Science*. Forthcoming.
- Wulfraat, M. (2012). *Is Kiva systems a good fit for your distribution center? An unbiased distribution consultant evaluation*. http://www.mwvpl.com/html/kiva_systems.html. Accessed 14.07.16.
- Wurman, P. R., D'Andrea, R., & Mountz, M. (2008). Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine*, 29(1), 9–19.