

Robot-storage zone assignment strategies in mobile fulfillment systems

Debjit Roy^{a,*}, Shobhit Nigam^b, René de Koster^c, Ivo Adan^d, Jacques Resing^e

^a Production and Quantitative Methods Area, Indian Institute of Management Ahmedabad, Gujarat, India

^b Department of Mathematics, Pandit Deendayal Petroleum University, Gandhinagar, Gujarat, India

^c Department of Technology and Operations Management, Rotterdam School of Management, Erasmus University, The Netherlands

^d Department of Industrial Engineering, Technische Universiteit Eindhoven, The Netherlands

^e Department of Mathematics and Computing Science, Technische Universiteit Eindhoven, The Netherlands



ARTICLE INFO

Keywords:

Mobile shelves
Robot assignment
Queueing models
Design insights

ABSTRACT

The robotic mobile fulfillment system (MFS) is widely used for automating storage pick and pack activities in e-commerce distribution centers. In this system, the items are stored on movable storage shelves, also known as inventory pods, and brought to the order pick stations by robotic drive units. We develop stylized performance evaluation models to analyze both order picking and replenishment processes in a mobile fulfillment system storage zone, based on multi-class closed queueing network models. To analyze robot assignment strategies for multiple storage zones, we develop a two-stage stochastic model. For a single storage zone, we compare dedicated and pooled robot systems for pod retrieval and replenishment. For multiple storage zones, we also analyze the effect of assigning robots to least congested zones on system throughput in comparison to random zone assignment. The models are validated using detailed simulations. For single zones, the expected throughput time for order picking reduces to one-third of its initial value by using pooled robots instead of dedicated robots; however, the expected replenishment time estimate increases up to three times. For multiple zones, we find that robots that are assigned to storage zones with dedicated and shortest queues provide a greater throughput than robots assigned at random to the zones.

1. Introduction

Many big E-Commerce retailers are increasingly employing warehouse automation technologies to reduce operational costs, increase customer satisfaction and improve operational efficiency and productivity. A robotic mobile fulfillment system – a parts-to-picker order pick system – is a new paradigm for automating pick, pack and ship activities in distribution centers that significantly improves worker productivity and throughput capacity. The mobile fulfillment system (MFS), based on mobile-rack technology, was pioneered by Kiva Systems, now Amazon Robotics ([Wurman et al., 2008](#); [D'Andrea and Wurman, 2008](#); [Mountz, 2012](#); [Azadeh et al., 2018](#); [Boysen et al., 2018](#)). Today, several other similar robotic mobile fulfillment systems exist, such as the CarryPick (Swisslog), Butler (GreyOrange), Racrew (Hitachi), and Suning. Further, other variants of mobile robotic systems such as Ocado, Locus robotics systems, Fetch robotics, and the Perfect pick system can be found in practice (see [Azadeh et al., 2018](#)). In MFS, the items are stored on movable storage shelves in cartons, totes, or pallets (also referred as the inventory pods, see Fig. 1a and b). The search and retrieval of

* Corresponding author.

E-mail address: debjit@iima.ac.in (D. Roy).

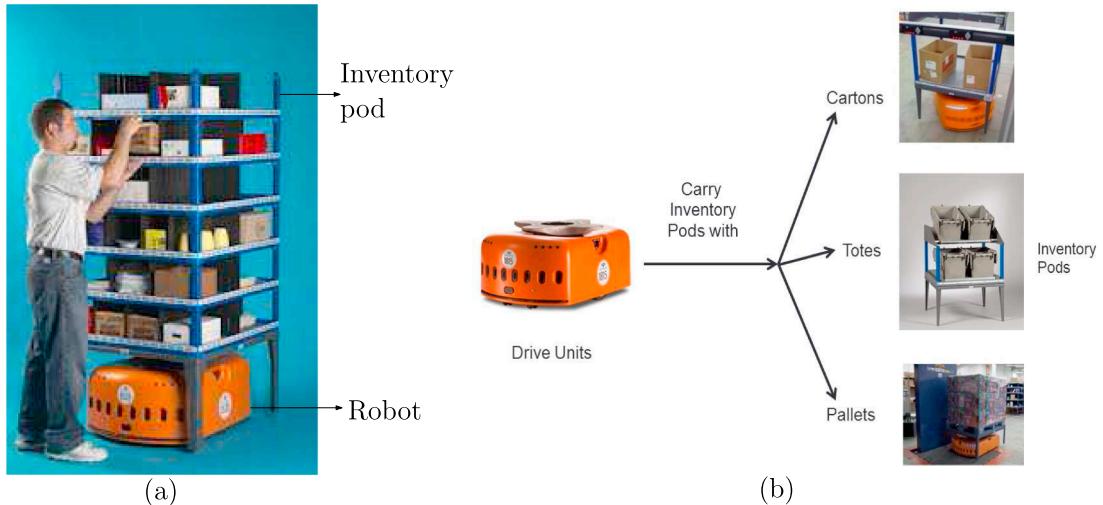


Fig. 1. (a) An inventory pod being carried by an Amazon Robotics' pick bot (D'Andrea, 2011), and (b) Variants of inventory pods with different load types that can be picked by the drive unit (source: <https://www.amazonrobotics.com>).

the inventory pods are performed by small, autonomous drive-units (also referred as robots). The MFS is depicted in Fig. 2.

Up to 55% of the operating costs at a distribution center are due to its order pick costs, which include costs associated with item picking, consolidation, and order packing (De Koster et al., 2007). The traditional part-to-picker order pick systems such as the miniload AS/RS has three primary inefficiencies: (1) a sequential order picking flow (only one item can be picked simultaneously in an aisle); (2) not easily scalable and (3) inability to handle peak demands. These inefficiencies, which result in delivery time delays and high operating costs, can be potentially overcome by an MFS. In an MFS, movable storage shelves, which contain items ordered by a customer, are brought automatically to an order picker. Due to the large number of robots which can transport many item-pods simultaneously, an order picker can complete the order in a shorter time and can complete more orders in a day compared to traditional picker-to-parts systems (Enright and Wurman, 2011). An MFS provides added benefits of flexibility and scalability in addition to the associated advantages of automation. Most e-retailers have massive peaks in their distribution shipping volumes (during special occasions such as Thanksgiving or Christmas). By adding more robots, pods and/or work stations, the throughput capacity of handling additional orders can be addressed economically and in a relatively short time span (http://www.mwpv.com/html/kiva_systems.html).

Fig. 3 shows typical flows within a mobile fulfillment system (Enright and Wurman, 2011). In this configuration, the picker prepares the shipping cartons and transfers them to the pack station using conveyors. The flow of the pod movements during the order picking and replenishment process is shown using solid and dotted lines, respectively. The location of a pod (carrying a



Fig. 2. Warehouse system from Amazon Robotics (source: D'Andrea, 2011).

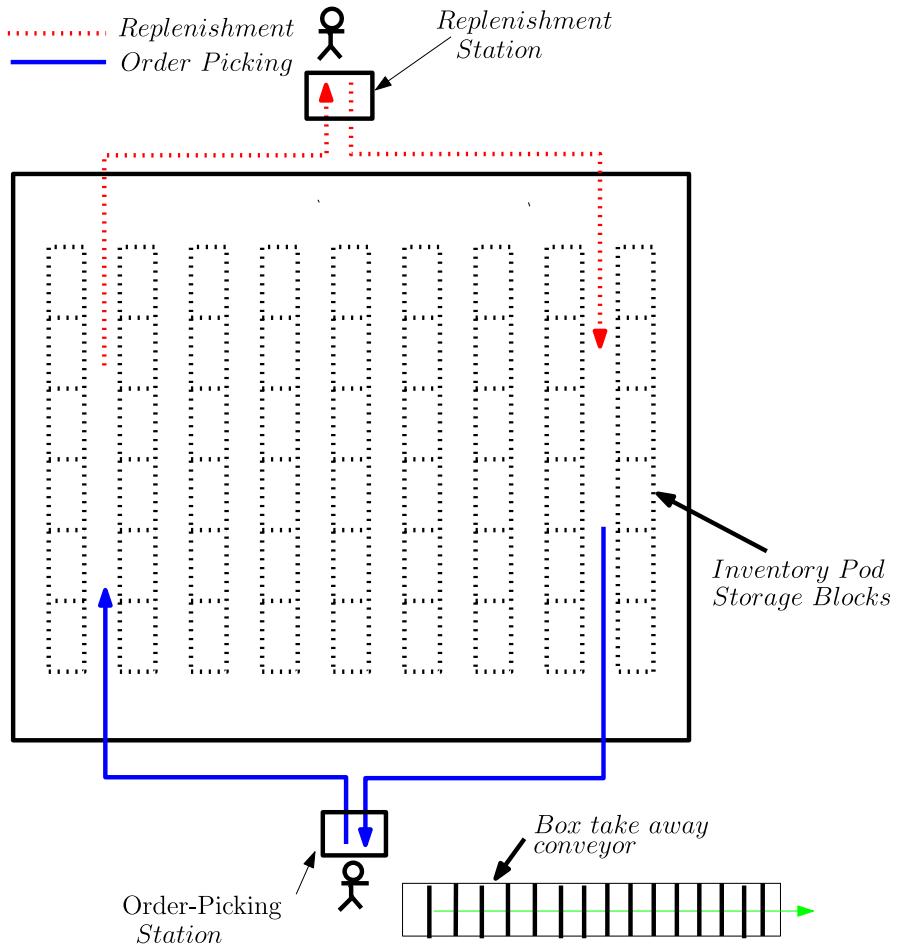


Fig. 3. Typical configuration of a mobile fulfillment system (adapted from Wurman et al. (2008)).

particular item) is dynamic and is determined by the item turnover. The most frequently used items are stocked closer to the order pick station.

Several operational decisions affect the system throughput performance. For instance, operational decision questions such as the choice of pod storage location, the choice of order assignment to the order pick stations, and the choice of the robot to fetch the pod, affect the throughput performance. Likewise, the robot assignment strategies such as maintaining a dedicated vs. a pooled fleet of robots in a storage zone for order pick and replenishment processes, and assignment of robots to a least congested vs. random storage zone may affect the system throughput performance. Robots may be pooled to perform both order pick and replenishment processes, where the robots complete an order pick operation followed by replenishment (if needed), in a so-called dual-command operation. If robots are shared between the order-picking and replenishment processes, the flexibility of robot allocation between the processes may reduce the waiting times in accessing an idle robot. However, this decrease in wait time may be accompanied with an increase in additional robot travel time due to combination of order picking and replenishment processes. It is not clear whether a dedicated fleet of robots performing single-command cycle operations (either order picking or replenishment) provides better throughput performance. Likewise, in the case of multiple storage zones, dedicated robots working in a storage zone may result in imbalance of workload among the zones and congested travel paths in high workload zones. Assignment of robots to the least congested storage zone may seem to balance congestion among zones albeit with additional robot travel times. Hence, it is not clear if the benefits of reduced congestion within a zone are lost with additional robot travel between the zones. In the dedicated storage zone, a specific number of robots is confined to a particular storage zone to process transactions. In least congested zones, the robots are shared between the zones, but a robot accesses the zone that has the least number of active robots at that point in time. Note that during busy periods, transactions are always waiting to be processed in both zones. Hence, accessing the least congested zone may improve the system throughput.

In this research, we specifically investigate the performance of robot assignment strategies to the storage zones with a random open location pod storage policy. The random open location storage is an efficient and commonly used storage policy, where any pod (after an order pick) is equally likely to be stored in any of the open locations. A prime advantage of this policy is that it maximizes space utilization.

We address the following research questions:

- For a single storage zone, do dedicated robots in order picking and replenishment (single command cycles) perform better than pooled robots, which are shared between order picking and replenishment?
- For multiple storage zones (with identical copies of SKU pods), is assigning robots to the least congested zone beneficial over assignment of robots to dedicated storage zones?

The MFS is modeled as a closed queueing network with realistic robot movements. When carrying a pod, the robots move only along the aisles and cross-aisles present in the order pick area, whereas the empty robots can also travel underneath the pods to access another aisle. The model development follows a two-stage approach: (1) *Parameter estimation*: Markov-chain based models are developed to analyze the aisle travel time with random open storage location assignment. Probabilistic travel time expressions are developed to estimate the robot travel time underneath the pods. (2) *Performance analysis*: Customized queueing network models are developed to analyze order picking and replenishment processes, and to answer both research questions. To demonstrate the robot movements within a storage zone, we first develop a single-class closed queueing network model. To analyze the effect of robot allocation (pooled vs dedicated), we develop two queueing network models with multiple robot classes. In the first multi-class closed queueing network model, the robots are pooled and they switch classes based on transaction probabilities (order picking vs replenishment) whereas in the second multi-class model, two dedicated fleets are used, one for order picking and another for replenishment. The model also captures the interference delays that could potentially occur in the aisles (if multiple robots attempt to access the aisle at the same time). We use a collision-avoidance protocol for aisle movement. To analyze the robot assignment strategies for multiple storage zones, we adopt network reduction techniques in combination with continuous-time Markov chain analysis.

The models are evaluated using the approximate mean value analysis algorithm and validated with simulation. The analytical models present an attractive alternative to simulation for optimizing design parameters and improving system performance measures. The performance measures obtained from the models include robot utilization, system throughput and expected throughput time for order picking and replenishment.

The remainder of this paper is organized as follows. Section 2 reviews literature on order pick systems. Section 3 describes the system operations and the modeling assumptions used in this paper to develop the queueing network model. In Section 4, we describe the system and modeling assumptions. The queueing network models developed for the MFS with various layout configurations are presented in Section 5. The numerical results and the insights obtained from the analytical models are discussed in Section 6. Section 7 reports our conclusions and provides directions for future research.

2. Literature review

In this section, we review literature in the area of parts-to-picker order pick systems with a specific focus on MFS and vehicle-based warehouse systems. In all cases, we restrict our review to travel-time and performance evaluation models, which is also the scope of our study.

Order picking is the most labour-intensive operation in warehouses and a very capital-intensive operation in warehouses with automated systems (De Koster et al., 2007). Several travel time models for order picking systems that consider a specific equipment, storage policy and order picking area layout combinations are proposed in the literature. Daniels et al. (1998) develop an order picking model in which they determine the assignment of inventory to an order and the associated sequencing decisions in which the selected locations are visited. A detailed literature overview on typical design and control problems such as optimal layout design, storage assignment methods, routing methods, order batching and zoning in manual order picking processes is provided by De Koster et al. (2007).

Several different parts-to-picker systems can be distinguished such as the miniload AS/RS, MFS, and vehicle-based tote handling systems (for an overview of such systems, see Azadeh et al., 2018). Our work is motivated by the robotic MFS in which movable storage shelves can be lifted by small autonomous robots, see Enright and Wurman (2011), D'Andrea (2012). A detailed description of the robotic MFS is given in Wulfaat (2012).

Research on analyzing performance of the MFS is limited. Using semi-open queueing network models, Lamballais et al. (2017) develop an analytical model of the MFS with only order picking and show that the maximum order throughput is quite insensitive to the length-to-width ratio of the storage area. They also show that maximum order throughput is affected by the location of the workstations around the storage area. Yuan and Gong (2017) develop an open queueing network to jointly estimate the optimal number of the robots and their required average velocity to achieve a certain throughput time. Zou et al. (2017) propose a workstation assignment rule based on handling speeds of workstations and design a neighbourhood search algorithm to find a near optimal assignment rule. They also show that handling-speeds-based assignment rule significantly outperforms the random assignment rule when the workers have large handling time difference. Using semi-open queueing network models, Zou et al. (2018) show that throughput time performance can be significantly affected by the battery charging policies. They show that inductive charging performs best, and that battery swapping outperforms plug-in charging by about 5% in terms of retrieval transaction throughput time. Lamballais et al. (2017a) develop an analytical model to optimize the inventory allocation across the pods. They find that spreading inventory units across multiple pods is a better allocation strategy for e-commerce order fulfillment. Boysen et al. (2017) jointly optimize the order pick process and the corresponding pod sequence at the MFS pick workstations. They show that with a better pick sequence, the number of robots can be reduced by 50%. Using a fluid model, Yuan et al. (2018) analyze the impact of

velocity-based storage policies and show that a two or three class-based pod storage policy reduces the travel distance by 8–10%.

There are several studies on performance analysis of autonomous vehicle-based warehouse systems that closely resemble MFSs in terms of robot aisle and cross-aisle movement. For instance, some papers discuss the estimation of cycle times for autonomous devices in Autonomous Vehicle-based Storage and Retrieval Systems (AVS/RS). An AVS/RS relies on autonomous vehicles to provide horizontal movement within a tier and uses lifts to provide vertical movement between tiers. [Malmborg \(2003\)](#) and [Malmborg \(2003\)](#) propose cycle-time estimation models based on Markov Chains. [Fukunari and Malmborg \(2008\)](#) develop an iterative computation algorithm based on queueing approximations to estimate the cycle times in AVS/RS. [Fukunari and Malmborg \(2009\)](#) extend the previous work by developing a queueing network approach to estimate the proportion of single and dual command cycles in the system. [Kuo et al. \(2007\)](#) model the movement of autonomous vehicles as an $M/G/V$ queue nested within an $M/G/L$ queue for estimating the expected travel times of vehicles and lifts. In such an AVS/RS, vehicle blocking delays in the aisles and cross-aisles can significantly impact system throughput and transaction cycle times. [Roy et al. \(2014\)](#) develop protocols to model vehicle blocking using a queueing network model and evaluate design trade-offs. A comprehensive review of collision prevention strategies in AGV systems can be found in [Le-Anh and De Koster \(2006\)](#).

[Roy et al. \(2012\)](#) develop a multi-class semi-open queueing network model to investigate several design decisions such as the configurations of aisles and columns, allocation of resources to zones, and vehicles assignment rules that significantly affect the performance of a system. [Eken et al. \(2013\)](#) and [Cai et al. \(2013\)](#) develop semi-open queueing network models for analyzing the performance of the multi-tier AVS/RS with pooled vehicles. These models provide insights on the impact of tier configuration and sizing decisions on expected cycle times. However, note that existing studies on AVS/RS have been restricted to only pallet-based high density storage systems and not order pick systems.

Replenishment activities are very important to prevent shortage of products during intensive order picking ([De Koster et al., 2007](#)). However, current literature on vehicle-based storage and retrieval systems do not explicitly model replenishment strategies. The performance of replenishment activities becomes more frequent and crucial during a busy picking periods (for example, Thanksgiving and Christmas). To the best of our knowledge, this research is a first attempt to model a mobile-racks based order pick system and to answer the design tradeoffs by considering both order picking and replenishment activities. Since the robots perform both order-picking and replenishment, and these tasks can be done it parallel, the best robot assignment strategy for these processes are not clear. Further, the assignment of the robots based on the zone congestion levels is an interesting avenue for research. Taking inspiration from the existing literature, we investigate the effect of storage strategies and investigate the performance of dedicated and pooled robot configurations in the MFS. [Table 1](#) illustrates the positioning our study in MFS literature.

3. System operations and modeling assumptions

In this section, we first describe the operations and the modeling assumptions considered for an MFS consisting of a single storage zone. We then describe the protocol we develop for avoiding collisions of the robots in an aisle and extend the analysis to multiple zones.

3.1. System operations

The system consists of an even number of aisles A . This simplifies the analysis, but it is straightforward to adapt expressions for an odd number of aisles. We assume that an equal number of mobile shelves (pods) of identical dimensions are present along the two sides of an aisle. A single pod is assumed to be l meter long and w meter wide. The order pick area is D meters long and W meters wide. The warehouse shape factor is characterized by depth/width, which is the ratio between D and W . D proxies the number of aisles and W proxies the number of pod storage locations per aisle. The two terms, d_{arr}^{dep} and d_{arr}^{loc} denote the distance between the departure and the arrival paths, and the distance between the arrival path and the first bay location, respectively.

To answer our research questions, we consider two different layouts of a warehouse: layout 1 with only an order pick station, and layout 2 with both an order pick and a replenishment station (shown in [Fig. 4](#)).

Layout 1: We consider an order pick station located in the middle of the cross-aisle (in front of the aisles). This area is called the order pick area. A replenishment station is not considered in this layout. After completing an order pick, the robot first travels using

Table 1

Positioning of current research.

Citation	Process considered	Research objective	Model
Nigam et al. (2014)	Order pick	Class-based storage	Closed queueing network
Lamballais et al. (2017)	Order pick	Ratio of length-to width	Semi-open queueing network
Yuan and Gong (2017)	Order pick	Pooled vs dedicated robots	Open queueing network
Zou et al. (2017)	Order pick	Robot assignment to workstation	Semi-open queueing network
Boysen et al. (2017)	Order pick	Optimal pod sequence for pick	Optimization
Zou et al. (2018)	Order pick	Battery charging policies	Semi-open queueing network
Yuan et al. (2018)	Order pick	Class-based storage	Fluid
Our paper	Order pick and replenishment	Pooled vs dedicated robots; Robot assignment to pods	Closed queueing network

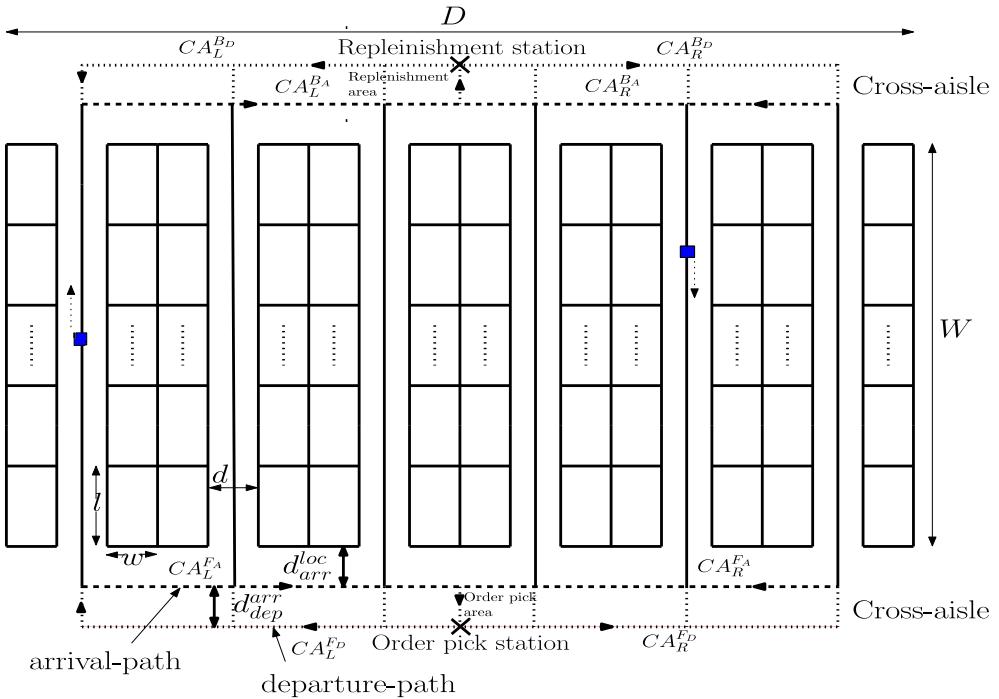


Fig. 4. Layout of a warehouse with both order pick and replenishment station.

the *departure path* of a cross-aisle and then travels along the center line of an aisle to store the pod in a pod storage location. Then the robot lifts a pod from another storage location from the same side of the aisle and brings it along the *arrival path* to the order pick station. Note that the robots can travel bidirectionally within an aisle. Since the aisles are short, we allow only one robot to access the aisle at a time. We analyze this layout using random pod storage strategy.

Layout 2: We consider two areas, an order pick area and a replenishment area in a warehouse where the order pick station is in the order pick area and the replenishment station is in the replenishment area (opposite of the order pick area), see Fig. 4. When the pod is emptied, a robot lifts the pod and travels to the replenishment station to replenish the pod. Using this layout, we later answer the trade-offs between usage of dedicated and pooled robots for order picking and replenishment.

A picker is stationed at the order pick station for picking items (filling orders) and a worker is stationed at the replenishment point for restocking the bins in the inventory pods. We assume that there is always a sufficient number of orders waiting to be processed by the robots, i.e. the robots never wait for order arrivals.

3.2. Assumptions

The following assumptions are made for the analysis of the MFS:

- The order pick station and the replenishment station have sufficient waiting space for robots (with pods).
- All storage locations and the pods have the same size. Therefore all storage locations are candidates for storing or retrieving any pod.
- Robots are scheduled according to the First-Come-First-Served (FCFS) policy.
- The travel velocity of a robot is constant and robot acceleration/deceleration effects are ignored.
- A dual-command cycle is considered for the storage and retrieval, i.e., a robot will perform a pod storage followed by a pod retrieval request in a single trip for both order picking and replenishment. This assumption is also observed in practice where the robots fetch a pick up pod and brings it to the pick station, and then waits with the pod until the pick has finished. Then it stores the pod and retrieves a next pod, if available. Since our goal is always to find the maximum system throughput, we assume that there is always an order (i.e. pod) waiting for picking.
- Each side of an aisle has at least one open location available for pod storage. Each robot stores a pod at the side of an aisle where the next retrieval has to be carried out. This assumption is also close to reality because these storage systems work with many open spaces (20–30%), see Fig. 2. We use this assumption to derive the aisle travel times with a random pod storage strategy. Again, this assumption is made to obtain the maximum throughput for the robotic system under idealistic conditions. Hence, our assumption fits well with both the modeling and design objectives of this paper.
- All items for filling an order are present in one inventory pod.
- There are always orders waiting to be served by the robot, i.e., the queue of orders is never empty.

The current model does not capture many operational details such as real-time coordination among the robots or the real-time collision-avoidance protocols. Detailed simulation-emulation models could capture the operational details quite well. The routing of the robots is described in Section 3.1, for two different layouts, and illustrated in Fig. 4. Although the robots take a shortest path between an origin and destination (taking into account the Manhattan topology) as sketched in Fig. 4, we model only limited routing choices of the robots using our analytical model. We attempt to model the random routing policy (i.e., location selection) using probabilities. We later use state-dependent routing to dispatch the robots into least congested zones. Note that we develop models to obtain the maximum throughput capacity for our system. Further, our model assumes that no two robots are allowed in the same aisle at the same time, which implies robots may have to wait before entering an aisle (see Section 3.3).

3.3. Aisle Protocol for the MFS

In the MFS, blocking can occur during the movements of a robot in the aisles. That is, a robot in the process of storing and retrieving the pod in an aisle could be blocked by another robot entering the same aisle. To avoid collisions or blocking during robot movements, it is assumed that a robot will only enter the aisle when the previous robot has completed its service in the aisle. Note that the aisles are typically narrow to allow only one robot movement in the aisle. The robots can also go underneath the racks when empty. These protocols also allow us to model robot movements between multiple zones.

4. Modeling the robotic MFS

Using a queueing model, we analyze the effect of multiple design parameters on system performance measures. In this section, we describe the nodes in the queueing models and the service time distributions at the nodes.

4.1. Network nodes

As discussed in Section 3 we consider two warehouse layouts. In the first layout, there is a departure and arrival path only in the order pick area. In the second layout, the departure and arrival paths are present in both the order pick (front) and the replenishment areas (back). The order pick (OP) station is located at the middle of the cross-aisle in the departure path. We divide the cross-aisle departure path into two equal segments (CA_L^{FD} and CA_R^{FD}) corresponding to the left and right segment of the cross-aisle in the order pick area). Likewise, the arrival path along the cross-aisle is divided into two equal segments (CA_L^{FA} and CA_R^{FA}) corresponding to the left and right segment of the cross-aisle in the arrival path). A similar cross-aisle segmentation is done for the departure and arrival paths at the replenishment station (CA_L^{BD} and CA_R^{BD} for the departure path and CA_L^{BA} and CA_R^{BA} for the arrival path). Each segment of the cross-aisle is modeled as an Infinite Server (IS) queue. A robot starts its service and accesses either the left or the right side of the segments on the departure path (with equal probability, 1/2). If A is the number of aisles, the robot then chooses any one of the aisles with equal probability $p = \frac{1}{A/2}$, and then accesses either pick face of the aisle and moves towards an open rack location to store a pod (based on the pod storage strategy). Next it moves towards a pick-up location, retrieves the pod and exits the aisle. As soon a robot exits the aisle it accesses the arrival path segment based on its current position from either the left or the right side of the cross aisle.

Based on the aisle protocol, only one robot can enter the aisle at a time; therefore, each aisle is modeled as a single server queue with an infinite buffer size. The picking and the replenishment stations are modeled as single server queues with service rates μ_{OP} and μ_R respectively. Since we assume that orders are always waiting to be served by the robot, we model the layouts 1 and 2 as closed queueing networks (see Fig. 5).

The stations in the network have general service times, which are characterized by the mean and the squared coefficient of variation (SCV).

We now present expressions for the first two moments of the service times describing the robot movements in the aisles and cross-aisles, and the fulfillment activities at the order pick and replenishment stations. To estimate the service times in aisles, we first determine the steady state probabilities of open locations at each side of an aisle for random pod storage strategy.

4.2. Aisle service time estimation

To obtain the mean and variance of the travel and handling time of a robot during storage and retrieval of a pod in an aisle, we describe the aisle by a discrete-time Markov chain. The status of storage locations in each side of an aisle can be either *open* or *used*. A location is denoted as open when there is no pod at that location and denoted as used when a pod is stored at that location. The state of an aisle can then be defined as a vector $s = (a_1, a_2, \dots, a_N)$, where a_k denotes the status of the storage location index k :

$$a_k = \begin{cases} 1, & \text{when location } k \text{ is used;} \\ 0, & \text{when location } k \text{ is open.} \end{cases} \quad (1)$$

Let N denote the total number of locations at each side of an aisle, out of which m locations are open. The state space S_m consists of all vectors s for which $a_1 + \dots + a_N = N - m$. Hence $M = \binom{N}{m}$ is the number of states in S_m .

Under the random open location storage strategy and assuming the robots do double cycles, exactly $m(N - m)$ states can be reached by a robot (during pod storage and retrieval), with equal probability, from every state $s \in S_m$. A state transition occurs when a robot stores a pod in an open location and retrieves a pod from an used location. Since for every state transition, the reverse

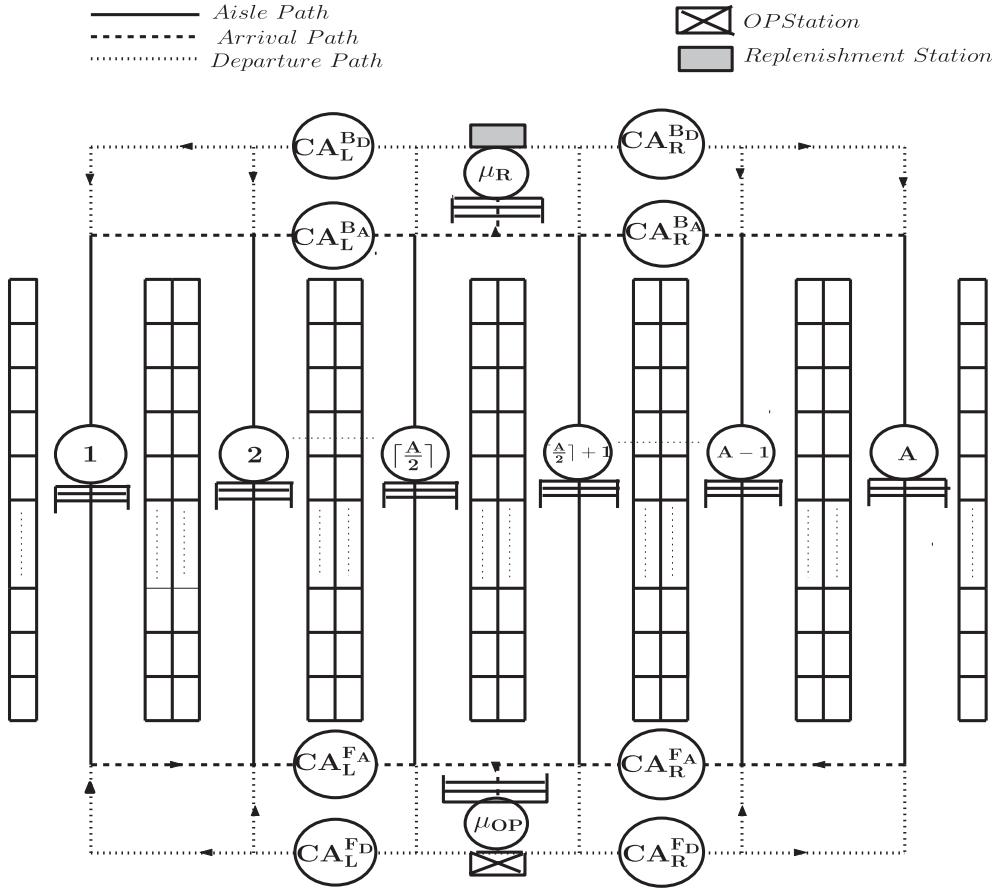


Fig. 5. Queueing model description with storage and retrieval from the same aisle.

transition is also feasible, we can immediately conclude that the Markov chain is double stochastic, and hence, the stationary distribution is uniform, i.e.,

$$\pi(s) = \frac{1}{M}, \quad \forall s \in S_m.$$

Note that this result also follows from symmetry.

4.3. Expectation and variance of the service time within an Aisle

In this section, we determine the average service time in an aisle for the random pod storage policy. The robot service time in aisles includes the total travel time required to store a pod and to retrieve another pod, and the pod handling times (pick-up and set-down times).

The service time in aisle, $t_{Aisle,R}$, for the random open location storage strategy will include the travel time of a robot to store a pod and retrieve another pod, including the return travel ($t_{WL,R}$), handling time to store a pod (t_{store}), handling time to retrieve a pod ($t_{retrieval}$) and the time associated to travel between the arrival-path (onward as well as return) of a cross-aisle to the starting point of the aisle location, d_{arr}^{loc} with a robot velocity v_r , which gives the following form:

$$t_{Aisle,R} = t_{WL,R} + t_{store} + t_{retrieval} + \frac{2d_{arr}^{loc}}{v_r} \quad (2)$$

Note that the travel time associated with the travel between the arrival and the departure paths is included in the cross-aisle service time (which will be explained later). Let i_s denote the location for storing a pod and i_r denote the location of retrieving a pod. Let $D_R(i_s, i_r)$ denote the distance travelled to store at location i_s and retrieve a pod from location i_r and return. In general, the distance travelled $D_R(i_s, i_r)$ for storing a pod at location i_s and retrieving a pod from location i_r for the random open location storage strategy, and returning back to the same cross-aisle is given by the following expression:

$$D_R(i_s, i_r) = 2li_m - l \quad (3)$$

where $i_m = \max(i_s, i_r)$.

From the Markov chain analysis (of the random open location storage strategy), we can conclude that both i_s and i_r are uniform on the N locations (*independent* of the number of open locations). Therefore, the joint probability distribution of i_s and i_r is expressed as follows:

$$P(i_s = k, i_r = l) = \frac{1}{N(N - 1)} \quad \forall k, l \in \{1, \dots, N\}, k \neq l \quad (4)$$

Using (4), the probability mass function, and the first and second moments of i_m are obtained using the following expressions.

$$P(i_m = k) = \frac{2(k - 1)}{N(N - 1)} \quad \text{where } k = 2, \dots, N \quad (5)$$

$$E[i_m] = \frac{2(N + 1)}{3} \quad (6)$$

$$E[i_m^2] = 2(N + 1) \left[\frac{N}{4} + \frac{1}{6} \right] \quad (7)$$

Using (3), (6), and (7), we can immediately obtain the first and the second moments of the distance travelled to store and retrieve a pod within the locations in an aisle, $E[D_R]$ and $E[D_R^2]$, using the following expressions.

$$E[D_R] = \frac{(4N + 1)l}{3} \quad (8)$$

$$E[D_R^2] = \frac{(6N^2 + 2N - 1)l^2}{3} \quad (9)$$

The expected service time by a robot in the aisle can be obtained by taking the expectations of the terms in (2), which is given as:

$$E[t_{Aisle,R}] = E[t_{WL,R}] + t_{store} + t_{retrieval} + \frac{2d_{arr}^{loc}}{v_r} \quad (10)$$

Since $t_{WL,R} = \frac{D_R}{v_r}$, $E[t_{WL,R}]$ and $E[t_{WL,R}^2]$ are obtained from (11) and (12) as follows:

$$E[t_{WL,R}] = \frac{(4N + 1)l}{3v_r} \quad (11)$$

$$E[t_{WL,R}^2] = \frac{(6N^2 + 2N - 1)l^2}{3v_r^2} \quad (12)$$

Therefore using (10), we obtain the service rate associated with the robot service time for the random open location storage strategy,

$$\mu_{Aisle,R} = \frac{1}{E[t_{Aisle,R}]} \quad (13)$$

Using (10) and (11), we obtain the expected service time $E[t_{Aisle,R}]$ of a robot in a aisle.

The second moments of the robot service time in an aisle can be calculated as:

$$\begin{aligned} E[t_{Aisle,R}^2] &= E[t_{WL,R}^2] + \left(t_{store} + t_{retrieval} + \frac{2d_{arr}^{loc}}{v_r} \right)^2 \\ &\quad + 2 \left(t_{store} + t_{retrieval} + \frac{2d_{arr}^{loc}}{v_r} \right) E[t_{WL,R}] \end{aligned} \quad (14)$$

The variance and squared coefficient of variation (SCV) of the robot service time within an aisle is given by

$$Var(t_{Aisle,R}) = E[t_{Aisle,R}^2] - E[t_{Aisle,R}]^2 \quad (15)$$

$$cv_{Aisle}^2 = \frac{Var(t_{Aisle,R})}{E[t_{Aisle,R}]^2} \quad (16)$$

4.4. Average service time for cross-aisles

The cross-aisles are modeled as infinite server stations and hence the expected distance travelled by a robot on the cross-aisle on any of the eight cross-aisle segments (four in the order pick area and four in the replenishment area), $E[D_{CA}]$ can be obtained by the following expression:

$$E[D_{CA}] = \frac{A}{2} \left(w + \frac{d}{2} \right) + d_{arr}^{dep} \quad (17)$$

Using (17), the average travel time in a cross-aisle segment, $E[T_{CA}]$ can be obtained from

$$E[T_{CA}] = \frac{1}{v_r} E[D_{CA}] = \frac{1}{v_r} \left[\frac{A}{2} \left(w + \frac{d}{2} \right) + d_{arr}^{dep} \right] \quad (18)$$

where A is the total number of aisles.

The cross-aisle service rate is then expressed as:

$$\mu_{CA} = \frac{1}{E[T_{CA}]} \quad (19)$$

4.5. Service time at order pick and replenishment station

The service time of the picker at the order pick station is given by t_{pick} and of the worker at the replenishment station is given by $t_{replenish}$, which are both deterministic times and therefore $E[t_{pick}] = t_{pick}$ and $E[t_{replenish}] = t_{replenish}$.

$$\mu_{OP} = \frac{1}{t_{pick}} \quad \text{and} \quad \mu_{Rep} = \frac{1}{t_{replenish}}. \quad (20)$$

We now extend the double-command cycle to multiple aisles, where the empty robots can move underneath the pods and also travel between two storage zones (if required).

4.6. Underneath travel service time estimation

Let (i, j) denotes the location for storing a pod and (n, m) denotes the location for retrieving a pod, where i and n represent the aisle index, j and m represent the location index in a warehouse. Let $D(i, j; n, m)$ denotes the distance travelled from storage location (i, j) to retrieval location (n, m) within a storage zone. The distance $D(i, j; n, m)$ can be obtained from the following equation:

$$D(i, j; n, m) = |i - n| \times (2w + d) + |j - m| \times l \quad (21)$$

Average distance AD_{ij}^{nm} travelled by a robot underneath the rack location can be obtained by:

$$AD_{ij}^{nm} = \frac{1}{A^2 N^2} \sum_{s=k_3}^{k_4} \sum_{r=k_1}^{k_2} D(i, j; r, s) \\ k_3 = \min(m, j) \quad k_1 = \min(i, n) \\ k_4 = \max(m, j) \quad k_2 = \max(i, n) \quad (22)$$

where A denotes total number of aisles in a storage zone and N denotes total number of locations on each side of aisle.

The expected time for a robot for underneath travel within a zone, $E[t_{uz}]$, is given by:

$$E[t_{uz}] = \frac{AD_{ij}^{nm}}{v_r} \quad (23)$$

where v_r denotes the velocity of a robot. We now discuss the estimation of robot movement between two adjacent storage zones. Consider two storage zones Z_1 and Z_2 . Let i and n denotes aisle index of Z_1 and Z_2 , respectively and j and m denotes location index of zone Z_1 and Z_2 , respectively. When a robot switches the zone i.e. travelling from Z_1 to Z_2 or vice versa, the distance from a zone Z_p of aisle i and location j to a zone Z_q of aisle n and location m ($\forall p, q = 1, 2, \& p \neq q$), denoted by $D(i, j; n, m)$ is defined as:

$$D(i, j; n, m) = ((A + |i_1 - i_2|) \times (2w + d) + |j_1 - j_2| \times l)|z_p - z_q|, \\ \forall p, q = 1, 2, \& p \neq q \quad (24)$$

Average distance, $AD_{(Z_p, ij)}^{(Z_q, nm)}$, travelled by a robot from a zone Z_1 to zone Z_2 or vice versa can be obtained by:

$$AD_{(Z_p, ij)}^{(Z_q, nm)} = \frac{1}{A^2 N^2} \sum_{s=k_3}^{k_4} \sum_{r=k_1}^{k_2} D(i, j; r, s), \\ k_3 = \min(m, j) \quad k_1 = \min(i, n) \\ k_4 = \max(m, j) \quad k_2 = \max(i, n) \\ \forall p, q = 1, 2, p \neq q \quad (25)$$

The expected time $E[t_{unz}]$ for a robot to travel from one storage zone to another is given by:

$$E[t_{bz}] = \frac{AD_{(Z_p, ij)}^{(Z_q, nm)}}{v_r}, \forall p, q = 1, 2, p \neq q \quad (26)$$

where v_r denotes velocity of a robot. We now describe the queueing network models of the MFS.

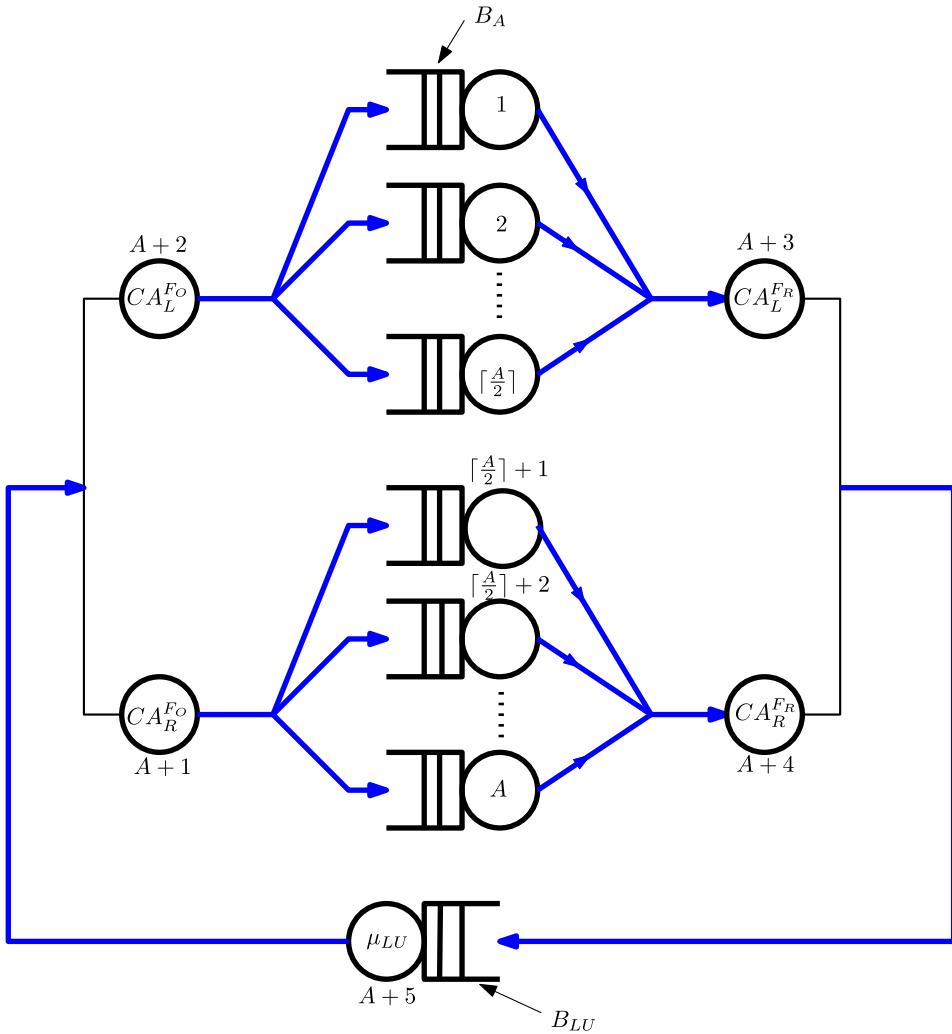


Fig. 6. Single-class queueing network model for order picking using a robotic MFS.

5. Queueing network models

In this section, we develop three closed queueing network models to answer the research questions. First, we develop a single-class queueing network model with one order picking station. Since our aim is to verify if dedicated robots for order picking and replenishment perform better than pooled robots in terms of throughput capacity, we propose two queueing network models with multi-class robots.

5.1. Single zone queueing network model with order-picking

In this subsection, we develop a closed queueing network model in which a single-class of V robots are considered for the order picking activity. The model is based on layout 1 described in Section 3.1. The queueing network shown in Fig. 6 has $A + 5$ nodes, where A denotes the number of aisles. Node $A + 5$ represents the time spent by a robot at the order picking station, nodes $A + 1$ and $A + 2$ represent the time spent by the robot in the left and right side of the departure path in the cross-aisle, nodes $A + 3$ and $A + 4$ represent the time spent by the robot on the left and right side of the arrival path in the cross-aisle.

A robot begins the order pick activity by fetching a pod from the storage location and proceeds towards the arrival path of a cross-aisle and chooses CA_L^{FA} or CA_R^{FA} . Then it proceeds towards the order picking station and waits in the buffer area for service at the order picking station (node $A + 5$). Then the robot chooses either a left or right node (i.e., CA_L^{FD} or CA_R^{FD}) with equal probability. Thereafter, a robot chooses any of the $\frac{A}{2}$ aisles with probability $\frac{1}{A/2}$ and stores the pod based on a storage policy. Using Approximate Mean Value Analysis (AMVA), we obtain the expected cycle time for order picking $E[CT_{op}]$, the throughput of order picking X_{op} , and expected queue lengths at various nodes.

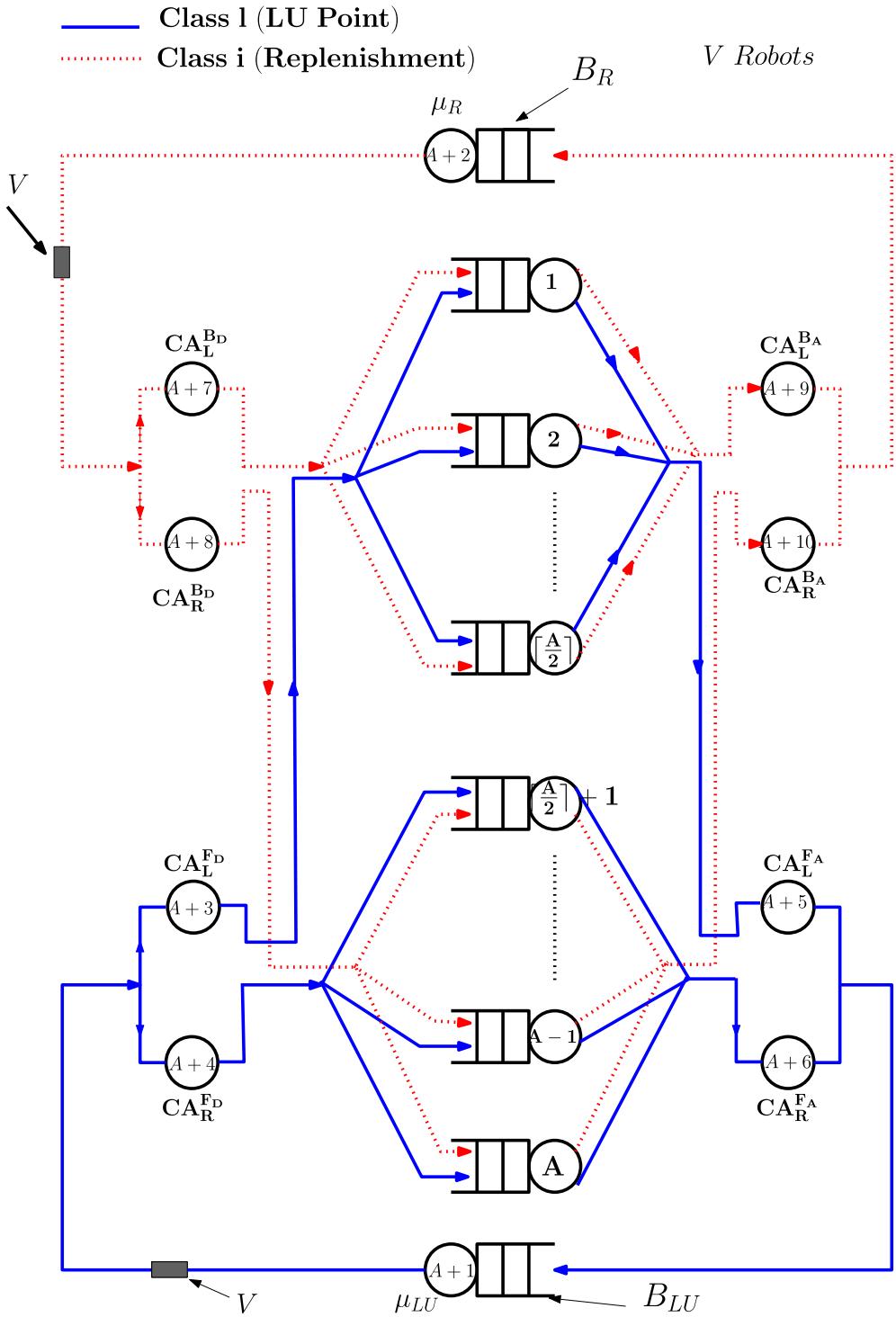


Fig. 7. Queueing network model for the MFS with dedicated robot classes.

5.2. Single zone queueing network model with order-picking and replenishment

In this subsection, we develop a closed queueing network model in which two classes of robots, o and r , are considered for the order picking and replenishment station respectively. We denote the number of robots for each class as V_c , where $c \in \{o, r\}$. The network corresponding to Fig. 7 consists of $A + 10$ service nodes, which denote the travel time in aisles, cross-aisles, order pick station, and replenishment station.

5.2.1. Queueing network model with dedicated robots

We first deploy two robot classes for the order pick and replenishment service, denoted by o and r respectively. The robots of class o which are deployed for the order pick service, begin their service by fetching a pod from an aisle storage location and travel back to access the cross-aisle nodes, either $A + 5$ (i.e., CA_L^{FA}) or $A + 6$ (i.e., CA_R^{FA}) (i.e., arrival-path of a cross aisle in forward direction), and proceed toward the node $A + 1$ order pick station (see Fig. 7). The robot waits in the order pick queue for its service (in a buffer area B_{OP}) and then accesses one of the two cross-aisle nodes, node $A + 3$ (i.e., CA_L^{FD}) or node $A + 4$ (i.e., CA_R^{FD}) with equal probability, and then accesses any one of the $\frac{A}{2}$ aisle resources with probability $\frac{1}{A/2}$. The robot stores the pod in an open storage location in the aisle. Then the pod storage-retrieval cycle is repeated by the robot i.e., the robot moves towards a retrieval location, retrieves a pod from the same side of an aisle and proceeds to the order pick station. The picker takes t_{pick} time to fetch an item from the inventory pod. The movement of a class r , replenishment robot, is identical to the class o robot except that the class r robot is serviced at the replenishment station and it uses the cross-aisle resources in the replenishment area. Note that the aisle resources are shared by robots from both classes (see Fig. 7). Similarly, a robot of class r , which is deployed for replenishment, fetches an inventory pod from a storage location, accesses either $A + 9$ (i.e., CA_L^{BA}) or $A + 10$ (i.e., CA_R^{BA}) based on its current location, proceeds towards the node $A + 2$ (replenishment station), and waits in the queue for its service. The worker at replenishment station takes $t_{replenish}$ time to replenish a pod which causes the robots to wait in a buffer area B_R . After being served, the robot accesses one of the two cross-aisle nodes $A + 7$ (i.e., CA_L^{BD}) or $A + 8$ (i.e., CA_R^{BD}) (i.e., departure-path of cross-aisle in replenishment area) with probability 0.5, and then accesses any one of the $\frac{A}{2}$ aisles with probability $\frac{1}{A/2}$ to store the replenishment pod. This marks the end of one retrieval-storage replenishment cycle. After storing the pod, the robot moves towards a retrieval location, retrieves another replenishment pod, and the cycle repeats.

The movement of class r and class o robots are shown by dotted and solid lines respectively in Fig. 7. Class switching is not allowed which means that the class o robots are not allowed to access the cross-aisle and replenishment station node of the replenishment area and the class r robots are not allowed to access the cross-aisle and order pick nodes of the order pick area in the queueing network model. Both classes of robots access the same aisle nodes because aisles are shared resources and only one robot can enter in an aisle at a time (refer the aisle protocol). Therefore, there is a buffer area, B_b , in front of aisle nodes, $i \in \{1, \dots, A\}$. Since both classes o and r access the same aisles to complete their service, the waiting times of a class o robots at aisle nodes is also dependent on the queue length of class r robots in front of the aisle.

Using Approximate Mean Value Analysis (AMVA) for multi-class closed queueing networks, we obtain the expected cycle times $E[CT_{op}]$ and $E[CT_{rep}]$, the class-specific expected queue lengths at all nodes, and the throughputs X_{op} and X_{rep} for the order pick and replenishment robot class, respectively (see Buitenhok et al., 2000 for details of the Approximate MVA algorithm). The indices c , n_c , and k refer to the robot class, the number of robots in class c , and the node respectively. The terms $V'_{c,k}$, $\bar{s}_{c,k}$, and $cv_{s_{c,k}}$ denote the visiting ratio, the average service time, and the service time coefficient of variation of class c robot at node k . The terms $Q_{c,k}(\vec{n})$, $U_{c,k}(\vec{n})$, $X_{c,k}(\vec{n})$, and $R_{c,k}(\vec{n})$ correspond to the average queue length, utilization, throughput, and residence time of class c robot at node k with \vec{n} state of robots in the network.

5.2.2. Queueing network model with pooled robots

A multiple class queueing network model with a single chain is developed for the MFS with pooled robots. Fig. 8 illustrates a system with $A + 10$ nodes and 2 robot classes with class switching. The system operates in the following way. There are two robot classes, one for order picking and the other for replenishment service, denoted by class index o and r respectively. The robots of class o , which are deployed for the order picking, access one of the cross-aisle nodes, node $A + 3$ (i.e., CA_L^{FD}) or node $A + 4$ (i.e., CA_R^{FD}) with probability 0.5, and the robots of class r , which are deployed for the replenishment station, accesses one of the cross-aisle nodes, node $A + 7$ (i.e., CA_L^{BD}) or node $A + 8$ (i.e., CA_R^{BD}) with probability 0.5. Then a robot of any class l or r accesses any one of the $\frac{A}{2}$ aisles with probability $\frac{1}{A/2}$. Within an aisle, the robot first stores a pod and then moves toward a pod retrieval location. With a probability p_r , the retrieval request corresponds to the order picking activity at the order pick station and with a probability $(1 - p_r)$, the retrieval request corresponds to the replenishment activity. Hence, probabilistic robot class switching is allowed in this queueing network model. The decision on the robot class switching should ideally be made immediately after the pod is stored in the storage location. After this service type decision, the robot either fetches an inventory pod for order picking or a replenishment pod for item replenishment. Since we model the complete aisle operation using a single service time distribution, we illustrate the class change operation in the Figure after the aisle service operation. Four cases are considered in which the robots either remains in the same class or switches to another class (see Fig. 8).

1. If a class o robot receives a retrieval request for order picking (with probability p_r) then the class o robot remains in the same class o , retrieves another pod for order picking and then accesses the cross-aisle nodes either $A + 5$ (i.e., CA_L^{FA}) or $A + 6$ (i.e., CA_R^{FA}) (i.e., arrival-path of a cross aisle in order pick area) to reach the order pick station.
2. If a class o robot receives a retrieval request for replenishment activity (with probability $1 - p_r$) then after completing its service in aisle nodes, class o robot changes its class to class r robot, retrieves the replenishment pod, and accesses the cross-aisle nodes either $A + 9$ (i.e., CA_L^{BD}) or $A + 10$ (i.e., CA_R^{BD}) to reach the replenishment station.
3. If a class r robot receives a retrieval request for the replenishment activity (with probability $1 - p_r$) then the class r robot after completing a service in aisle nodes remains in the same class r , retrieves another replenishment pod, and accesses the cross-aisle nodes either $A + 9$ (i.e., CA_L^{BD}) or $A + 10$ (i.e., CA_R^{BD}) (i.e., arrival-path of a cross aisle in replenishment area), to reach the replenishment station.

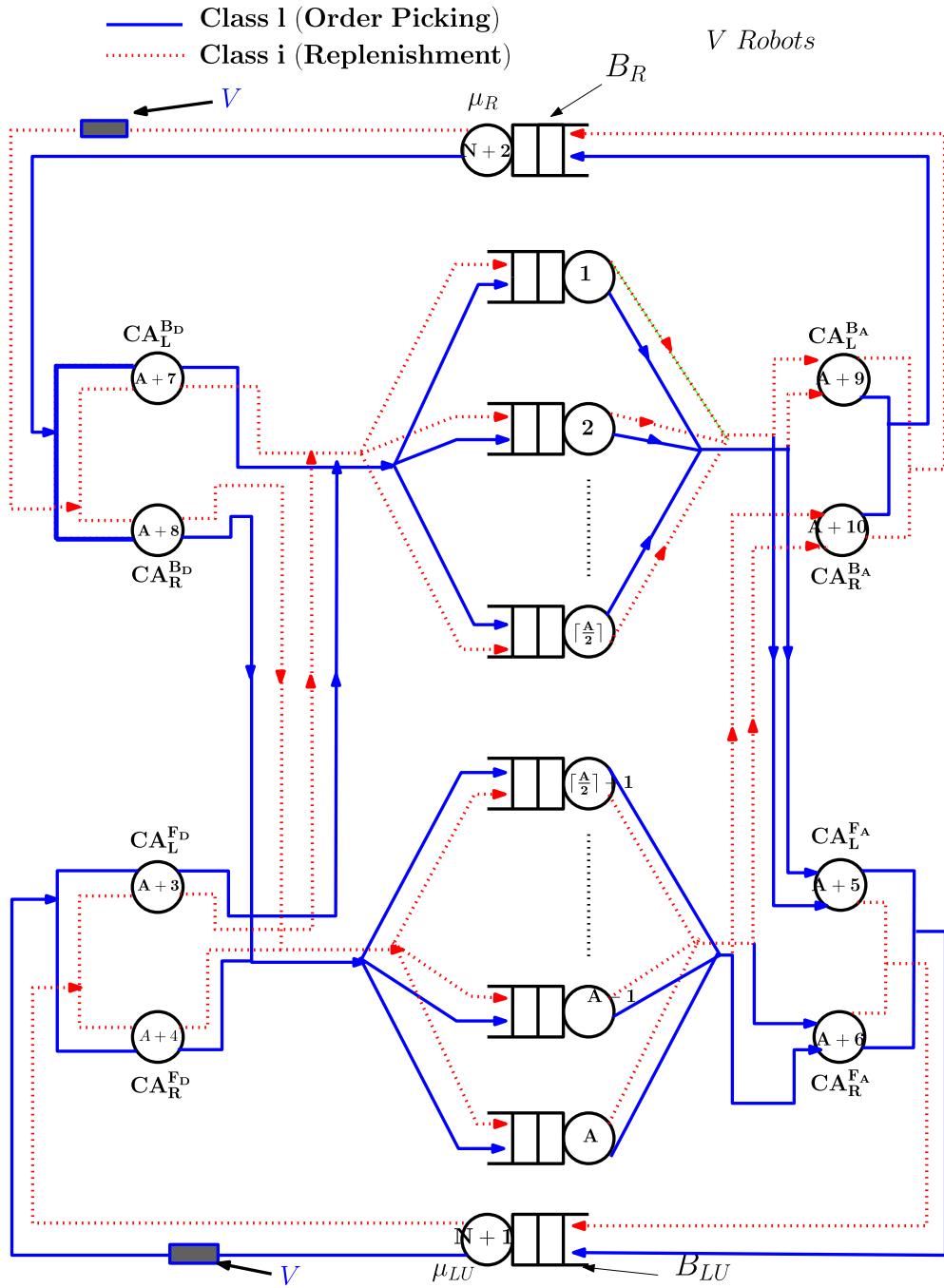


Fig. 8. Queueing network model for the MFS with pooled robots.

4. If a class r robot receives a retrieval request for order picking (with probability p_r) then the class r robot after completing its service in aisle nodes, changes its class to class o robot, retrieves an inventory pod for order picking and accesses the cross-aisle nodes either $A + 5$ (i.e., CA_L^{FA}) or $A + 6$ (i.e., CA_R^{FA}) to reach the order pick station.

In this way, a single chain is formed in which the number of robots is constant in a chain; however, within the chain, the robots change their class depending on the type of processing request. After being serviced at the cross-aisle nodes, a robot with or without class switch will proceed towards the node $A + 1$ (order pick station) or $A + 2$ (replenishment station) with probability 1.

Note that both classes of robots access the same aisle nodes based on the developed aisle protocol and it is assumed that only one robot can enter an aisle at a time.

Since the robots can switch their class in the aisles for the retrieval request, the cycle times are associated with the following

movements: a robot of any class starts its service by fetching a pod from a storage location, accesses the cross-aisles, service station (order pick or replenishment), accesses the cross-aisles to store the pod, and accesses the aisle to store the pod. Note that this cycle is equivalent if we consider the movement of the robot from either the order pick or the replenishment station in an aisle, travel for pod storage and another pod retrieval, and then return to either the order pick or the replenishment station to complete its service. Note that though the movement cycles are similar, the throughput time distributions in a cycle are different due to difference in the amounts of queueing at the nodes.

Using Approximate Mean Value Analysis (AMVA) for the multi-class closed queueing network with single chain, we obtain the expected cycle times $E[CT_{op}]$ and $E[CT_{rep}]$, the class-specific expected queue lengths at all nodes, and the throughputs X_{op} and X_{rep} for the order picking and replenishment robot class, respectively (Bolch et al., 2006). We also obtain the expected queue lengths (Q_{OP} , Q_{Rep} , Q_{Aisle}) and utilizations (U_{OP} , U_{Rep} , U_{Aisle}) for the order pick, replenishment, and aisle resources respectively.

5.3. Queueing network model with multiple storage zones and alternate robot assignment strategies

In this section, we focus on performance of the MFS with multiple storage zones, random pod storage strategy, and three alternate pod retrieval routing policies – dedicated, random, and shortest queue zone policy. We assume that multiple storage zones have identical SKU pods; which is an important aspect of an e-commerce order fulfillment center. Note that multiple orders may request for the same item at the same time and duplicating pods with the same SKU gives MFS the flexibility to fulfill orders simultaneously without delays. In this section, each storage zone is modeled as discussed in Section 5.1; albeit the robots can perform pod retrieval (double command cycle) from an aisle that is different from the pod storage aisle. Also, the robots can move between two zones depending on the assignment strategy. Models are developed for a system having three different robot assignment policies.

A single class queueing network model (a variant of the model illustrated in Fig. 5) is developed for the robotic MFS. In Section 4.6, we derived expressions for underneath travel of robot under the rack locations and travel time between two zones. We use the travel time expressions for analysis of system throughput using Markov chain models.

5.3.1. Queueing network model

For the purpose of illustration, we consider an MFS with two symmetric storage zones and an even number of aisles in each zone. Fig. 9 depicts closed queueing network of a single zone. We develop a closed queueing network model in which a single-class of V

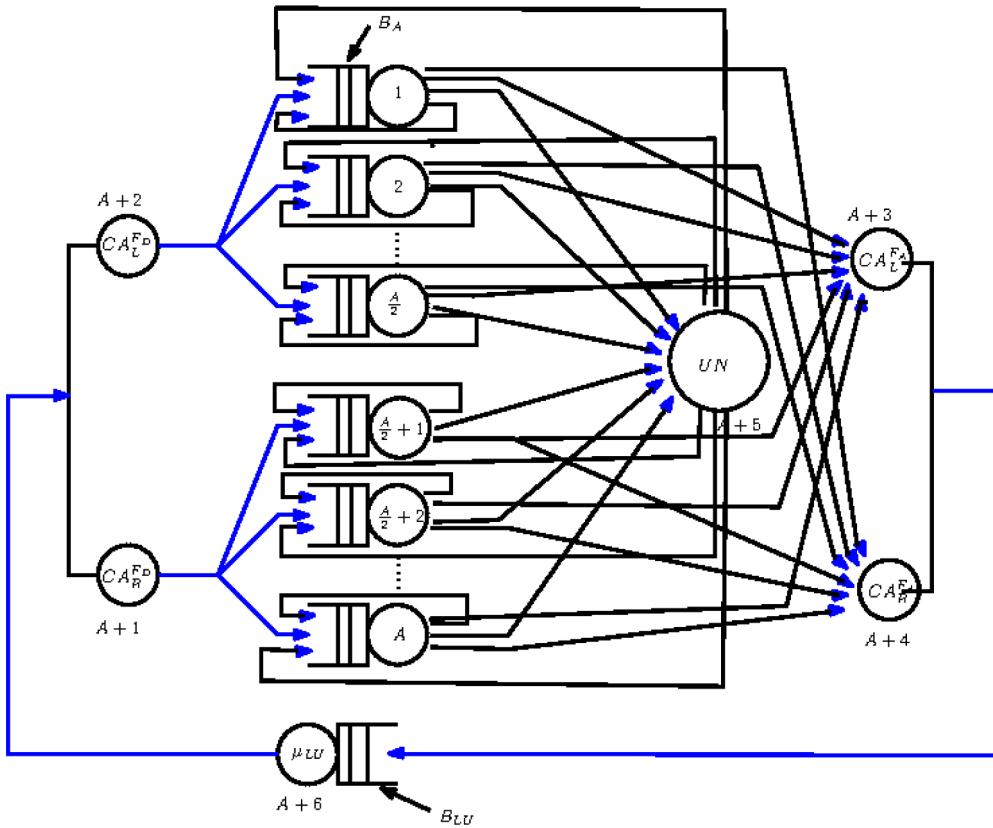


Fig. 9. Single-class queueing network model for storing and order picking with underneath travel, UN represents the Infinite-Server station corresponding to the underneath travel delay with expected time, $E[t_{u_x}]$.

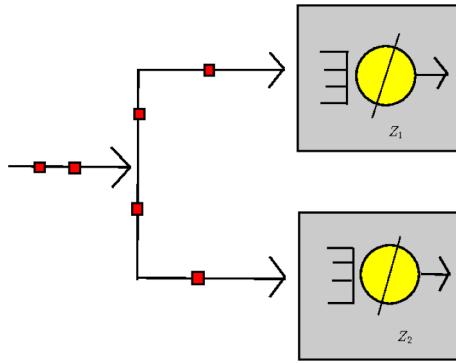


Fig. 10. Routing of robots between the zones based on dedicated policy.

robots are considered for order picking activity. A robot accesses either Z_1 or Z_2 with equal probability. The queueing network shown in the Fig. 9 has $A + 6$ nodes, where A denotes number of aisles. Node $A + 6$ represents the time spent by a robot at order pick station, nodes $A + 1$ and $A + 2$ represent the time spent by a robot in left and right side of departure path in the cross-aisles, nodes $A + 3$ and $A + 4$ represent the time spent by a robot on left and right side of arrival path in returning cross-aisles. Node $A + 5$ represents the time spent by robot travelling under the rack location for retrieving a pod. As before, a robot begins the order pick activity by fetching a pod from the storage location and proceeds towards the arrival path of a cross-aisle and chooses CA_L^{FA} or CA_R^{FA} . After storing a pod in any of an open location, a robot will retrieve a pod from one of a closed location. Here, robot has two possibilities for retrieving a pod, either it retrieves from the same aisle where it stores a pod with probability $\frac{1}{A} \times 0.5$ or it can move towards the other aisles by accessing the node $A + 5$ with probability $\frac{A-1}{A} \times 0.5$, and proceeds towards order pick station. Using Approximate Mean Value Analysis (AMVA), we obtain the throughput of the system. For random policy and shortest queue policy, the analytical results are obtained using a two-stage stochastic model. To analyze the two-zone system performance with state-dependent distribution of robots, we first estimate the steady state distribution of robots in the two zones using a discrete-time Markov chain. In the second stage, we determine the expected system throughput using the state-dependent zone throughput and the steady state probabilities obtained in the first stage. We discuss them in the next section.

5.3.2. Routing policies

First, the closed queueing network model corresponding to each zone is reduced to a load-dependent queue using Norton's theorem (see Viswanadham and Narahari, 1992). Note that this network reduction to a load-dependent queue is an approximate analysis because the underlying closed network is not product-form. The resultant load-dependent rates also need not be exponentially distributed. However, for throughput analysis, we assume the rates to be also exponential. Figs. 10 and 11 depict a warehouse having two zones namely, Z_1 and Z_2 . We consider three zoning policies, the dedicated, random and shortest queue zone policy. Note that a shortest queue zone policy may be preferred to address the temporal local congestion within the zone. In Fig. 10, a robot travels within a single zone (shown with a load-dependent queue) to store and retrieve a pod, which is considered as a dedicated storage zone policy, while in random and shortest queue zone policy robots can access either Z_1 or Z_2 (each represented with a load-dependent queue). Fig. 11 shows the random policy where storage and retrieval operations can be carried from either Z_1 or Z_2 . Similarly, in shortest queue zone policy, a robot can access any of the zone for storing a pod but for retrieving a pod, it moves

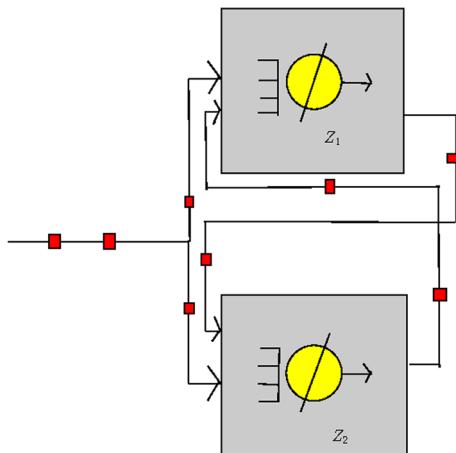


Fig. 11. Routing of robots between the zones based on random policy and shortest queue policies.

towards the least congested zone (measured by the number of robots present in the zone).

5.3.3. Throughput expressions for random zone policy and shortest queue zone policy

Consider the system with two zones and V robots operating in the random zone policy. The robots switch from one zone to another zone with expected time $E[t_{bz}]$, the analytical expression for calculating throughput $X_{Z_1}^A$ of Z_1 and $X_{Z_2}^A$ of Z_2 can be expressed as follows. The closed-queueing networks corresponding to Figs. 10 and 11 with V robots can be analyzed using continuous time Markov chains. The probability estimate, $\pi(s_i^r)$, denotes the steady state probability corresponding to state i in the state-dependent random zone assignment policy. The estimate of the state-dependent robot distribution is explained in Appendix A. Note that the error in the throughput estimation is introduced by the term, travel time between zones.

$$X_{Z_1}^A \approx \sum_{i=1}^V \left(\pi(s_{i+1}^r) \frac{1}{\left(\frac{1}{\mu_i} + E[t_{bz}] \right)} \right) \quad (27)$$

$$X_{Z_2}^A \approx \sum_{i=1}^V \left(\pi(s_{(V+1)-i}^r) \frac{1}{\left(\frac{1}{\mu_i} + E[t_{bz}] \right)} \right) \quad (28)$$

The analytical expression for calculating throughput of zone, $X_{Z_j}^A$, in shortest queue zone policy is expressed as follows. $\pi(s_i^{sq})$ denotes the steady state probability corresponding to state i in the state-dependent shortest queue zone assignment policy.

$$X_{Z_j}^A \approx \sum_{i=1}^V \left(\pi(s_{i+1}^{sq}) \frac{1}{\left(\frac{1}{\mu_i} + E[t_{bz}] \right)} \right), \forall j = 1, 2 \quad (29)$$

where μ_i denotes throughput of single zone with i number of robots, $\pi(s_i)$ denotes steady state probabilities, $E[t_{bz}]$ denotes service time for underneath travel of robot from one zone to another zone. Note that all throughput expressions correspond to the maximum system throughput assuming the robots are always busy processing transactions.

6. Numerical experiments and design insights

To perform the numerical experiments, we use the system dimension data from the trade websites as well as from discussions with practitioners (see Table 2). We employ discrete-event simulations to validate the proposed analytical models for order picking and replenishment processes.

Simulation experiments are conducted with ARENA software with 8000 time units (that corresponds to about 1 million order picks) and 15 replications were used to obtain the results. The confidence intervals for all performance measures are about $\pm 2\%$ of the average measure. Robots decide to access either the left or the right cross-aisles with probability $\frac{1}{2}$. Then a robot uses any one of the $\frac{A}{2}$ aisles with probability $\frac{1}{A/2}$. To access an aisle a robot has to wait for the preceding robot to complete its service. As soon as a preceding robot exits the aisle, the next robot enters an aisle to complete its service.

The absolute error in percentage deviations, δ_{P_k} of performance measure P at node k , with respect to the simulation and analytical methods are calculated by the following expression:

$$\delta_{P_k} = \left| \frac{A_{P_k} - S_{P_k}}{S_{P_k}} \right| \times 100 \quad (30)$$

where A_{P_k} is the value of the performance measure, P at node k obtained from the analytical model and S_{P_k} is the value of the performance measure, P at node k obtained from the simulation model. A summary of input parameters for the experiments used to validate the model is presented in Table 1 and Table 2. Note that the $\frac{D}{W}$ ratio is set at two levels: 1 and 2; the number of aisles (A) is set at two levels: 8 and 10, the number of storage locations (N_S) is set at two levels: 200 and 400; and the number of drive units (robots, V) is varied at three levels: 3, 4, and 5 in the single-class model (with only order picking) and 6, 8, 10, and 14 in the multi-class model (with both order picking and replenishment). The number of robots in real systems can be quite large (> 500). However, in practice, robots are typically assigned to one or two zones only, to avoid the large distance travels in practice (it is allocated to minimize the idle run of the robots). Hence, the number of robots in two zones is limited (8–10). The assignment of robots to specific zones reduces the algorithmic complexity of our model and our model can be run rapidly. In all cases the simulation model took about 15 min to run on a high configuration PC. In contrast, the analytical model took less than 15–20 s to run for the same configuration. Hence, the analytical models can be very useful for quickly exploring various alternatives in the large design space of a warehouse.

The maximum and the average percentage errors for all performance measures including queue lengths, resource utilization, throughput, throughput times is about 10% and < 5% respectively. The errors are higher in some occasions because the absolute value of the performance measure is itself very small.

In the next section, we present design insights based on the analysis of the three queueing network models.

Table 2
System dimensions and operation parameters.

Symbol	Description	Values
d_{arr}^{dep}	Distance between arrival & departure path	1.2 m
d_{arr}^{loc}	Distance between arrival path & starting point of rack locations	1 m
$\frac{d}{2}$	Distance among each side of rack locations to aisle	1 m
l	Gross length of pod location	0.99 m
w	Gross width of pod location	1 m
v_r	Speed of a robot	3 m/s
t_{store}	Time needed to store a pod	5 s
$t_{retrieve}$	Time needed to retrieve a pod	5 s
t_{pick}	Time needed for order picking at OP station	15 s
$t_{replenish}$	Time needed for replenishing a pod	90 s
p_r	Probability of an order pick transaction	0.8

6.1. Insights for the order picking queueing network model

To study the effect of the number of storage locations, we fix the $\frac{D}{W}$ ratio of the storage area, storage strategy and number of robots V . It is expected that the travel times within aisles (and hence expected throughput times) increase with an increase in the number of storage locations (Table 3 shows that the cycle times $E[CT_{op}]$ increase when the number of locations increase from 200 to 400), which results in the reduction in system throughput X . We analyze the $\frac{D}{W}$ ratio which affects the performance of the robotic MFS. With increase in the $\frac{D}{W}$ ratio, the number of aisles increases. Therefore, the queue lengths in front of aisles decrease and the waiting times reduce. However, the effect of overall delay in aisles and cross-aisle is not clear because by increasing the number of aisles, the service times on the cross-aisle also increase. In Table 3, we note that when the ratio of $\frac{D}{W}$ increases from 1 to 2, the robots take 1–2% less cycle times and warehouse throughput per hour increase. With $N_s = 200$ locations and $V = 3$ robots, the throughput of a system X increases from 210 to 217 picks/hr when the $\frac{D}{W}$ increases from 1 to 2. It is expected that the expected queue lengths Q_{Aisle} at the aisles and the order pick station Q_{OP} increases with the increase in the number of robots. This observation can be confirmed by the results given in Table 3.

6.2. Insights for the multi-class queueing network model

6.2.1. Dedicated class of robots for order picking and replenishment

Our observations on better system design with the dedicated robot classes are similar to the single-class case. For instance, from Table 4, we see that by increasing the locations in a warehouse, the waiting times at the aisles increase. This results in throughput reduction for both order picking and replenishment classes. Likewise, from Table 4 with 200 locations, we observe that when the ratio of $\frac{D}{W}$ increases from 1 to 2, the robots take about 1–3% less throughput times and the system throughput increases from 1% to 3%. Similarly, with 400 locations, by increasing the $\frac{D}{W}$ ratio from 1 to 2, the robots take 2–3% less cycle times and results in 1–3% more throughput capacity. We observe that an increase in $\frac{D}{W}$ ratio has a marginal impact on the system performance measures.

6.2.2. Pooled class of robots for order picking and replenishment

In this case, a robot can process both order picking and replenishment activity. From Table 5, it can be seen that the insights that hold for a single-class or a multi-class with dedicated robots also hold true for the pooled robot system.

This work is motivated to the answer the question if pooled robots in usage perform better than dedicated robots for order picking and replenishment activity.

From the numerical results, we see that dedicated robots take less throughput times than the pooled robots for the order picking process whereas the dedicated robots that more throughput times than the pooled robots for the replenishment process. In the pooled classes, we use a 80% chance that the robots can switch to the order picking process and 20% chance that the robots switches to the replenishment process. Note that a pooled resource allocation may result in more variability in the queue lengths at the order pick and replenishment stations. Further, the replenishment station has a longer service time requirements than the order pick service time. Therefore, pooled robot assignment to the replenishment process results in two to three times more average waiting time than the dedicated robot assignment (refer Tables 4 and 5). However, the pooled robot assignment results in 30–60% reduction in order pick throughput times in comparison to the dedicated robot assignment.

In Fig. 12, expected replenishment and order pick throughput times are compared for a system with six robots but different robot allocations. For instance, if the number of robots dedicated to replenishment and order picking are 2 and 4 respectively, then we compare the average throughput time measures with a pooled system where the robot switches to replenishment and order picking with probability (2/6) and (4/6) respectively after completing every process cycle. We see that for all set of robot allocations, the average queue length at the order pick station is lower for the pooled case than the dedicated robot case. However, the average queue length at the replenishment station is higher for the pooled case than the dedicated case. Therefore the overall waiting time at the

Table 3
Performance measures of MFS using a single robot class.

Policy	N_s	Input						Output: Performance measure														
		$\frac{D}{W}$	V	A_{QOP}	S_{QOP}	δ_{QOP}	A_{Q_Aisle}	S_{Q_Aisle}	δ_{Q_Aisle}	A_{UOP}	S_{UOP}	δ_{UOP}	A_{U_Aisle}	S_{U_Aisle}	δ_{U_Aisle}	$A_E[CTop]$	$S_E[CTop]$ (seconds)	$\delta_E[CTop]$	A_x (per hour)	S_x (per hour)	δ_x	
137	Random	200	1	3	1.70	1.65	2.8%	0.19	0.4%	85%	87%	1.9%	0.17	2.3%	0.17	52.70	51.43	2.5%	204.84	209.98	2.4%	
		4	2.53	2.53	0.1%	0.22	0.21	2.6%	0.21	0.94%	95%	1.2%	0.19	0.6%	0.19	63.92	62.22	2.7%	225.36	231.44	2.6%	
		5	3.45	3.43	0.6%	0.23	0.22	3.6%	0.22	0.98%	98%	0.3%	0.20	4.7%	0.19	76.78	75.33	1.9%	234.36	238.94	1.9%	
	2	3	1.81	1.80	0.6%	0.09	0.09	5.1%	0.10	88%	89%	0.6%	0.09	0.08	12.1%	50.89	49.81	2.2%	212.40	216.84	2.0%	
	4	2.70	2.71	0.5%	0.10	0.10	3.9%	0.10	96%	97%	1.1%	0.10	0.09	8.1%	62.56	61.98	0.9%	230.04	232.32	1.0%		
	5	3.65	3.67	0.5%	0.11	0.10	7.7%	0.11	100%	99%	1.3%	0.10	0.10	0.2%	75.95	74.98	1.3%	236.88	240.07	1.3%		
	Random	400	1	3	1.54	1.51	1.7%	0.16	0.15	6.3%	81%	83%	1.9%	0.15	0.14	6.1%	55.26	53.80	2.7%	195.48	200.76	2.6%
		4	2.32	2.33	0.4%	0.18	0.18	1.9%	0.18	92%	93%	1.6%	0.17	0.16	4.3%	65.57	65.55	0.0%	219.60	219.70	0.0%	
		5	3.21	3.15	1.8%	0.20	0.19	3.3%	0.19	97%	97%	0.5%	0.18	0.17	3.6%	77.70	77.48	0.3%	231.48	232.30	0.4%	
	2	3	1.61	1.59	1.4%	0.12	0.11	5.3%	0.12	84%	85%	1.6%	0.11	0.10	9.7%	53.80	52.97	1.6%	200.88	203.89	1.5%	
	4	2.43	2.44	0.3%	0.13	0.13	0.8%	0.13	93%	95%	2.0%	0.12	0.12	1.8%	64.43	63.44	1.6%	223.56	227.00	1.5%		
	5	3.35	3.35	0.1%	0.14	0.13	6.5%	0.14	97%	99%	1.5%	0.13	0.12	6.5%	76.95	75.18	2.4%	234.00	239.43	2.3%		

Table 4
Performance measure of MFS with dedicated robots.

Input			Output: Performance measure (dedicated class)													
$\frac{D}{W}$	N_s	V	A_{QOP}	S_{QOP}	δ_{QOP}	$A_{QR_{rep}}$	$S_{QR_{rep}}$	$\delta_{QR_{rep}}$	A_{Q_Aisle}	S_{Q_Aisle}	δ_{Q_Aisle}	A_{UOP}	S_{UOP}	δ_{UOP}	$A_{UR_{rep}}$	$S_{UR_{rep}}$
1	200	6	3.41	3.40	0.2%	0.78	0.77	1.7%	0.25	0.9%	96%	0.5%	78%	77%	78%	77%
	8	5.34	5.34	0.0%	0.78	0.77	1.6%	0.27	0.26	2.3%	100%	0.5%	78%	78%	78%	78%
	10	6.31	6.33	0.3%	1.72	1.72	0.0%	0.28	0.27	3.1%	100%	0.0%	99%	99%	99%	99%
	14	9.29	9.31	0.2%	2.71	2.71	0.1%	0.28	0.28	1.4%	100%	0.0%	100%	100%	100%	100%
2	200	6	3.56	3.59	0.8%	0.80	0.80	0.3%	0.12	0.12	0.5%	97%	99%	1.5%	80%	80%
	8	5.52	5.49	0.5%	0.80	0.80	0.2%	0.12	0.12	3.5%	100%	0.0%	80%	80%	80%	80%
	10	6.51	6.53	0.4%	1.75	1.76	0.5%	0.12	0.12	3.3%	100%	0.0%	100%	100%	99%	99%
	14	9.50	9.52	0.3%	2.74	2.75	0.3%	0.13	0.13	0.4%	100%	0.0%	100%	100%	100%	100%
1	400	6	3.16	3.10	1.9%	0.75	0.75	0.5%	0.22	0.20	9.0%	95%	97%	2.4%	75%	74%
	8	5.06	5.03	0.6%	0.75	0.76	0.9%	0.23	0.20	14.4%	99%	98%	0.7%	75%	75%	75%
	10	6.03	6.02	0.2%	1.68	1.68	0.2%	0.24	0.23	4.5%	100%	0.5%	98%	98%	96%	96%
	14	9.00	8.99	0.1%	2.66	2.67	0.4%	0.25	0.23	6.9%	100%	0.0%	100%	100%	100%	100%
2	400	6	3.28	3.30	0.7%	0.77	0.77	0.3%	0.15	0.14	9.8%	96%	98%	2.5%	77%	77%
	8	5.20	5.21	0.2%	0.77	0.77	0.3%	0.16	0.16	0.1%	99%	99%	0.2%	77%	77%	76%
	10	6.18	6.20	0.4%	1.70	1.70	0.0%	0.17	0.16	4.6%	100%	0.1%	98%	98%	96%	96%
	14	9.16	9.17	0.2%	2.69	2.60	3.3%	0.17	0.17	0.1%	100%	0.0%	100%	100%	100%	100%
Input			Output: Performance measure (dedicated class)													
$\frac{D}{W}$	$\delta_{UR_{rep}}$	A_{U_Aisle}	S_{U_Aisle}	δ_{U_Aisle}	$A_E[COp]$	$S_E[COp]$	$\delta_E[COp]$	$A_E[CR_{rep}]$	$S_E[CR_{rep}]$	$\delta_E[CR_{rep}]$	A_{XOp}	S_{XOp}	δ_{XOp}	A_{XRep}	S_{XRep}	δ_{XRep}
1	1.7%	22%	22%	1.4%	77.8	76.8	1.2%	114.9	112.3	2.3%	231.5	234.3	1.2%	31.3	32.1	2.3%
	0.3%	23%	22%	4.2%	105.5	104.2	1.2%	115.1	112.0	0.7%	239.0	241.8	1.1%	31.3	31.5	0.6%
	0.1%	24%	23%	3.3%	119.8	121.4	1.3%	181.7	189.3	4.0%	240.5	237.2	1.4%	39.6	38.0	4.1%
	0.0%	24%	24%	0.0%	163.7	165.7	1.2%	262.6	265.4	1.1%	242.1	239.0	1.3%	41.0	40.7	0.9%
2	0.3%	11%	10%	10.0%	76.9	77.5	0.7%	112.9	111.8	0.9%	230.4	232.3	0.8%	32.0	32.2	0.5%
	0.2%	12%	11%	4.6%	104.9	105.4	0.5%	112.2	113.3	1.0%	241.2	239.0	0.9%	32.0	31.8	0.9%
	1.0%	12%	11%	8.1%	119.4	121.0	1.3%	179.8	189.9	5.3%	230.4	238.0	3.2%	37.4	37.9	1.2%
	0.0%	12%	11%	9.0%	163.4	165.2	1.1%	261.5	263.5	0.7%	242.3	239.7	1.1%	41.4	41.0	1.0%
1	1.9%	20%	19%	2.9%	79.2	77.2	2.6%	119.3	120.3	0.8%	227.2	233.1	2.5%	30.2	29.9	1.1%
	0.4%	20%	20%	1.5%	106.4	108.8	2.3%	119.5	120.5	0.8%	236.9	231.6	2.3%	30.1	29.9	0.8%
	1.8%	21%	20%	5.6%	120.6	121.6	0.9%	184.3	190.9	3.5%	239.0	236.8	0.9%	39.2	37.7	4.1%
	0.0%	21%	20%	7.2%	164.1	166.7	1.5%	264.2	278.9	5.3%	241.2	237.6	1.5%	41.0	38.7	6.0%
2	0.3%	14%	13%	9.2%	78.5	78.2	0.4%	117.2	119.2	1.7%	229.3	230.1	0.3%	30.6	30.2	1.3%
	1.0%	15%	14%	4.9%	105.9	106.2	0.3%	117.3	119.4	1.8%	238.0	237.2	0.3%	30.6	30.1	1.5%
	2.5%	15%	15%	1.6%	120.2	121.3	0.9%	182.9	191.2	4.3%	239.8	237.4	1.0%	39.2	37.7	4.2%
	0.0%	15%	14%	10.3%	163.9	161.7	1.4%	263.3	264.3	0.4%	241.6	245.0	1.4%	41.0	40.9	0.4%

Table 5
Performance measure of MFS with pooled robots.

Input										Output: Performance measure (Pooled Class)									
$\frac{D}{W}$	N_e	V	A_{QOP}	S_{QOP}	δ_{QOP}	A_{QRep}	S_{QRep}	δ_{QRep}	A_{QAisde}	S_{QAisde}	δ_{QAisde}	A_{UOP}	S_{UOP}	δ_{UOP}	A_{URep}	S_{URep}	δ_{URep}		
1	200	6	1.34	1.30	2.7%	3.31	3.40	2.7%	0.19	0.19	0.5%	67%	65%	2.8%	99%	99%	98%		
	8	1.58	1.50	5.3%	5.02	5.19	3.3%	0.20	0.19	3.9%	69%	66%	3.8%	99%	99%	98%			
	10	1.71	1.69	0.9%	6.89	6.90	0.2%	0.20	0.19	4.5%	69%	65%	5.7%	100%	100%	100%			
	14	1.77	1.69	5.0%	10.83	10.65	1.7%	0.20	0.19	3.7%	68%	66%	3.4%	100%	100%	100%			
2	200	6	1.36	1.31	3.8%	3.41	3.50	2.6%	0.09	0.09	0.0%	67%	64%	5.0%	94%	96%	98%		
	8	1.60	1.48	7.8%	5.14	5.25	2.0%	0.09	0.09	3.1%	69%	66%	4.0%	99%	99%	98%			
	10	1.71	1.69	1.4%	7.02	7.10	1.1%	0.09	0.09	3.3%	69%	66%	4.1%	100%	100%	100%			
	14	1.78	1.78	0.2%	10.97	10.88	0.8%	0.09	0.09	2.7%	68%	66%	3.4%	100%	100%	100%			
1	400	6	1.29	1.20	7.6%	3.12	3.15	0.9%	0.17	0.16	3.5%	66%	66%	0.2%	99%	99%	99%		
	8	1.55	1.51	2.7%	4.80	4.81	0.3%	0.17	0.17	1.5%	68%	63%	8.5%	99%	99%	99%			
	10	1.69	1.62	4.4%	6.64	6.66	0.3%	0.17	0.17	2.2%	69%	68%	1.0%	100%	100%	100%			
	14	1.77	1.73	2.5%	10.57	10.60	0.3%	0.17	0.17	1.6%	68%	68%	0.5%	100%	100%	100%			
2	400	6	1.31	1.30	0.9%	3.20	3.21	0.3%	0.12	0.11	6.1%	67%	66%	0.8%	100%	97%	99%		
	8	1.57	1.51	3.6%	4.90	4.91	0.3%	0.12	0.12	0.7%	68%	66%	3.7%	100%	99%	100%			
	10	1.70	1.59	6.9%	6.75	6.79	0.5%	0.12	0.12	1.2%	69%	66%	4.1%	100%	100%	100%			
	14	1.77	1.73	2.6%	10.69	10.70	0.1%	0.12	0.12	0.5%	68%	68%	0.4%	100%	100%	100%			
Input										Output: Performance measure (Pooled Class)									
$\frac{D}{W}$	δ_{URep}	A_{UAisde}	S_{UAisde}	δ_{UAisde}	$A_{E[CTop]}$	$S_{E[CTop]}$	$\delta_{E[CTop]}$	$A_{E[CTRep]}$	$S_{E[CTRep]}$	$\delta_{E[CTRep]}$	A_{XOp}	S_{XOp}	δ_{XOp}	A_{XRep}	S_{XRep}	δ_{XRep}			
1	1.0%	17%	17%	0.1%	54.4	54.7	0.7%	321.5	323.9	0.7%	331.2	329.0	0.7%	11.2	11.1	0.7%	7.5	2.8%	
	1.0%	17%	17%	2.6%	59.1	59.5	0.7%	464.1	476.8	2.7%	426.3	423.4	0.7%	7.8	7.5	2.1%	11.3	2.1%	
	0.0%	18%	17%	2.9%	61.8	60.9	1.6%	625.9	638.7	2.0%	465.8	473.1	1.5%	11.5	11.3	0.9%	11.2	0.9%	
	0.0%	17%	17%	2.2%	63.6	65.3	2.6%	976.0	966.8	1.0%	622.9	606.8	2.6%	11.1	11.2	0.1%	10.9	0.1%	
2	2.2%	8%	8%	0.0%	52.3	51.3	2.1%	326.2	349.9	6.8%	343.9	351.2	2.1%	11.0	10.3	7.2%	7.4	2.9%	
	9%	8%	8%	8.9%	56.9	54.3	4.9%	471.8	485.6	2.8%	442.8	464.5	4.7%	7.6	7.4	2.0%	11.3	2.0%	
	9%	8%	8%	9.0%	59.5	56.6	5.0%	635.2	682.3	6.9%	484.4	508.5	4.7%	11.3	10.6	0.1%	11.2	0.3%	
	9%	8%	8%	8.2%	61.1	61.8	1.1%	986.4	987.2	0.1%	648.6	641.3	1.1%	10.9	10.9	0.1%	11.2	0.3%	
1	0.2%	15%	15%	0.5%	58.1	60.2	3.5%	311.9	315.3	1.1%	309.8	298.9	3.6%	11.5	11.4	1.1%	11.4	2.0%	
	0.0%	16%	15%	3.9%	63.1	66.2	4.8%	449.9	456.0	4.5%	399.6	380.4	5.0%	8.0	7.6	4.8%	11.6	2.4%	
	0.0%	16%	15%	4.5%	66.0	71.2	7.3%	609.2	621.3	1.9%	436.3	404.3	7.9%	11.8	11.6	2.0%	11.5	1.2%	
	0.0%	16%	15%	3.8%	68.0	69.2	1.7%	957.7	960.3	0.3%	582.2	572.3	1.7%	11.3	11.2	0.3%	11.2	0.3%	
2	2.8%	11%	10%	9.0%	56.4	57.3	1.5%	315.5	321.7	1.9%	318.9	314.0	1.6%	11.4	11.2	2.0%	11.2	2.0%	
	1.0%	11%	11%	2.0%	61.3	59.1	3.6%	456.0	466.8	2.3%	411.3	426.2	3.5%	7.9	7.7	2.4%	11.7	1.2%	
	0.0%	11%	11%	2.4%	64.1	61.5	4.2%	616.7	624.0	1.2%	449.4	468.4	4.1%	11.7	11.5	1.2%	11.2	1.2%	
	0.0%	11%	11%	1.7%	66.0	65.2	1.1%	966.2	963.2	0.3%	600.3	607.0	1.1%	11.2	11.2	0.3%	11.2	0.3%	

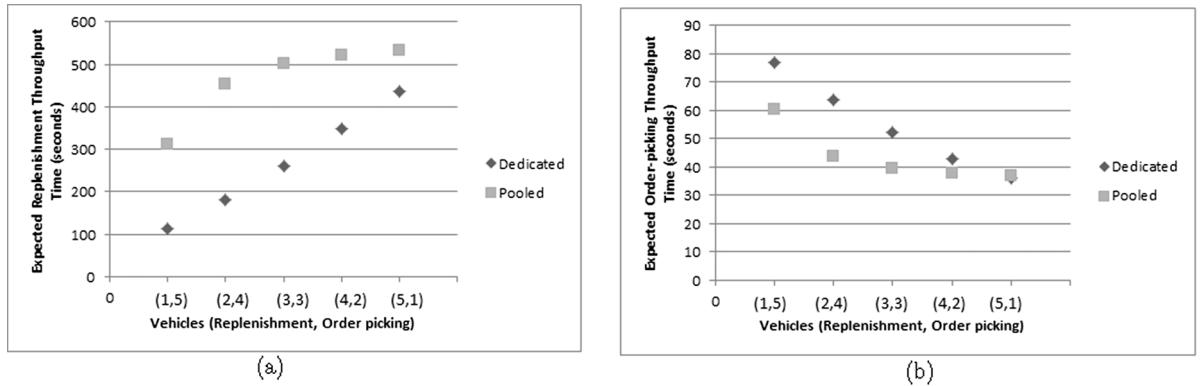


Fig. 12. Comparison of expected replenishment and order picking throughput times (a and b) with dedicated and pooled robots.

order pick station decreases in the pooled robot case, which decreases the overall throughput time for order picking (up to 60% less).

6.3. Insights for the single-class queueing network model with alternate routing policies for retrieval

Numerical experiments are carried out with 6, 8, and 10 robots in a system. We consider five locations on each side of the aisle, with three open locations. For different policies we obtained throughput and analyze the variation in routing policies on system throughput. Experiments are carried out to study the effect of aisles, we fix robots in a system and aisles are varied. The results provided in Table 6 shows that, as the number of aisles increases the throughput decreases (as expected due to increase in expected robot travel time) but the shortest queue zone policy provides slightly better throughput than the random zone policy. The results also indicate that the throughput of the shortest queue zone policy is almost same as the throughput obtained with dedicated zone policy. This results suggest that shortest queue zone policy can be adopted to balance the time-varying workload within a zone and achieve a throughput performance, which is close to that of a dedicated zone.

Table 6

Throughput of two zones with different routing policies.

Policy	Robots	Aisles	$X_{Z_1}^A$ (picks/hr)	$X_{Z_2}^A$ (picks/hr)
Dedicated	6	8	251.28	251.28
		10	247.32	247.32
		12	241.92	241.92
		8	315.36	315.36
	8	10	313.92	313.92
		12	309.24	309.24
		8	370.8	370.8
		10	372.96	372.96
	10	12	370.08	370.08
		8	234.51	234.51
		10	233.39	233.39
		12	230.14	230.14
Random	6	8	289.89	289.89
		10	291.71	291.71
		12	289.96	289.96
		8	336.58	336.58
	8	10	341.72	341.72
		12	341.87	341.87
		8	250.09	250.09
		10	246.39	246.39
	10	12	241.18	241.18
		8	313.49	313.49
		10	312.43	312.43
		12	308.03	308.03
Shortest	8	8	368.22	368.22
		10	370.86	370.86
		12	368.34	368.34

7. Conclusion and future scope

In this paper, we present analytical models to estimate the performance measures of mobile-shelves based order pick systems. Using these models, we analyze the system operations for both single and multiple storage zones with dedicated or pooled robots. Using multi-class closed queueing network models, we observe that by using pooled robots instead of dedicated robots, the expected throughput time for order picking reduces up to one-third of its initial value; however, the expected replenishment time estimate increases up to three times. Using a two-stage stochastic model, we analyze the performance of multiple zones. In particular, we analyze the effect of balancing congestion across the zones by allocating robots to retrieve pods from storage zones with less congestion. Here, we observe that shortest queue zone policy yields similar throughput performance as the dedicated zone policy, and both outperform the random zone assignment policy. Analysis of more than two storage zones by scaling the model can also be done in the future albeit with additional expansion of the state space in the parameter estimation stage of the stochastic model.

Acknowledgments

This research work is supported by the Research & Publication Office, IIM Ahmedabad. The authors are thankful to research scholar Ms. Rozaline Maharana for performing additional numerical experiments and testing the robustness of results. The authors are also thankful to Pete Wurman, Chief Technology Officer, at Kiva Systems, Inc. (now Amazon Robotics) for helpful discussions during the initial part of the study.

Appendix A. Estimation of steady-state probabilities for performance analysis of multiple storage zones

We determine the steady state probabilities for random zone and shortest queue zone policy using continuous-time Markov chain (CTMC) analysis. The transition rate matrix for the random zone is first explained below. Suppose there are total V number of robots and m number of zones in the distribution warehouse, then there will be total $M = \binom{V+m-1}{m-1}$ states i.e. $V+1$ states. In this work, we consider two zones and even number of aisles. Let us define the state i, s_i^r , of the CTMC with a two-tuple vector (c, d) where c and d denote the number of robots present in zone z_1 and z_2 , respectively. For random policy, suppose there are $V = 4$ robots and $m = 2$ zones in a warehouse, then $M = 5$ states will be formed, which are denoted by $s_1^r = (0, 4)$, $s_2^r = (1, 3)$, $s_3^r = (2, 2)$, $s_4^r = (3, 1)$, $s_5^r = (4, 0)$. The transition rate from $(0, 4)$ to $(1, 3)$ is $0.5\mu_4$. Likewise, the transition rate from $(1, 3)$ to $(2, 2)$ and $(0, 4)$ is $0.5\mu_3$ and $0.5\mu_1$, respectively. Likewise, for shortest queue zone policy, let $V = 4$, then the number of states will be $M = 5$. The states are $s_1^{sq} = (0, 4)$, $s_2^{sq} = (1, 3)$, $s_3^{sq} = (2, 2)$, $s_4^{sq} = (3, 1)$, $s_5^{sq} = (4, 0)$. The transition rate from $(0, 4)$ to $(1, 3)$ is μ_4 . Likewise, the transition rate from $(1, 3)$ to $(2, 2)$ is μ_3 . For both cases, we obtain the steady state probabilities by solving the system of linear equations $\Pi Q = 0$ with $\Pi e = 1$, where Q is the rate matrix.

Appendix B. Supplementary material

Supplementary data associated with this article can be found, in the online version, at <https://doi.org/10.1016/j.tre.2018.11.005>.

References

- Azadeh, K., De Koster, R., Roy, D., 2018. Robotized warehouse systems: developments and research opportunities warehousing in the e-commerce era: a survey. *Transport. Sci.* (in press) doi: <https://doi.org/10.2139/ssrn.2977779>.
- Bolch, G., Stefan, G., Meer, H.D., Trivedi, K.S., 2006. *Queueing Networks And Markov Chains: Modeling and Performance Evaluation with Computer Science Applications Vol. 2* John Wiley and Sons, Hoboken, New Jersey.
- Boysen, N., Briskorn, D., Emde, S., 2017. Parts-to-picker based order processing in a rack-moving mobile robots environment. *Eur. J. Oper. Res.* 262 (2), 550–562.
- Boysen, N., De Koster, R., Weidinger, F., 2018. Warehousing in the e-commerce era: a survey. *Eur. J. Oper. Res.* <https://doi.org/10.1016/j.ejor.2018.08.023>.
- Buitenhak, R., van Houtum, G-J., Zijm, H., 2000. AMVA-based solution procedures for open queueing networks with population constraints. *Ann. Oper. Res.* 93 (1/4), 15–40.
- Cai, X., Heragu, S.S., Liu, Y., 2013. Modeling and evaluating the avs/rs with tier-to-tier vehicles using semi-open queueing network. *IIE Trans.* <https://doi.org/10.1080/0740817X.2013.849832>.
- D'Andrea, R., 2011. Mobile-robot-enabled smart warehouses. In: Samad, T., Annaswamy, A.M. (Eds.), *The Impact of Control Technology: Overview, Success Stories, and Research Challenges*. IEEE Control Systems Society. <http://www.ieeeccs.org/general/impact-control-technology>.
- D'Andrea, R., 2012. Guest editorial: A revolution in the warehouse: A retrospective on kiva systems and the grand challenges ahead. *IEEE Trans. Autom. Sci. Eng.* 9 (4), 638–639.
- D'Andrea, R., Wurman, P., 2008. Future challenges of coordinating hundreds of autonomous vehicles in distribution facilities. In: *IEEE International Conference on Technologies for Practical Robot Applications*. TePRA 2008, pp. 80–83.
- Daniels, R.L., Rummel, J.L., Schantz, R., 1998. A model for warehouse order picking. *Eur. J. Oper. Res.* 105 (1), 1–17.
- De Koster, R., Le-Duc, T., Roodbergen, K.J., 2007. Design and control of warehouse order picking: a literature review. *Eur. J. Oper. Res.* 182 (2), 481–501.
- Ekren, B.Y., Heragu, S.S., Krishnamurthy, A., Malmborg, C.J., 2013. An approximate solution for semi-open queueing network model of an autonomous vehicle storage and retrieval system. *IEEE Trans. Autom. Sci. Eng.* 10 (1), 205–215.
- Enright, J.J., Wurman, P.R., 2011. Optimization and coordinated autonomy in mobile fulfillment systems. In: *Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence*. AAAI Publications, pp. 33–38.
- Fukunari, M., Malmborg, C.J., 2008. An efficient cycle time model for autonomous vehicle storage and retrieval systems. *Int. J. Prod. Res.* 46 (12), 3167–3184.
- Fukunari, M., Malmborg, C.J., 2009. A network queuing approach for evaluation of performance measures in autonomous vehicle storage and retrieval systems. *Eur. J. Oper. Res.* 193, 152–167.
- Kuo, P.H., Krishnamurthy, A., 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. *Appl. Math. Model.* 31, 2332–2346.

- Lamballais, T., Roy, D., De Koster, M.B.M., 2017a. Inventory allocation in robotic mobile fulfillment systems. <https://doi.org/10.2139/ssrn.2900940>.
- Lamballais, T., Roy, D., De Koster, M.B.M., 2017. Estimating performance in a robotic mobile fulfillment system. *Eur. J. Oper. Res.* 256 (3), 976–990.
- Le-Anh, T., De Koster, M.B.M., 2006. A review of design and control of automated guided vehicle systems. *Eur. J. Oper. Res.* 171 (1), 1–23.
- Malmborg, C.J., 2003. Interleaving dynamics in autonomous vehicle storage and retrieval systems. *Int. J. Prod. Res.* 41 (5), 1057–1069.
- Malmborg, C.J., 2003. Design optimization models for storage and retrieval systems using rail guided vehicles. *Appl. Math. Model.* 27 (12), 929–941.
- Mountz, M., 2012. Kiva the disrupter. *Harvard Bus. Rev.* 90, 74–80.
- Nigam, S., Roy, D., De Koster, R., Adan, I., 2014. Analysis of class-based storage strategies for the mobile shelf-based order pick system, chapter *Progress in Material Handling Research*, College Industry Council on Material Handling Education (CICMHE).
- Roy, D., Krishnamurthy, A., Heragu, S.S., Malmborg, C.J., 2012. Performance analysis and design trade-offs in warehouses with autonomous vehicle technology. *IIE Trans.* 44 (12), 1045–1060.
- Roy, D., Krishnamurthy, A., Heragu, S.S., Malmborg, C.J., 2014. Blocking effects in warehouse systems with autonomous vehicles. *IEEE Trans. Autom. Sci. Eng.* 11 (2), 439–451.
- Viswanadham, N., Narahari, Y., 1992. Performance modeling of automated manufacturing systems. Englewood Cliffs, N.J.: Prentice Hall.
- 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 [Online; accessed 2018-04-15].
- Wurman, P.R., D'Andrea, R., Mountz, M., 2008. Co-ordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine* 29 (1), 9–20.
- Yuan, R., Graves, S.C., Cezik, T., 2018. Velocity based storage assignment in semi-automated storage systems. *Prod. Oper. Manage.* <https://doi.org/10.1111/poms.12925>.
- Yuan, Z., Gong, Y., 2017. Bot-in-time delivery for robotic mobile fulfillment systems. *IEEE Trans. Eng. Manage.* 64, 83–93.
- Zou, B., Gong, Y., Xu, X., Yuan, Z., 2017. Assignment rules in robotic mobile fulfillment systems for online retailers. *Int. J. Prod. Res.* 55 (20), 6175–6192.
- Zou, B., Xu, X., Gong, Y., De Koster, R., 2018. Evaluating battery charging and swapping strategies in a robotic mobile fulfillment system. *Eur. J. Oper. Res.* 267 (2), 733–753.