Blocking Effects in Warehouse Systems with Autonomous Vehicles
Debjit Roy, Ananth Krishnamurthy, Sunderesh S. Heragu and Charles J. Malmborg

Abstract—Autonomous vehicle-based storage and retrieval system (AVS/RS) offers considerable flexibility with respect to throughput capacity in the transfer of unit loads in high density storage areas. AVS/RS relies on autonomous vehicles to provide horizontal movement within a tier and uses lifts to provide vertical movement between tiers. In these systems, vehicle blocking delays in the aisles and cross-aisles could significantly impact system throughput and transaction cycle times. In this research, protocols are developed to address vehicle blocking, and a semi-open queuing network model is proposed to analyze system performance and evaluate design trade-offs. A decomposition-based method is used to solve the queuing network and quantify the effect of blocking. This model is adopted to analyze the effect of varying tier configuration parameters such as number of storage locations, depth/width ratio, number of vehicles, and vehicle utilization on blocking delays. These insights are useful for design conceptualization using AVS/RS.

Note to Practitioners—Autonomous vehicle-based storage and retrieval (AVS/R) systems represent a promising technology to handle palletized unit loads in a warehouse. Due to potential flexibility offered by this technology, it is being investigated as an alternative technology to traditional crane-based systems in many applications. However, analytical models are required to help develop design insights that will allow practitioners to design efficient systems that satisfy throughput, cycle time and flexibility requirements of particular application. This paper develops efficient analytical models to evaluate system performance under various configurations. A distinguishing aspect of this research is that it captures the effect of vehicle blocking on system performance. The results show that blocking effects could be significant in many cases and provide design insights that would be useful in minimizing blocking effects and improving system performance.

Index Terms—Autonomous vehicles, AVS/RS, AS/RS, blocking, semi-open queuing network, decomposition.

I. INTRODUCTION

With increasing importance of supply chain flexibility, distribution centers need to be more efficient and responsive to varying customer demands. Distribution centers are moving beyond traditional crane-based automated storage/retrieval system (AS/RS) technologies towards autonomous vehicle-based storage and retrieval (AVS/R) technologies. This technology offers additional flexibility in warehouse operations because throughput capacity can be varied by changing the number of vehicles in the system.

A system view of a warehouse that is operated using autonomous vehicle-based storage and retrieval systems (AVS/RS) is shown in Figure 1(a). The main components of an AVS/RS are lifts, autonomous vehicles, and a system of rails in the rack area. Lifts provide vertical movement (along z-axis) between tiers (Figure 1(b)) whereas autonomous vehicles provide horizontal movement (along x- and y-axis) within a tier using rails (Figure 1(c)). A tier of the storage area is composed of a set of aisles with storage racks on both sides of each aisle and a cross-aisle that runs orthogonal to the aisles. A vehicle travels between aisles using the cross-aisle rail (Figure 2). The throughput and transaction cycle times of an AVS/RS are influenced by system sizing decisions as well as operational decisions. System sizing decisions typically relate to the tier configuration parameters such as depth/width ratio, number of vehicles and lifts, location of the cross-aisle, number of zones, and location of load/unload points, whereas the operational decisions include vehicle assignment...
rule, type of command cycle, transaction scheduling policy, storage policy and choice of dwell points. Several queuing models have been developed to aid design engineers during the conceptualization stage of system development and to determine the impact of design parameters on throughput capacity and cycle times for storage and retrieval transactions (see [1], [2], [3], [4]). These models have provided insights on the impact of tier configuration and sizing decisions on expected cycle times; however, analyzing the effect of vehicle blocking on expected cycle times has received limited attention. Since blocking delays are significant within tiers, this paper focuses on understanding blocking effects within a single tier.

Blocking delays in a tier could either occur at the cross-aisle or in the aisles when multiple vehicles attempt to use the same resource. When vehicles interfere, vehicles must wait till the path to the destination location is cleared. Vehicle blocking could occur at three locations within a tier: on the cross-aisle, within an aisle, and at the intersection of an aisle and cross-aisle. Vehicle blocking could occur at the cross-aisle if one vehicle is traveling on the cross-aisle and another vehicle attempts to travel on the cross-aisle in the opposite direction. Vehicle blocking could also occur in the aisles if one vehicle is processing a transaction in the aisle and another vehicle enters the same aisle to process a transaction. A third scenario in which vehicle blocking occurs is at the intersection of an aisle and cross-aisle if one vehicle is waiting for the cross-aisle access at the end of an aisle and another vehicle attempts to enter this aisle. These blocking delays could significantly impact throughput capacity and cycle times. This research develops analytical models to investigate the effect of blocking delays on AVS/R system performance. The goal is to develop efficient models of single tier that can subsequently be integrated into models of multi-tier systems.

The overall approach is as follows. Blocking protocols are first developed to determine rules for usage of aisles and cross-aisles. Then a queuing network model is built to analyze system performance under these protocols. Since the resulting queuing network is a multi-class semi-open queuing network with class switching, exact solutions are hard. Consequently, the network is analyzed using a decomposition-based approach. The solution approach first estimates the conditional queue length distributions at the various nodes of the network that are subsequently linked to obtain the unconditional probability distributions and other performance measures of interest. Subsequently, blocking effects are studied for systems with varying tier configuration parameters such as depth/width (\( \frac{D}{W} \)) ratio, number of locations, and number of vehicles. Numerical studies also investigate the effect of tier configuration, number of vehicles and system load on blocking delays.

The rest of this article is organized as follows. Section II reviews the related literature. Section III describes the system, protocols for vehicle blocking, modeling assumptions, and the queuing model. Section IV explains the decomposition-based approach to solve the queuing network and defines the performance measures. Section V describes numerical experiments and design insights. Conclusions are summarized in Section VI.

II. LITERATURE REVIEW

This section first reviews existing design conceptualization models for AVS/RS. Recently, several analytical conceptualization models have been developed to analyze vehicle-lift interfaces and their effect on cycle times in AVS/RS. A common goal of these models is to analyze the effect of rack configuration (the number of rack tiers, aisles and columns), storage capacity, and the number of vehicles and lifts on cycle times and throughput under random storage policy and FCFS transaction scheduling. The models differ in their methodology, computational performance, and numerical accuracy.

Malmborg [5] develop a state equation-based model of AVS/R system and compare its performance with AS/RS. In a subsequent study, Malmborg [1] adapt the state equation-based approach to model interleaving dynamics in AVS/RS and determine performance measures. As state equations are computationally expensive, Kuo et al. [2] present a nested queuing model where the queuing dynamics between vehicles and transactions is modeled using an \( M/G/V \) queue and the dynamics between transactions/vehicles and lift is modeled using a \( G/G/L \) queue where \( V \) and \( L \) are the number of vehicles and lifts in the system respectively. Fukunari and Malmborg [3] develop a queuing network model to account for the time spent in the material flow interfaces. To minimize the errors in transaction waiting time for resources, Zhang et al. [6] develop a model that dynamically adjusts approximations based on the variance of the transaction inter-arrival times observed in a system. This model is developed for the situation where the number of vehicles equals the number of lifts. Ekren and Heragu [7] perform a simulation based regression analysis to determine rack configuration of an AVS/RS. Ekren et al. [8] develop a simulation model to identify the design parameters.
that influence the performance of AVS/RS. Heragu et al. [9] propose an open queuing network approach and use an existing tool called Manufacturing Performance Analyzer (MPA) to analyze the performance of AVS/RS. They also compare the AVS/RS performance with AS/RS. Mahadevan and Narendran [10] developed a two-step analytical approach to evaluate the performance of a material handling system. While the first step uses a rough-cut analysis to estimate the number of material handling units, the second step uses a $G/G/c$ queue to obtain operational performance measures. Zhang et al. [11] extend their earlier work to explicitly account the effect of lift dynamics on system performance. They develop a computational algorithm to model non-exponential lift service times when the number of lifts and vehicles are both design variables using a $G/G/L/V$ queue. The analytical models discussed so far are developed for handling unit-loads. [12] develop open queuing network models for handling product totes using AVS/RS.

Although these recent efforts provide valuable insights on performance of AVS/RS, none of these studies focus on blocking effects. While this research is an initial attempt to investigate blocking effects in AVS/RS, blocking effects have been extensively studied in Automated Guided Vehicle (AGV)-based material handling systems. AGVs use travel guide-paths similar to AVS/RS, and are commonly used to transfer pallet loads from one station to another in manufacturing shops. Strategies used to prevent collisions and blocking in AGV systems include use of a better routing algorithm, use of segmented flow topology (SFT) configurations, use of forward sensing and backtracking to avoid imminent collisions, enforcement of zone control, and extensive route pre-planning (see [13], [14], [15], [10], [16], and [17]). A comprehensive review of collision prevention strategies in AGV systems can be found in Le-Anh and De Koster [18].

However, the design of systems with AGVs differs from AVS/RS in two main aspects. In AGVs with single-loop configuration, vehicles usually travel in one unidirectional loop without any shortcut or alternative routes ([18]). However, AVS/RS systems do not have loops and vehicles use bi-directional cross-aisles and aisles to process transactions. Further, in AGVs with tandem configuration, guide-path system contains multiple nodes. Typically, only one vehicle serves each node and transfer stations are used to interface between nodes ([19]). However, in AVS/RS, autonomous vehicles process transactions in a single pass. Moreover, a zone in AVS/RS may require more than one vehicle to meet throughput requirement. Due to these differences between AVS/RS and AGVs design, new blocking models need to be developed to analyze blocking effects in AVS/RS systems.

III. SYSTEM DESCRIPTION AND QUEUING MODEL

This section first describes the assumptions and basics of system operation. Then, the blocking protocols and semi-open queuing network model used to analyze vehicle blocking in AVS/RS are discussed. Finally, service time expressions for AVS/RS with blocking are derived.

A. System Description

In an AVS/R system, a tier of a storage area is composed of a set of aisles with storage racks on both sides of each aisle. A cross-aisle is located at the end of the tier and it runs orthogonal to the aisles. Therefore, vehicles travel between aisles using the cross-aisle. A system of rails guides the rectilinear movement of vehicles along $x$ and $y$ dimensions. The load/unload (LU) point is located at the middle of the cross-aisle. In other words, the LU point divides the cross-aisle into two equal segments (CA$_R$ and CA$_L$: corresponding to the right and left segment of the cross-aisle). The vehicle dwells at the LU point (Figure 3). This implies that a vehicle that completes a retrieval transaction dwells at the LU point. After a vehicle completes a storage transaction, it travels to the LU point to serve the next transaction. The scope of this paper is limited to systems that execute single-command cycles only, i.e., storage and retrieval cycles are not combined. Further, it is assumed that all vehicles are pooled i.e., any free vehicle can process any type of transaction. Without loss of generality, the number of aisles in the tier is assumed to be even.

![Fig. 3. System description (top view of a single tier)](image)

In an AVS/R system, transaction cycle times consist of several components such as waiting time for vehicles and lifts, horizontal and vertical travel times, load/unload times, and vehicle blocking delays. If a vehicle originates from the LU point to process a transaction, then the components for storage and retrieval transactions for vehicle travel on a single tier can be written as follows:

$$CT_v = W_T + L_t + W_{cla} + \frac{|x_{lu} - X_v|}{h_v} + \frac{|y_{lu} - Y_v|}{h_v} + W_{at}$$

(1)

$$CT_r = W_T + W_{cla} + \frac{|X_{lu} - X_r|}{h_v} + \frac{|Y_{lu} - Y_r|}{h_v} + L_t + W_{at} + \frac{|Y_{lu} - Y_{lu}|}{h_v} + W_{crk} + \frac{|X_r - X_{lu}|}{h_v} + U_t$$

(2)

The notations used in the cycle time expressions are explained in Table I. Note that the cycle time of a transaction is composed of three components: waiting time to access a
free vehicle \((W_v)\), blocking delays at the cross-aisle and aisle \((W_{crk}, W_{ais}, W_{a}),\) and travel time components. The blocking delays at aisles are distinguished between storage and retrieval transactions \((W_{a} \text{ for retrieval and } W_{a} \text{ for storage})\) because the processing of a storage transaction is complete after the vehicle unloads the pallet at an aisle location. After processing the storage transaction, the vehicle returns to the LU dwell point and the time taken for this return travel is given by \(W_{a} + \frac{Y-L}{v_h} + W_{crk} + \frac{X_u-X_s}{v_h}\). However, this time is not included in Equation 1 because from a storage transaction perspective, the cycle time ends with unloading the pallet at the storage location. Note that the waiting time component and the blocking delays cannot be ascertained without explicitly modeling the stochastic interactions between storage and retrieval transactions on the tier. These delays depend on several factors such as number of vehicles in the system, transaction arrival rate, and tier configuration parameters. Therefore, a queuing network model is developed to quantify blocking delays, vehicle utilization, and cycle times.

### TABLE 1

NOTATIONS FOR THE TERMS USED IN CYCLE TIME EXPRESSIONS

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(V)</td>
<td>Number of vehicles in the system</td>
</tr>
<tr>
<td>(CT_s, CT_r)</td>
<td>Cycle time to complete storage and retrieval transaction</td>
</tr>
<tr>
<td>(W_V)</td>
<td>Waiting time to access a free vehicle</td>
</tr>
<tr>
<td>(W_{eu})</td>
<td>Waiting time to access the cross-aisle from the LU point</td>
</tr>
<tr>
<td>(W_{ek})</td>
<td>Waiting time to access the cross-aisle from the end of aisle</td>
</tr>
<tr>
<td>(W_{a}, W_{a})</td>
<td>Waiting time due to blocking within an aisle for storage and retrieval transaction</td>
</tr>
<tr>
<td>(X_{iu}, X_u)</td>
<td>x and y coordinates of LU point</td>
</tr>
<tr>
<td>(X_r, Y_r)</td>
<td>x and y coordinates of retrieval location</td>
</tr>
<tr>
<td>(X_s, Y_s)</td>
<td>x and y coordinates of storage location</td>
</tr>
<tr>
<td>(h_l)</td>
<td>Horizontal velocity of a vehicle</td>
</tr>
<tr>
<td>(L_{s}, L_{t})</td>
<td>Load and unload time</td>
</tr>
<tr>
<td>(N)</td>
<td>Number of nodes</td>
</tr>
<tr>
<td>(D)</td>
<td>Depth of the tier</td>
</tr>
<tr>
<td>(W)</td>
<td>Width of the tier</td>
</tr>
<tr>
<td>(u_d)</td>
<td>Unit depth of the rack</td>
</tr>
<tr>
<td>(w_d)</td>
<td>Width of an aisle</td>
</tr>
<tr>
<td>(x_{ia})</td>
<td>X-way distance from cross-aisle to the start of the aisle</td>
</tr>
</tbody>
</table>

### B. Protocols for Vehicle blocking

As mentioned in Section 1, in an AVS/RS, vehicle blocking within a tier can occur at three locations: 1) within a cross-aisle, 2) within an aisle, and 3) at the intersection of an aisle and cross-aisle (Figure 4). In this research, a set of protocols is developed to model vehicle blocking. These protocols are described below.

**Cross-aisle Protocol:** While processing storage and retrieval transactions, vehicles travel from Load/Unload (LU) point to an aisle using the cross-aisle rail and return to the LU point after processing the transaction. Therefore, vehicles access the cross-aisle segment twice for processing each transaction. Since the cross-aisle can permit only vehicles traveling in one direction at any point in time, to access the cross-aisle, vehicles might be blocked and have to wait either at the LU point or at the end of an aisle (EOA). If multiple vehicles are waiting at the LU point and EOA to use the cross-aisle simultaneously, blocking on the cross-aisle is handled by using a simple switching protocol for cross-aisle usage. Vehicles at the LU point and at the EOA get alternate use of the cross-aisle; and the cross-aisle is seized by vehicles traveling from the LU point (EOA) to the aisles (LU point) till all the vehicles complete their travel along the cross-aisle. Note that this protocol permits simultaneous usage of the cross-aisle by multiple vehicles traveling in the same direction on the cross-aisle.

**Aisle Protocol:** A vehicle servicing a transaction within an aisle could be blocked by another vehicle entering the same aisle. In the aisle protocol, the former vehicle yields to other incoming vehicles. In other words, it is assumed that the former vehicle travels to the last available bay location and waits until the latter vehicle completes its operation. Note that under this protocol, a vehicle could be blocked multiple times during its service within an aisle, if other vehicles enter this aisle before the former vehicle completes its aisle service.

**Intersection Protocol:** A vehicle waits for cross-aisle access after completing its service within an aisle. This vehicle could be blocked at the intersection if another vehicle attempts to enter the same aisle. To avoid vehicle blocking at the intersection, after completing service within an aisle, vehicles leave the aisle and wait in the rack in front of the aisle to access the cross-aisle (see Figure 4).

### C. Queuing Network Model

Based on these protocols, a queuing network model for the single tier is developed as shown in Figure 5. The vehicles in the tier processing storage and retrieval transactions are modeled as resources. Vehicles are assumed to travel with a constant horizontal travel velocity and the acceleration/deceleration effects are ignored. Further, the vehicle assignment rule is random i.e., any idle vehicle is equally likely to be chosen to process a transaction. The storage and retrieval transaction arrivals are assumed to follow a Poisson process with rates \(\lambda_s\) and \(\lambda_r\) respectively. These transactions are served based on FCFS scheduling discipline.

The number of nodes in the semi-open queuing network is \(N + 4\), where \(N\) nodes correspond to the \(N\) aisles, nodes \(N + 1\) and \(N + 2\) correspond to two cross-aisle segments, \(CA_k\) and \(CA_{L}\) respectively, node \(N + 3\) corresponds to the LU location, and the last node \(N + 4\) corresponds to the synchronization station where vehicles and transactions are matched together. Due to distinct routings of storage and retrieval transactions, vehicles processing storage and retrieval transactions are assumed to belong to class \(s\) and class \(r\) respectively. A vehicle belonging to class \(s\) can switch to a vehicle belonging to class \(r\) by processing a retrieval transaction.
When a storage transaction arrives, it is matched with a free vehicle (if available) at the synchronization node, \( N + 4 \). Then the vehicle picks the load at LU point node \( N + 3 \) and the vehicle waits for using either left cross-aisle (node \( N + 1 \) or right cross-aisle (node \( N + 2 \)), depending on its destination. The routing probabilities to the nodes \( N + 1 \) and \( N + 2 \) are 0.5 each. After completing the cross-aisle travel, the vehicle proceeds towards an aisle (node 1-node \( N \)) for storing the load. Note that the routing probability from the cross-aisle to an aisle is \( \frac{2}{N} \). After unloading the pallet, the empty vehicle begins its return travel to dwell at the LU point. The vehicle completes its travel on the aisle, waits for the cross-aisle, and travels on the cross-aisle to dwell at the LU point. Upon reaching the LU point, the free vehicle queues in buffer \( B_2 \). Similarly, when a retrieval transaction arrives, it is matched with a free vehicle (if available) at node \( N + 4 \) and waits for either left or right crossaisle usage (node \( N + 1 \) or node \( N + 2 \)). After completing the cross-aisle travel, the vehicle moves to an aisle depending on its destination, loads the pallet, and uses the cross-aisle again and finally unloads the pallet at the LU point (node \( N + 3 \)). After unloading, the idle vehicle is released and queues in buffer \( B_2 \). It is assumed that storage and retrieval locations are uniformly distributed within the tier and therefore, a free vehicle is equally likely to be assigned a transaction in any of the aisles.

Based on the blocking protocols for aisles, a vehicle in the aisle yields to other incoming vehicles. Therefore, the latter vehicle in the aisle gets priority for service over the former vehicle and the former vehicle resumes its service only when the latter vehicle completes its LU operation. This vehicle interruption phenomenon within an aisle is captured by modeling each aisle queue as a Last Come First Serve - Preemptive Resume (LCFS-PR) queue. The protocol for usage of the cross-aisle can be modeled by treating each cross-aisle segment (left and right) as a gated polling queue with zero switching times and two queues corresponding to vehicles waiting to use the cross-aisle to travel i) from LU point to the aisles and ii) from the end of aisle to the LU point.

The two queues correspond to buffers where vehicles either wait to use the cross-aisle to travel from the LU point to the aisles, or from the aisles back to the LU point. During service, the cross-aisle resource polls and serves the two queues in a cyclical order. Since the server serves only those vehicles that are already in queue at the polling instant, the transactions that arrive at the cross-aisle queue during the service to that queue wait and are only served during the next polling cycle (for further details on gated polling queues, see [20]). Finally, queuing at the LU point is modeled using an infinite server station. Next, the approach used to determine the service times at the different nodes of the queuing network is discussed. The notation \( D = (D - 2w_a - w_u) \) denotes the maximum travel distance of a vehicle along the entire length of the cross-aisle. Since load and unload times are assumed to be equal, \( \mu_{LU}^{-1} \) corresponds to the expected load (or unload) time.

\[
\mu_{LU}^{-1} = L_t = U_t \tag{3}
\]

The travel time in the aisle can be represented by a random variable \( X = X_1 + X_2 + X_3 + X_4 + X_5 \). \( X_1 \) and \( X_3 \) correspond to the time to travel the X-way; i.e., from the cross-aisle to the first bay location and back, respectively. This time is deterministic and each term equals to \( \frac{5}{2h_v} \). \( X_2 \) and \( X_4 \) correspond to the travel into and out of the aisle i.e. from the first bay location to the storage/retrieval location and back, respectively. This time has a uniform distribution with mean \( \frac{W}{2h_v} \). \( X_3 \) corresponds to the load (or unload) time and is a constant value equal to \( L_t \) (or \( U_t \)). It is assumed that \( L_t = U_t \). Let \( \mu_{A_i}^{-1} \) denote the expected service time at the aisle node \( i \) for both storage and retrieval vehicle classes. Then,

\[
\mu_{A_i}^{-1} = \frac{W}{h_v} + \frac{2x_w}{h_v} + L_t \text{ for } i = 1 \ldots N \tag{4}
\]

The service times at the polling queues corresponding to the cross aisle segments involves an approximation which is described in the next section.

Based on the description of the stations, resources, routing, and service protocols at the different stations of the network, it is clear that the queuing network model described above is a semi-open network, i.e. the network possesses the characteristics of both open as well as closed queuing networks. The network behaves like an open network because there is no limit on the number of transactions permitted to wait at buffer \( B_1 \). However, the network also behaves like a closed network because the number of number of vehicles is fixed and transactions need to wait for a vehicle to be free before being processed. Note that for the network to be stable, the transaction arrival rate \( \lambda_s + \lambda_r \) should be less than the throughput of the closed queuing network with \( V \) vehicles. Further, the semi-open queuing network described above does not possess product form solutions and solving the network in its original form would require complex analysis. Therefore, a decomposition-based approach is developed to solve the network and estimate desired performance measures.

IV. DECOMPOSITION APPROACH

The decomposition approach is based on solution of two sub-networks that are defined based on the number of items...
in buffers $B_1$ and $B_2$. Let $y$ denote the difference in the number of entities in buffers $B_1$ and $B_2$ at any point in time. Then in the network, when $y \leq 0$ a free vehicle waits in buffer $B_2$ for transactions, and when $y \geq 0$, transactions wait in buffer $B_1$ for free vehicles. The decomposition approach considers these two scenarios separately and defines a closed queuing network corresponding to the case $y \leq 0$, and an open queue corresponding to the case of $y \geq 0$, respectively. Note that when $y > 0$, another closed queuing network is defined to study the distribution of all busy vehicles. The case of $y = 0$ is common to both scenarios. The approach then consists of three steps: i) solve the closed queuing network corresponding to the case $y \leq 0$ to obtain conditional measures, ii) solve the open queue corresponding to the case $y \geq 0$ and the closed queuing network corresponding to the case $y > 0$ to obtain conditional measures, and iii) link results from step 1 and step 2 to obtain the unconditional performance measures of the original network. The details of the steps adopted in the decomposition approach and expressions to determine performance measures are provided in the following subsections.

A. Solving the Closed Queuing Network ($y \leq 0$)

When $y \leq 0$, idle vehicles wait for transactions and the system is analyzed as a closed queuing network with two classes of vehicles (Figure 6a). A vehicle belonging to the storage class $s$ (retrieval class $r$) could switch to a retrieval (storage) transaction. As mentioned earlier, the closed queuing network obtained from the decomposition approach does not have a product form solution. Therefore, to simplify the analysis, an approximation is developed to model the complex dynamics of a gated polling queue. This approximation along with the solution approach is explained in the following subsections.

1) Cross-aisle Approximation: As mentioned earlier, the operation of each segment of the cross-aisle (left and right) is modeled using a gated polling queue with zero switching times. For the left cross-aisle segment ($CA_L$), the service time parameters, $\mu_{CA_L} (LU, A)^{-1}$ and $\mu_{CA_L} (A, LU)^{-1}$, correspond to the expected travel time by a batch of vehicles traveling from the LU point to the end of an aisle and expected travel time by a batch of vehicles traveling from the end of an aisle to the LU point respectively. Precise estimation of the delays experienced by the vehicles for travel on the cross-aisle would require estimation of these service parameters based on the destinations of the individual vehicles in each batch. To simplify the analysis, the gated polling queue corresponding to each cross aisle segment is approximated by an Infinite Server (IS) queue. The mean service times, $\mu_{CA_L}$ and $\mu_{CA_R}$ at the two Infinite Server (IS) queues correspond to the mean delay experienced by vehicles potentially waiting for access to the cross-aisle and then traveling on the cross-aisle. These mean delays are estimated using probabilistic arguments. For example, the mean service time at the IS station ($\mu_{CA_L}^{-1}$) is provided by Equation 5. Note that IS approximation simplifies the analysis of the cross-aisle, but it does capture the key queuing delays in the cross-aisle (Refer Equation 5); and our numerical experiments indicate that this approximation is quite effective.

$$\mu_{CA_L}^{-1} = (1 - \rho_{CA_L}) \left( \frac{D}{4h_v} + \frac{\rho_{CA_L}}{2} \left( \frac{5D}{8h_v} \right) \right) + \frac{\rho_{CA_L}}{2} \left( \frac{3D}{8h_v} \right)$$

Note that in Equation 5, the term $\rho_{CA_L}$ is equal to $(\lambda_s + \lambda_r) \left( \frac{D}{4h_v} \right)$ and represents the probability that the left segment of the cross-aisle is busy (in use). Note that a free vehicle arriving at the cross-aisle queue would find the cross-aisle in one of three states: i) idle, ii) serving vehicles from its queue, or iii) serving vehicles from the other queue. The probability of the cross-aisle being in one of these three states is given by $(1 - \rho_{CA_L})$, $\left( \frac{\rho_{CA_L}}{2} \right)$, and $\left( \frac{\rho_{CA_L}}{2} \right)$, respectively. The term $\left( \frac{D}{4h_v} \right)$ corresponds to the mean delay when the vehicle arriving at the cross-aisle finds it idle (not in use). The term $\left( \frac{5D}{8h_v} \right)$ corresponds to the mean delay when an arriving vehicle finds the cross-aisle already processing vehicles from its queue. In this case, the vehicle has to wait for the current cycle to complete $\left( \frac{D}{4h_v} \right)$, then wait for the next cycle when the vehicles from the other queue is processed $\left( \frac{D}{4h_v} \right)$, and then gets serviced at the cross-aisle.
\begin{align*}
\left( \frac{\partial}{\partial t} \right) \text{corresponds to the mean delay for an arriving vehicle that finds the server processing vehicles from the other queue. This term is composed of two components, mean wait time for the current cycle to complete } \left( \frac{\partial}{\partial t} \right) \text{and the mean time to get serviced at the cross-aisle } \left( \frac{\partial}{\partial t} \right). \text{ Using a similar reasoning an expression for } \mu_{\text{CA} \theta}^{-1} \text{ is also obtained. Note that, since the storage policy is random, the distribution of travel time on the cross-aisle follows a uniform distribution denoted by } U[0, \frac{\partial}{\partial t}]. \text{ This implies that the mean delay estimated by Equation 5 is an under-estimate of the actual delay when the average number of vehicles that simultaneously access the cross-aisle is significantly greater than one. However, as the numerical studies reported later show, the simple approximation in Equation 5 does yield reasonably accurate performance estimates. Next, the solution algorithm is discussed.}
\end{align*}

2) Solution Algorithm: Another motivation for the particular approximation for the polling queue described in the previous section is that the closed queuing network now becomes a product-form (BCMP) network (Figure 6b). The aisle nodes \((1, \ldots, N)\) are of type LCFS-PR whereas the cross-aisle \((N+1, N+2)\) and LU \((N+3)\) nodes are of type IS. At node \(N+4\), idle vehicles wait for transaction arrivals. This node is referred as the ‘wait for transaction’ node. Since transactions arrive according to a Poisson process, the service time at this node has an exponential distribution with mean \((\lambda_s + \lambda_r)^{-1}\) and transactions are processed in an FCFS fashion. Based on model assumptions, the closed queuing network has product form (\([21]\)). The performance of the network is evaluated using the convolution algorithm (\([22]\)). The procedure first converts the multi-class network with class switching into a single equivalent class queuing network and determines class-specific performance measures based on approaches described in \([23]\) and \([24]\).

From the solution of this network, the class-specific marginal probabilities at the nodes, \(\pi_{j,c}(m)|y \leq 0\), where \(j = 1, \ldots, N+4\), \(m = 0, \ldots, V\), and \(c \in \{s, r\}\) are obtained. The conditional probabilities of finding \(i\) idle vehicles at node \(N+4\), \(\pi(y = i|y \leq 0)\), where \(i = 0, \ldots, -V\), are obtained from the class-specific marginal probabilities at the nodes, \(\pi_{j,c}(m)|y \leq 0\), as shown in Equation 6.

\begin{align*}
\pi(y = i|y \leq 0) &= \sum_{k,l:k+l=−i} ((\pi_{N+4,s}(k)|y \leq 0) \\
&+ (\pi_{N+4,r}(l)|y \leq 0)) \text{ for } k,l = 0, \ldots, V(6)
\end{align*}

Finally, class-specific conditional mean queue length, \(Q_{j,c}|y \leq 0\), where \(j = 1, \ldots, N+4\) and \(c \in \{s, r\}\) at all nodes are obtained.

B. Solving the Open Queue \((y \geq 0)\)

When \(y \geq 0\), transactions wait for free vehicles and an arriving transaction is matched with the first available vehicle. The system is analyzed as a single server queue with Poisson arrivals with parameter \(\lambda_s + \lambda_r\) (Figure 7a). The challenge involved in solving this queue is in determining the service time at this queue. The service rate, \(\mu_T\) denotes the rate at which an idle vehicle becomes available for processing transactions. However, \(\mu_T\) depends on the distribution of the busy vehicles in the aisles and cross aisles. In order to determine \(\mu_T\), another closed queuing network (Figure 7b) is solved. The queuing network in Figure 7b is identical to the closed queuing network developed for the case \(y \leq 0\) (Figure 6b) except that it does not include the wait for transaction node \(N+4\). This is because when a transaction is waiting for a free vehicle, all vehicles are busy and as soon as a vehicle becomes free, it is immediately reassigned to process a storage or a retrieval transaction. Note that this new network models the scenario where \(y > 0\) because a vehicle never waits for a transaction arrival. Based on the queuing network in Figure 7b, the service rate \(\mu_T\) is approximated by the throughput of this closed queuing network. Note that the queuing network Figure 7b is also of the product form type, and consequently, the network is solved using the approach described in the previous subsection. From the solution of this network, the class-specific marginal probabilities, \(\pi_{j,c}(m)|y > 0\), where \(j = 1, \ldots, N+3\) and \(c \in \{s, r\}\) at all nodes are obtained.

To precisely determine the distribution of transactions waiting for free vehicles, higher moments of the service time at the single server queue (Figure 7a) are required. However, determining higher moments of the service time distribution (beyond the mean) of the open queue requires complex analysis of the departure process of the closed queuing network in
which all vehicles are busy (Figure 7b). Specifically, we need higher moments of the transaction inter-departure times from various nodes, such as $N + 1$, $N + 2$, $N + 3$ and so on, which in itself presents a significant challenge. Therefore, simulation experiments were performed to analyze the higher moments of inter-departure times from the network and the results revealed that the service time distribution has low variance (with squared coefficient of variation (SCV) in the range of 0.1-0.2). Although such SCV could be modeled using an $M/E_1/1$ queue with suitable number of phases, we choose to approximate the open queue as an $M/D/1$ queue. Our numerical results show that this approximation yields fairly good results. The mean queue length and steady state probabilities, $\pi(y = i|y \geq 0)$, are obtained using Pollaczek-Khinchin mean queue and LS transform formulas respectively (25). From the solution of the open queue, the conditional queue length of number of transactions waiting in buffer $B_1$, $Q_{B_1}|y \geq 0$ is determined.

C. Linking Results from Decomposition

The analyses in Sections IV-A and IV-B provide the conditional steady state distributions ($\pi(y = i|y \leq 0)$ and $\pi(y = i|y \geq 0)$) for vehicles and transactions in the network under conditions $y \leq 0$ and $y \geq 0$. To obtain the unconditional distributions, the results obtained from conditional analysis are linked using the law of total probability and the fact that the state $y = 0$ is common to both the analyses. Recognizing that $\pi(y = 0)$ is common, the following equation is obtained.

$$\pi(y = 0|y \geq 0)\pi(y \geq 0) = \pi(y = 0|y \leq 0)\pi(y \leq 0) \quad (7)$$

The two unknowns in Equation 7 are $\pi(y \geq 0)$ and $\pi(y \leq 0)$. These two unknowns are obtained as follows. First, it is noted that $\pi(y = 0|y \geq 0)$ is equal to $1 - \rho$, where $\rho = (\lambda_s + \lambda_r)/\mu_r$ is the utilization of the $M/D/1$ queue. Therefore, Equation 8 is obtained by substituting the terms $\pi(y = 0|y \geq 0)$ in Equation 7.

$$(1 - \rho)\pi(y \geq 0) = \pi(y = 0|y \leq 0)\pi(y \leq 0) \quad (8)$$

Further, as $\sum_{k=-\infty}^\infty \pi(y = k) = 1$, the following equation is obtained:

$$\pi(y \leq 0) + \rho \pi(y \geq 0) = 1 \quad (9)$$

Equations 8 and 9 are solved to obtain the two unknowns $\pi(y \leq 0)$ and $\pi(y \geq 0)$. Knowing $\pi(y \leq 0)$, $\pi(y \geq 0)$, $(\pi(y = i|y \leq 0))$, and $(\pi(y = i|y \geq 0))$, $\pi(y = i)$ for the case $y \leq 0$ and $y \geq 0$ is determined from Equations 10 and 11.

$$\pi(y = i) = (\pi(y = i|y \leq 0))\pi(y \leq 0) \quad \text{for } i = 0, \ldots, -V \quad (10)$$

$$\pi(y = i) = (\pi(y = i|y \geq 0))\pi(y \geq 0) \quad \text{for } i = 0, \ldots, \infty \quad (11)$$

Solving Equations 10 and 11, the unconditional steady state distributions and all key performance measures such as the vehicle utilization, distribution of idle vehicles, and average number of vehicles (transactions) waiting for transactions (vehicles) are obtained.

D. Estimating Performance Measures

The solution of the semi-open queuing network model provides the average number of class $c$ vehicles present at node $i$ ($Q_{c,i}$, where $i \in \{1, \ldots, N + 4\}$, and $c \in \{s, r\}$). The average number of vehicles of class $c$ at node $i$ ($Q_{c,i}$) is determined by the Equation 12.

$$Q_{c,i} = (Q_{c,i}|y \leq 0)(\pi(y \leq 0)) + (Q_{c,i}|y > 0)(\rho \pi(y \geq 0)) \quad (12)$$

The main performance measures of primary interest are vehicle utilization ($U_V$), expected storage and retrieval cycle time ($E[CT_s]$ and $E[CT_r]$), average number of transactions waiting for service in the buffer $B_1$ ($Q_{B_1}$), and expected blocking delay times ($E[OBD_s]$, $E[OBD_r]$). The average number of idle vehicles at buffer $B_2$ ($Q_{B_2}$) is obtained by determining the average number of vehicles of both storage and retrieval classes at node $N + 4$ (Equation 13).

$$Q_{B_2} = (Q_{N+4,s}|y \leq 0) + (Q_{N+4,r}|y \leq 0)\pi(y \leq 0) \quad (13)$$

Using the value of $Q_{B_2}$, the fraction of idle vehicles is written as $Q_{B_2}/V$. Vehicle utilization, or the fraction of busy vehicles, is written as

$$U_V = 1 - \frac{Q_{B_2}}{V} \quad (14)$$

The average number of transactions waiting for service ($Q_{B1}$) is determined using Pollaczek-Khinchin mean queue length formula for $M/D/1$ queue. The mean queue length is conditional upon the case $y \geq 0$ ($Q_{B1}|y \geq 0$). To obtain unconditional value ($Q_{B1}$), the conditional result is multiplied with $\pi(y \geq 0)$ as shown in Equation 15.

$$Q_{B1} = (\rho + \frac{\rho^2}{2(1 - \rho)}) \pi(y \geq 0) \quad (15)$$

The expected storage and retrieval cycle times ($E[CT_s]$ and $E[CT_r]$) are determined from the mean queue lengths and by applying Little’s Law (Equations 16 and 17). In these equations, the first component of the cycle time is the average wait time to obtain a free vehicle, whereas the second component denotes the transaction processing time.

$$E[CT_s] = \frac{Q_{B1}}{\lambda_s + \lambda_r} + \frac{\sum_{i=1}^N Q_{1,s} \alpha + \sum_{i=N+1}^{N+2} Q_{i,s}}{\lambda_s} \quad (16)$$

$$E[CT_r] = \frac{Q_{B1}}{\lambda_s + \lambda_r} + \frac{\sum_{i=1}^{N+3} Q_{i,r}}{\lambda_r} \quad (17)$$

Note that, the storage transaction is termed completed when the pallet is unloaded at the storage location. Therefore, to estimate the number of transactions in the aisle processing storage transaction, a term $\alpha$ is defined as the ratio between the time spent by a vehicle until the pallet is stored within an aisle to the total time spent by the vehicle within an aisle ($\frac{t_{st}}{t_{st} + t_{st} + t_l}$). Then, the term $Q_{i,s} \alpha$ provides an estimate of the average number of vehicles processing storage transactions within an aisle.
E[(OBDS)] = E[CT] - \frac{Q_B}{\lambda_s + \lambda_r} - L_t - \frac{D'}{4h_v} \\
- \left( \frac{W}{2h_v} + \frac{x_w}{h_v} + U_t \right) 
\tag{18}

E[(OBDR)] = E[CT] - \frac{Q_B}{\lambda_s + \lambda_r} - 2D' \frac{2}{4h_v} - \left( \frac{W}{h_v} + \frac{2x_w}{h_v} + L_t \right) - U_t 
\tag{19}

The expected cross-aisle blocking delay (CABD) times for storage and retrieval transactions (E[CABDS] and E[CABDR]) are given by Equations 20 and 21. The cross-aisle is accessed only once during processing a storage transaction, therefore only half of the average number of storage class vehicles accessing the cross-aisle (\(\frac{Q_{N+1,s} + Q_{N+2,s}}{2}\)) is considered to determine E[CABDS]. On the other hand, the cross-aisle is accessed twice during processing a retrieval transaction, therefore all retrieval class vehicles on average accessing the cross-aisle (\(Q_{N+1,r} + Q_{N+2,r}\)) is considered to determine E[CABDR].

E[CABDS] = \frac{Q_{N+1,s} + Q_{N+2,s}}{2\lambda_s} - \frac{D'}{4h_v} \tag{20}

E[CABDR] = \frac{Q_{N+1,r} + Q_{N+2,r}}{2\lambda_r} - 2\frac{D'}{4h_v} \tag{21}

Finally, the percentage blocking delay at the aisle, cross-aisle and overall can be obtained by taking a ratio of the expected blocking delays to the expected transaction cycle times. For storage transactions, these ratios are expressed as E[ABDS] / E[CT], E[ABDR] / E[CT], and E[OBDS] / E[CT]. Similarly, the ratios for the retrieval transactions are expressed as E[ABDR] / E[CT], E[ABDS] / E[CT], and E[OBDR] / E[CT]. The next section describes the numerical experiments conducted to assess the accuracy of the decomposition approach and obtain design insights related to blocking effects in AVS/RS.

V. NUMERICAL EXPERIMENTS AND DESIGN INSIGHTS

A detailed 3D simulation model is built using AutoMod® software to understand the effect of blocking on system performance and analyze the effects of tier configuration parameters on blocking delays. Note that the simulation model explicitly captures the vehicle travel using a 3D path mover system and we make no additional assumptions on the vehicle travel times. To obtain insights on the effects of blocking, the following design of experiments is used. In the design, three tier configuration parameters, namely, the number of storage locations - Ns, number of vehicles - V, and \(\frac{D}{W}\) ratio are varied. The number of storage locations is set at three levels (4080, 7200, and 15000); the number of vehicles is also set at three levels (3, 5, and 10); the \(\frac{D}{W}\) ratio is set at two levels (0.5 and 1.5). For each scenario, the transaction arrival rates are varied and their impact on blocking delays, average cycle times, utilizations, and average number of transactions waiting are considered. The transaction rate is varied from 40 pallets/hr to 200 pallets/hr at increments of 5 pallets/hr. The utilization levels are studied between 60% and 90%. This design of experiments yields 105 scenarios. For each scenario, 15 replications are run with a warm-up period of at least 5,000 transactions and a run time of at least 24,000 transactions.

Performance measures (expected transaction cycle times, vehicle utilization, expected number of transactions waiting for service, and expected blocking delays) are obtained within 95% confidence level and the half-width of a confidence interval is less than 2% of the average. The average absolute error percentage is determined by the expression \((-\frac{W}{2h_v} \times 100)\), where A and S corresponds to the estimate of the measures obtained from analytical and simulation model respectively. A comparison of the four performance measures: \(Q_{B,i}, U_V, E[CT], E[CT],\) and \(E[CT] \) obtained from analytical and simulation models for a tier with \(N_s=4080, 7200, V=3, 5, 10,\) and \(\frac{D}{W}=1.5\) is shown in Appendix A. In addition, this table includes the percentage blocking delays at the cross-aisle, and overall for retrieval transactions (E[CABDR], E[ABDR], and E[OBDR]), which are used to develop blocking insights. A summary of the errors for all tier configurations included in the design of experiments in illustrated in Figure 8. The average absolute error percentages for the four measures: storage cycle time, retrieval cycle time, utilization, and number of transactions waiting are 2%, 3%, 0.9%, and 8% respectively. The results suggest that the decomposition approach yields fairly accurate performance estimates.

For example, on a standard PC (with Duo Core Processor, 2GHz), the simulation model takes about 15 -20 minutes to run 15 different replications for a single design scenario with 10 vehicles. However, for the same configuration, the queuing network model takes less than 30 seconds to run on the same PC. Hence, queuing network models could be very useful in quickly exploring the vast space of AVS/RS designs and identifying promising configurations.

A. Design Insights

The design insights captures the effect of design parameters such as storage locations, number of vehicles, vehicle utiliza-
As discussed earlier, delays due to blocking could occur at the cross-aisle or within the aisles. In order to understand the effect of blocking on cycle times and get an insight into the blocking delay times, the components of storage and retrieval cycle time are studied in depth. The different components of storage cycle time in a chronological sequence are wait for vehicle, load pallet, blocking delay at the cross-aisle, travel on the cross-aisle, blocking delay within the aisle, travel on the aisle, and unload pallet. Similarly, the different components of retrieval cycle time in a chronological sequence are wait for vehicle, blocking delay at the cross-aisle, travel on the cross-aisle, blocking delay within the aisle, travel on the aisle, load pallet, blocking delay at the cross-aisle, travel on the cross-aisle, and unload pallet. The components of cycle times are presented for a tier with 7200 storage locations and \( D_s = 1.5 \) (Figure 9). In this scenario, the vehicle utilization is approximately 86%. Note that the blocking delays at the cross-aisles are more significant than the blocking delays within the aisles. Blocking delay within an aisle depends on the number of vehicles interrupting a vehicle in an aisle during its service. Blocking delay on the cross-aisle depends on the length of cross-aisle segment and the number of vehicles waiting to access the cross-aisle. In this scenario, the number of vehicles is 10 and the number of aisles is 44. So it is less likely that a vehicle in an aisle is interrupted during its service. Because each vehicle accesses the cross-aisle segment twice during its service, it is more likely that a vehicle experiences delay before gaining access to the cross-aisle. Blocking delays account for 19.8% and 14.9% of the expected retrieval and storage cycle times respectively.

**Effect of Number of Vehicles:** It is expected that the likelihood of vehicle blocking at the cross-aisle and aisles increases with the increase in the number of vehicles. This observation can be confirmed from the results in Appendix A. With a fixed number of storage locations, \( D_s \) ratio, and vehicle utilization range, the percentage of blocking delay increases with increase in the number of vehicles. For example, for \( N_s = 4080, \ D = 0.5, \) and \( U_V \) between 87%-91%, it can be observed from cases 3, 6, and 9 that the percentage of overall blocking delays for 3, 5, and 10 vehicles are 3.1%, 6.7%, and 15.4% respectively.

**Effect of Number of Storage Locations:** The effect of blocking on cycle times is expected to reduce when the number of storage locations increases while keeping the vehicle count constant. If the \( D_s \) ratio of tiers is kept constant, reduction of blocking delay is expected because the number of aisles increases with the increase in the number of locations. Interestingly, the numerical results show that as the number of storage locations increase, the average percentage of blocking delays does not change significantly. For example, the range of percentage blocking delays with 10 vehicles for retrieval transactions are between 17%-20% for \( N_s = 4080 \) and between 20%-23% for \( N_s = 15000 \). This observation can be explained by considering the protocols used for blocking in the cross-aisle and the aisles. For different tier configurations, the delay at the cross-aisle is proportional to the travel times on the cross-aisle. This delay at the cross-aisle typically increases with an increase in the number of storage locations. However, as the number of storage locations increases, the probability of interruption of a vehicle during its service within an aisle decreases. Therefore, the increase in blocking delay at the cross-aisle is offset by a decrease in blocking delay within an aisle. For instance, from cases 18 and 36, it can be observed that for \( V = 10 \) and \( D = 1.5 \), by increasing \( N_s \) from 4080 to 15000, the percentage of blocking delays at the cross-aisle increases from 13.4%-17.9% but the percentage of blocking delays at the aisle decreases from 3.4% to 1.9%. Therefore, the net increase in the overall percentage blocking delays is 2.9%. Therefore, the average percentage of blocking delays does not vary significantly with the increase in the number of storage locations.

**Effect of Vehicle Utilization:** It would be expected that increase in vehicle utilization would lead to more frequent vehicle blocking at the cross-aisles and aisles. However, numerical results show that with increase in vehicle utilization, the effect of blocking delays on cycle time diminishes. For instance, in a tier with \( N_s = 15000, V = 10, D = 1.5, \) and \( U_V = 68\% \), the
percentage blocking delays for retrieval transactions is 22.6% (case 34) whereas the percentage of time waiting for a free vehicle is 4.2%. For $U_V=89\%$, the percentage blocking delays decreases to 19.7% (case 36) whereas the percentage of time waiting for a free vehicle is 25.6%. An increase in utilization, leads to an increase in vehicles waiting time which is more pronounced than the increase in blocking delays. Therefore, the overall effect of blocking decreases.

**Effect of $D/W$ Ratio:** With increase in the $D/W$ ratio of a tier, the number of aisles increases. Therefore, the amount of blocking delay at the aisles is expected to decrease. However, the effect of $D/W$ on overall blocking delays (summation of blocking delays at aisles and cross-aisle) is not straightforward. Figure 10 shows the effect of varying $D/W$ ratio on percent blocking delays. For $D/W=1.5$, the blocking delays at the cross-aisle are more pronounced than the blocking delays at the aisles; however, for $D/W=0.5$, the blocking delays at the aisles are more pronounced than the blocking delays at the cross-aisle. For retrieval transactions, the average percentage blocking delays for $D/W=0.5$ with 3, 5, and 10 vehicles are 3.6%, 7.3%, and 15.7 % respectively, whereas the average percentage blocking delays for $D/W=1.5$ with 3, 5, and 10 vehicles are 6.0%, 11.0%, and 20.6 % respectively. A tier with a larger $D/W$ could lead to a greater percentage of overall blocking delay because the percentage blocking delay on the cross-aisle is more significant than the percentage blocking delay on the aisles.

**Impact of Blocking on Performance Measures:** In this research, the performance of AVS/RIS is evaluated by a detailed model that considers the effect of blocking at the aisles and cross-aisle. To understand the value of this elaborate model with blocking, the performance measures (expected transaction cycle times, $E[CT]_b$ and vehicle utilization, $U_V$) from the elaborate model are compared against the measures obtained from another semi-open queuing model that does not consider blocking ($E[CT]_{wb}$ and $U_{Vwb}$). In the model without blocking, the travel times are aggregated and the service times are modeled using Infinite Server (IS) stations (see [4]). The performance measures are obtained for a tier configuration with $D/W=1.5$, 5 vehicles, and 7200 locations. From Figure 11, it can be seen that the model without blocking underestimates the vehicle utilization by 15%-18% from the estimate obtained from model with blocking. Due to this gap in the estimate of vehicle utilization, the expected cycle times obtained from the model without blocking is also a lower estimate. When $\lambda_r$ varies from 45 pallets/hr to 54 pallets/hr, the cycle time estimate from the model without blocking is lower by 24% and 67% respectively.

**VI. SUMMARY AND CONCLUSIONS**

This paper presents a queuing network model to study the effects of vehicle blocking in unit-load operations in warehouses with autonomous vehicles. Although, these blocking effects could have a significant impact on system performance, prior studies have ignored these effects due the complexity involved in performance analysis. In this paper, first, efficient blocking protocols are developed to model blocking effects within aisles, in the cross aisle and at the intersection of aisles and the cross-aisle. The operation of vehicles servicing storage...
and retrieval transactions are modeled in detail using a semi-open queuing network. Since exact solutions to the semi-open queuing network are not available, a decomposition approach is proposed. Decomposition yields two sub-networks: a closed queuing network corresponding to the case when vehicles wait for transactions, and an open queue corresponding to the case when transactions wait for vehicles. Each sub-network is analyzed separately to obtain conditional queue length distributions at each node of the network. The solution algorithm then links the solution of the two sub-networks to obtain the unconditional queue length distributions and other relevant performance measures at each individual node in the network. Detailed simulations are carried out to show the efficacy of the analytical model and solution approach. Comparison of results with simulation shows that performance estimates are fairly accurate.

Numerical studies indicate that blocking delays could contribute to a significant amount (10%-20%) to the transaction cycle time. This underscores the importance of detailed performance evaluation models that explicitly incorporate blocking effects. This model also provides several insights with respect to the effect of design parameters such as number of vehicles, number of storage locations, and \( \frac{W}{D} \) ratio on blocking delays and cycle times. As expected, the percentage of blocking delays increase with the number of vehicles in the system. However, the effect of blocking decreases as the vehicle utilization increases, as the effect of increase in wait times to obtain a free vehicle starts to dominate in systems with high vehicle utilization. The studies also suggest that the number of storage locations does not have a significant impact on the blocking delay percentages, however, the tier configuration \( \frac{D}{W} \) ratio does influence the extent and type of blocking effects observed in a system.

These results indicate that it is important to quantify effects of blocking while estimating performance measures of AVS/R systems. These models can be used during the design conceptualization phase of AVS/R such as analyzing the effect of varying fleet size, \( \frac{W}{D} \) ratio, and storage locations on system performance. Using this analytical model, a large number of system configurations can be analyzed rapidly and a smaller set of candidate design profiles can be generated for detailed evaluation.

Acknowledgements: This work is supported in part by the National Science Foundation under Grants CMMI-0848756 and CMMI-0946706. The authors would like to thank the anonymous reviewers and the editor for providing detailed comments, which have significantly improved the quality of this paper.

References


Debjit Roy is a Ph.D. candidate and a research assistant in the Department of Industrial and Systems Engineering at the University of Wisconsin-Madison. His research interests are in supply chain management, performance modeling of automated warehouse and manufacturing systems, material handling technologies and queuing theory.

Ananth Krishnamurthy is an associate professor in the Department of Industrial and Systems Engineering at the University of Wisconsin-Madison. He is also the director of the Center for Quick Response Manufacturing and Manufacturing Systems Engineering Program. His research focuses on production inventory systems, assembly operations, product variety and customization, material handling and warehouse systems and quick-response manufacturing. He earned his Ph.D. in industrial engineering and an M.S. in manufacturing systems engineering from the University of Wisconsin-Madison.

Suderesh S. Heragu is professor and the Mary Lee and George F. Duthie Chair in Engineering Logistics in the Department of Industrial Engineering at the University of Louisville, where he is also director of the Logistics and Distribution Institute. Heragu has held regular or visiting appointments at State University of New York, Plattsburgh; Rensselaer Polytechnic Institute; the University at Buffalo; IBM’s T.J. Watson Research Center; the Technical University of Eindhoven and the University of Twente in the Netherlands. Heragu is a fellow of IIE and received IIE’s Award for Technical Innovation in Industrial Engineering in 2008.

Charles J. Malmborg is professor and head of the Department of Industrial and Systems Engineering at Rensselaer Polytechnic Institute. He has taught courses in facility design and materials handling, operations research, production systems, work systems, industrial engineering design, engineering economics and applied statistics. He is author or co-author of more than 200 technical publications on research in material flow logistics, facilities planning, decision analysis and production systems.
<table>
<thead>
<tr>
<th>Case</th>
<th>( N_i )</th>
<th>( V )</th>
<th>( \lambda_a, \lambda_v (p/\text{hr}) )</th>
<th>Output: Performance Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>58,58</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>63,33</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>68,88</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>74,44</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>80,00</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>85,55</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>91,11</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>96,66</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>102,22</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>107,77</td>
<td>5.10</td>
<td>16.35 156.8 3.1 15.8 19.3</td>
</tr>
</tbody>
</table>

**APPENDIX A: VALIDATION RESULTS FOR A TIER WITH 7200 STORAGE LOCATIONS AND \( D_{W}=1.5 \)**