EVALUATING TRANSACTION PAIRING STRATEGIES FOR VEHICLE-BASED HIGH-DENSITY STORAGE SYSTEMS

Debjit Roy, Ph.D.

Production and Quantitative Methods Area, Indian Institute of Management
Ahmedabad, Gujarat 380015, India

Harit Joshi

Production and Quantitative Methods Area, Indian Institute of Management
Ahmedabad, Gujarat 380015, India

Abstract

Vehicle-based storage and retrieval systems present an attractive choice for distribution center automation because it provides the flexibility in managing demand fluctuations without affecting transaction throughput times. In this research, we contend that while dual-command cycles can reduce the vehicle travel times for processing transactions, it may not be the best policy for reducing transaction throughput times when transactions arrive at random time instants. We develop stochastic models to test the transaction throughput time performance with multiple pairing strategies and present operational insights.

1. Introduction

Autonomous vehicle-based storage and retrieval solutions have found applications in high-density warehouse systems such as deep-frozen and distribution warehouses, where storage efficiency, throughput flexibility, and throughput time responsiveness are the key system requirements (*www.savoye.com*). In Autonomous Vehicle-based Storage and Retrieval Systems (AVS/RS), vehicles carry pallet loads in the vertical direction using lifts and move in the horizontal tiers using rail-guided paths (see Figure 1 for an illustration of the system configuration).

Prior research in AVS/RS analyzes the effect of several system and operational design parameter settings on system performance (Roy et al., 2012; Roy et al., 2015a; Roy et al., 2015b). For instance, Roy et al., 2012 consider the effect of different vehicle assignment rules on transaction throughput times whereas Roy et al., 2015a consider the effect of alternate vehicle dwell-point policies on system throughput times. These studies assume that the transactions are processed using a single command cycle policy. However, due to rectangular tier configuration, the dual command cycle pose an attractive alternative against the single command cycle. Some of the early research studies consider opportunistic interleaving of transactions in AVS/RS where a transaction is executed in a dual command cycle only if both storage and retrieval transactions are found waiting for a vehicle. If a storage or a retrieval transaction finds a vehicle on arrival, then the transaction is executed in a single command cycle (see Malmborg, 2002; Malmborg, 2003).

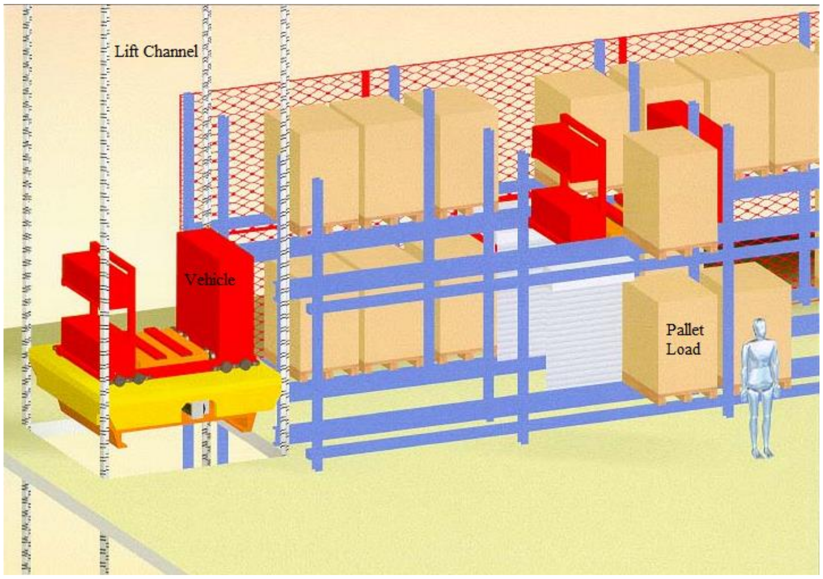


Figure 1: A section of a high-density vehicle-based storage and retrieval system

In this paper, we compare the system performance for three transaction pairing strategies with a single command cycle and opportunistic interleaving policy. In each pairing strategy, storage and retrieval transactions are paired together on the basis of transaction commonalities. For instance, in the first strategy (Same-aisle Pairing), a storage and a retrieval request is paired only if both transactions request access to a storage location in the same aisle of the tier. Likewise, in the second strategy (Same-side Pairing, FCFS), a storage and retrieval request is paired only if both transactions request access to a storage location from the same side (left or right) of the Load/Unload (LU) point. The third strategy (Same-side Pairing, closest) is similar to the Same-side Pairing where a storage and retrieval request is paired only if both transactions request access to a storage location present in the same side (left or right) of the LU point. However, the incoming transaction (storage or retrieval) chooses the matching (retrieval or storage) transaction that is nearest in its neighborhood. Rightly paired dual command cycles can reduce the transaction travel time substantially, but it is not clear if the dual command cycles also reduce the overall transaction throughput time, particularly when transactions arrive in a random fashion.

Using stochastic models, we analyze the relative performance for different system design parameters. We restrict our scope to a single tier of a high density storage system. The remainder of the paper is organized as follows. Section 2 provides a description of the system, the pairing strategies and the operational trade-offs between the throughput time components. Section 3 describes the stochastic model and performance measures. Section 4 presents the results of the numerical experiments and the operational insights. Section 5 summarizes the paper and also provides scope for future work.

2. System description

We consider a single tier with a set of vehicles and a single LU point. The tier has a set of aisles that run orthogonal to the cross-aisle (Figure 2). We consider the effect of alternate transaction pairing strategies on the transaction throughput time. In case of dual command cycles, the transaction throughput time consists of three components: waiting time to be paired with another transaction, waiting time for a vehicle, and travel time. Note that in a dual command cycle, the retrieval transaction follows the storage transaction. Hence, the throughput time expression for a retrieval transaction is always more than a storage transaction. We now describe the pairing strategies and their corresponding throughput time expressions using 2D coordinates of the vehicle dwell point, LU point, and storage locations. We also qualitatively compare the performance of single command cycle with the four pairing strategies using the throughput time components.

2.1 Single command cycle

In this case, one transaction is executed in every cycle (see Roy et.al (2012) for further details). The throughput time expressions for storage and retrieval transactions executed in a single command cycle are shown in Equations 1 and 2, respectively. The notations are described in Table 1.

$$TT(s) = W_y + + \left|\frac{X_d - X_{lu}}{v_l}\right| + \left|\frac{Y_d - Y_{lu}}{v_l}\right| + L_t + \left|\frac{X_s - X_{lu}}{v_l}\right| + \left|\frac{Y_s - Y_{lu}}{v_l}\right| + U_t + \left|\frac{Y_{lu} - Y_s}{v_l}\right| + \left|\frac{X_{lu} - X_s}{v_l}\right| \quad (1)$$

$$TT(r) = W_y + \left|\frac{X_r - X_{lu}}{v_l}\right| + \left|\frac{Y_r - Y_{lu}}{v_l}\right| + L_t + \left|\frac{Y_{lu} - Y_r}{v_l}\right| + \left|\frac{X_{lu} - X_r}{v_l}\right| + U_t \quad (2)$$

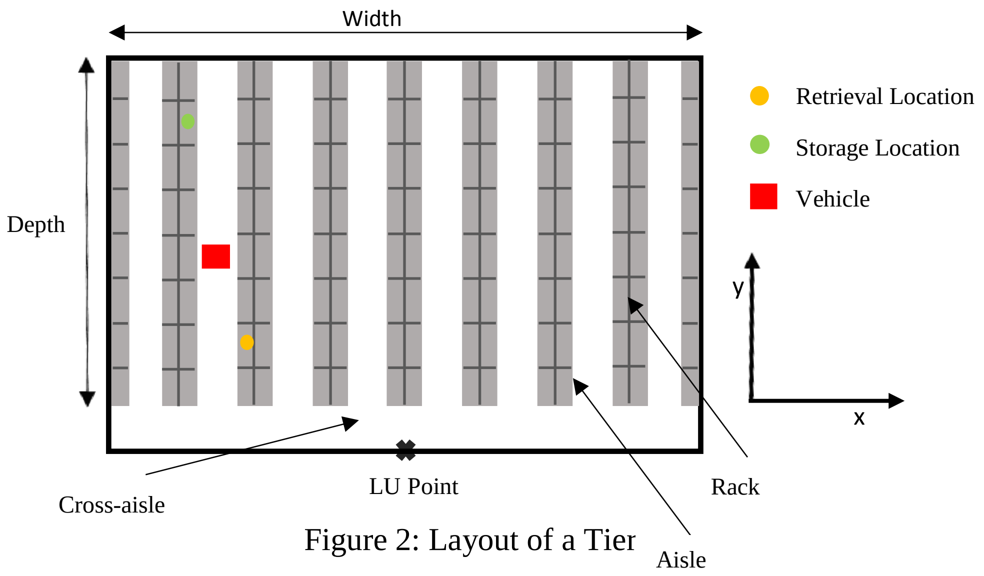


Figure 2: Layout of a Tier

Table 1: Notations for model parameters

Notation	Description
$\boldsymbol{CT(s)}$	Cycle time to complete storage transaction
$\boldsymbol{CT(r)}$	Cycle time to complete retrieval transaction
$\boldsymbol{W_y}$	Waiting time to access free vehicle
$\boldsymbol{W_p}$	Waiting time for pairing
$\boldsymbol{v_l}$	Horizontal travel velocity of the vehicle
$\boldsymbol{X_{lu}}$, $\boldsymbol{Y_{lu}}$	Coordinates of the LU or I/O point
$\boldsymbol{X_r}$, $\boldsymbol{Y_r}$	Coordinates of the retrieval point
$\boldsymbol{X_s}$, $\boldsymbol{Y_s}$	Coordinates of the storage point
$\boldsymbol{X_d}$, $\boldsymbol{Y_d}$	Coordinates of the dwelling point
$\boldsymbol{L_t}$	Loading time
$\boldsymbol{U_t}$	Unloading time

2.2 Opportunistic interleaving

In this policy, dual command cycles are used only when transactions are waiting for a vehicle. Hence, there is no exclusive wait for pairing two transactions. Hence, the transactions are executed in a mixed command cycle mode: dual and single. There are four operational scenarios in this case. They are: 1) Transaction arrives and finds vehicles busy, then the transactions wait in the buffer S or R based on its type, storage or retrieval, respectively, 2) Transaction arrives and finds free vehicles, then the transaction is assigned to a vehicle based on a vehicle assignment policy. The transaction in this case is executed in a single command cycle, 3) Vehicles arrive and find transactions waiting to be completed. If all transactions belong to either storage or retrieval type only, a single command cycle is performed, but in case both storage and retrieval transactions are waiting, then a dual cycle is implemented, 4) if vehicles arrive and find no transactions in the wait list then the vehicles idle. The throughput times for the storage and retrieval transaction types are provided in Equations 3 and 4, respectively.

$$TT(s) = W_p + W_y + \left|\frac{X_d - X_{lu}}{v_l}\right| + \left|\frac{Y_d - Y_{lu}}{v_l}\right| + L_t + \left|\frac{X_s - X_{lu}}{v_l}\right| + \left|\frac{Y_s - Y_{lu}}{v_l}\right| + U_t \quad (3)$$

$$TT(r) = TT(s) + \left|\frac{Y_{lu} - Y_s}{v_l}\right| + \left|\frac{X_s - X_r}{v_l}\right| + \left|\frac{Y_r - Y_{lu}}{v_l}\right| + L_t + \left|\frac{Y_{lu} - Y_r}{v_l}\right| + \left|\frac{X_{lu} - X_r}{v_l}\right| + U_t \quad (4)$$

If no transactions are waiting, then the throughput time expressions are similar to single command cycle. If either storage or retrieval transactions are waiting for a vehicle, then the throughput time expressions can be obtained by setting Wp to zero. Further, the expression for estimating the storage throughput time needs to be altered to account from the empty travel from the point of previous service completion (X_d, Y_d) to the LU point. Note that the travel time components: $\left|\frac{X_d - X_{lu}}{v_l}\right|$ and $\left|\frac{Y_d - Y_{lu}}{v_l}\right|$can be positive if the previous transaction was storage and executed in a single command cycle mode.

2.3 Same-aisle pairing strategy

In this case, storage and retrieval transactions that need to access the same aisle are paired together to provide maximum savings in travel time. In this pairing strategy, transactions wait in 2N separate wait lists (N for storage and N for retrieval) depending on the type of job and the transaction aisle number, as they arrive. Once the transactions are paired, they wait together for a vehicle. Once a vehicle is allocated to the paired transaction, it completes both storage and retrieval jobs in tandem, which we denote as a dual command cycle. Vehicle moves to the required aisle, performs storage operation and instead of dwelling at the point of storage transaction completion, it performs retrieval from the same aisle. This pairing leads to a substantial reduction in travel time because for both storage and retrieval transaction, two instances of cross-aisle travel are reduced. However, on the other hand waiting time may increase significantly because transaction pairing now occurs at every aisle level. The throughput times for the two transaction types are show in Equations 5 and 6.

$$TT(s) = W_p + W_y + L_t + \left|\frac{X_s - X_{lu}}{v_l}\right| + \left|\frac{Y_s - Y_{lu}}{v_l}\right| + U_t \tag{5}$$

$$TT(r) = TT(s) + \left|\frac{Y_s - Y_r}{v_l}\right| + L_t + \left|\frac{Y_r - Y_{lu}}{v_l}\right| + \left|\frac{X_r - X_{lu}}{v_l}\right| + U_t \tag{6}$$

2.4 Same-side pairing

We define two sides of a tier: the left side and the right side. The left side corresponds to the aisles and storage locations that lie to the left of the aisle along the LU point. Likewise, the right side corresponds to the aisles and storage locations that lie to the right of the aisle along the LU point. In this case, the storage and retrieval transactions are paired such that both storage and retrieval locations lie either on the left or on the right of the aisle passing along the LU point. A storage (or a retrieval) transaction waits in a queue until a retrieval (or storage) transaction with locations in the same side of the tier arrives. The transactions are served on a first come first serve (FCFS) basis. We expect the waiting time for pairing transactions to be lower than the same-aisle strategy because now the pairing is done for any aisle on the same side of LU point. However, we expect the travel time in this strategy to be higher than same-aisle pairing. The throughput time expressions are illustrated in Equations 7-8. Note that the retrieval transaction has two cases depending on the relative position of the storage and retrieval location.

$$TT(s) = W_p + W_y + L_t + \left|\frac{X_s - X_{lu}}{v_l}\right| + \left|\frac{Y_s - Y_{lu}}{v_l}\right| + U_t \tag{7}$$

Case 1: the retrieval aisle is same as the storage aisle

$$TT(s) = W_p + W_y + L_t + \left|\frac{X_s - X_{lu}}{v_l}\right| + \left|\frac{Y_s - Y_{lu}}{v_l}\right| + U_t \tag{8}$$

Case 2: the retrieval aisle differs from the storage aisle

$$TT(r) = TT(s) + \left|\frac{Y_{lu} - Y_s}{v_l}\right| + \left|\frac{X_s - X_r}{v_l}\right| + \left|\frac{Y_r - Y_{lu}}{v_l}\right| + L_t + \left|\frac{Y_{lu} - Y_r}{v_l}\right| + \left|\frac{X_{lu} - X_r}{v_l}\right| + U_t \tag{9}$$

2.5 Same-side pairing strategy (closest neighbor)

In this strategy as well, the transactions on the same side of LU point are paired. However, preference is given to the closest neighbor among the transactions waiting in the pairing queue. The summary of the tradeoffs among the components of the throughput time for the five strategies are included in Table 2.

Table 2: Qualitative comparison of pairing strategies with single command cycle

Strategy	Waiting time for pairing	Waiting time for vehicle	Waiting time for vehicle
1. Single Cycle	0	High	High
2. Opportunistic Interleaving	0	Lower than Strategy 1	Lower than Strategy 1
3. Aisle only pairing	Highest	Lowest	Lowest
4. Pairing for same side (FCFS)	High	Higher than Strategy 3	Higher than Strategy 3
5. Pairing for same side (Closest neighbor)	High	Higher than Strategy 3	Higher than Strategy 3

3. Model assumptions and description

A Poisson arrival process is assumed for both storage as well as retrieval jobs with rates λs and λr, respectively. Except the same side (closest neighbor) pairing strategy, the scheduling policy for both storage and retrieval transactions follows an FCFS discipline. Vehicles travel at a constant speed in the aisles and the cross-aisle, and the effect of acceleration/deceleration is ignored. Storage and retrieval locations are uniformly distributed in the aisle locations. The vehicles dwell at the same point after completion of the service. The tier configuration is such that the LU point is placed at the center of cross-aisle with equal number of aisles on both sides making the total number of aisles even.

In Figure 3, we sketch the semi-open queuing network models for a single command cycle policy and different transaction pairing strategies. The model for a single command cycle policy is provided in Roy et al. (2012). In each model, we clearly highlight the waiting and the travel time components. For example, in Figure 3c, the transactions wait at two queues. First the transaction waits in the left or the right side buffer to get paired and then waits for a free vehicle to be available. The final component is the transaction travel and pallet pick-up and drop-off times.

Note that the networks represented in Figures 3(b)-3(d) are not work-conserving because the transactions may be waiting in the pairing buffers and vehicles may be waiting simultaneously at the wait for transaction buffer. To evaluate the semi open queuing networks, we would need to maintain a record of possible storage and retrieval locations. While the queuing network models can be evaluated using continuous time Markov chains, non-work-conserving queues can make the network quickly unstable. We evaluate the performance measures using discrete-event simulation model developed using AutoMod^{TM} software. The model represents a single tier of the storage system. The storage system is represented using the path mover system, through which travel paths, locations, loads and vehicles can be modeled. The paths comprise of a cross–aisle path and N aisle paths where N is the number of aisles considered. The key locations to be modelled include the storage locations modelled on each aisle and the LU point.

Alternate pairing strategies are modelled through separate logic functions. The modelling logic controls the choice of pairing strategy. For example, in case of opportunistic inter-leaving, transaction information is maintained in a separate order list for both storage as well as retrieval, but pairing is done only if the idle vehicle order list is empty. Similarly, pairing based on different pairing strategies is achieved based on comparing entries of the storage and retrieval order lists. Once a paired or single transaction is allocated a vehicle, the simulator controls the movement of vehicle to the desired storage location and then proceeds to the retrieval location.

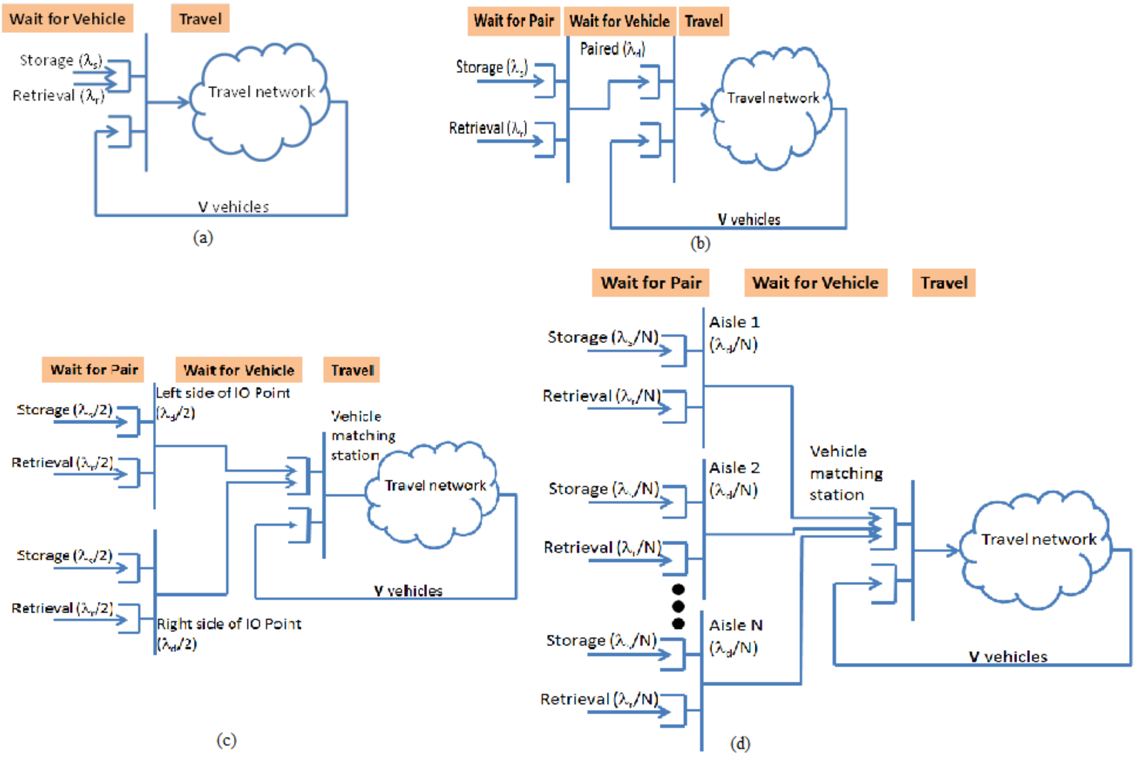


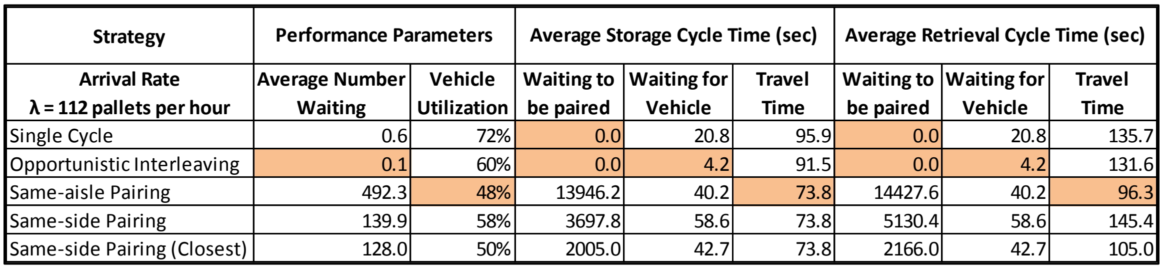
Figure 3: Semi-open queuing network models for (a) single command cycle policy, (b) dual command cycle policy (c) same-side pairing strategy (d) same-aisle pairing strategy

4. Numerical experiments and performance measures

The throughput time of a transaction cycle which includes the waiting time for pairing (in case of dual strategies), waiting time for a vehicle, and travel time, is an important performance measure. Each pairing strategy represents a trade-off among the components of the throughput time while the overall throughput time helps us to identify the pairing strategy that performs better in the long run. The relative weightage of each of these time duration (waiting and travel time) is in turn dependent on design parameters of storage system such as Depth/Width (D/W) ratio of the storage space, cross-aisle travel distance, vehicle travel speed, and number of vehicles in the system. Each strategy is evaluated using a common set of parameters including transaction arrival rates.

For comparing the performance of different dual command cycle pairing strategies and single command cycle, a tier configuration with D/W ratio of 0.5 is considered with 5 vehicles. The tier has 20 aisles, 10 on each side of LU point with each aisle, and 102 bay locations on each side of an aisle. In sum, the tier has 2040 storage locations. The arrival rates vary between 112 pallets per hour to 191 pallets per hour in increment of 5 pallets per hour. We consider 15 replications for each scenario with a 5 days warm-up period, 20 days run length, and 95% CI.

Table 3: Numerical results



Strategy	Performance Parameters		Average Storage Cycle Time (sec)			Average Retrieval Cycle Time (sec)		
Arrival Rate λ = 112 pallets per hour	Average Number Waiting	Vehicle Utilization	Waiting to be paired	Waiting for Vehicle	Travel Time	Waiting to be paired	Waiting for Vehicle	Travel Time
Single Cycle	0.6	72%	0.0	20.8	95.9	0.0	20.8	135.7
Opportunistic Interleaving	0.1	60%	0.0	4.2	91.5	0.0	4.2	131.6
Same-aisle Pairing	492.3	48%	13946.2	40.2	73.8	14427.6	40.2	96.3
Same-side Pairing	139.9	58%	3697.8	58.6	73.8	5130.4	58.6	145.4
Same-side Pairing (Closest)	128.0	50%	2005.0	42.7	73.8	2166.0	42.7	105.0

Table 3 present values of performance measures for an arrival rate, λ = 112 pallets per hour. First, we can observe that retrieval cycle time for dual command cycles are more than that of storage cycle time because the storage process steps are a subset of the retrieval process steps. The total time expressions for various pairing strategies in section 2 clearly indicate this difference. We also observe that for retrieval transactions, same aisle pairing has lowest travel time followed by same side pairing strategy with closest neighbor. Also, same-aisle pairing results in lowest vehicle utilization of 48%. However, we note that the number of transactions that wait for pairing increases significantly in comparison to the benefit obtained with travel time reduction. Hence, the waiting time for pairing is a significant component in the various pairing strategies. However, the waiting time for pairing reduces with increase in arrival rates (See Figure 4).

Opportunistic interleaving strategy provides a benefit over single command cycle because some of the transactions get paired resulting in overall lower throughput time.

Further, opportunistic interleaving reduces waiting time for a vehicle because of lower vehicle utilization in comparison to a single command cycle. In our numerical experiments, we also observed that the benefit of opportunistic interleaving over single command cycle further increases as arrival rate increases with many more transactions getting an opportunity to pair. In sum, opportunistic interleaving strategy is a better operational policy in comparison to other transaction pairing strategies, particularly when transactions arrive in a random fashion.

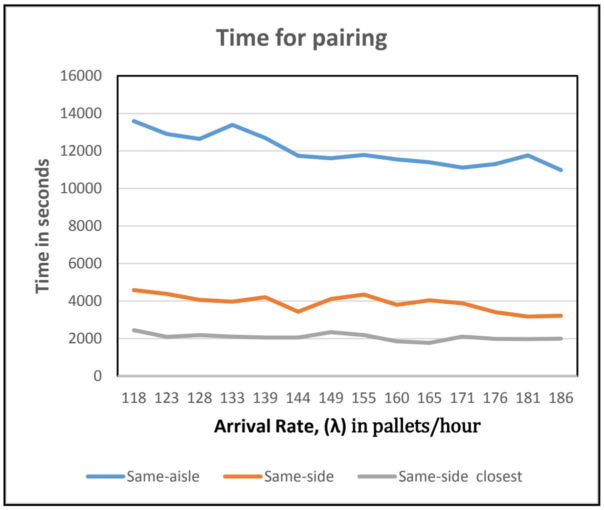


Figure 4: Comparison of average pairing times for varying arrival rates

5. Conclusions and future extensions

In this paper, we evaluate if there is a merit in adopting alternate transaction pairing strategies to reduce the throughput time if transactions arrive at random. Towards this pursuit, we evaluate four alternate pairing strategies that promise a reduction in the vehicle travel times. It is observed that opportunistic interleaving policy provide time savings over single command cycle while other pairing strategies, though result in reduction in travel time, introduce a substantial waiting time for pairing. Such transaction waiting times make the strategy an inefficient choice. In this research, we consider stochastic arrivals for storage and retrieval jobs. In practice, if the transactions are released in waves, both storage and retrieval jobs can be paired better by considering all transactions in a wave with better sequencing. This area can be a potential subject of future research.

References

[1] C.J. Malmborg, Conceptualizing tools for autonomous vehicle storage and retrieval systems. International Journal of Production Research, 40(8), 1807–1822 (2002).

[2] C.J. Malmborg, Interleaving dynamics in autonomous vehicle storage and retrieval systems. International Journal of Production Research, 41(5), 1057–1069 (2003).

[3] D. Roy, A. Krishnamurthy, S.S. Heragu and C.J. Malmborg. Performance analysis and design tradeoffs in warehouses with autonomous vehicle technology, IIE Transactions, 44(12): 1045-1060 (2012).

[4] D. Roy, A. Krishnamurthy, S.S Heragu, C. Malmborg, Queuing models to analyze dwell-point and cross-aisle location in autonomous vehicle-based warehouse systems, European Journal of Operational Research, 242(1), 72-87 (2015a).

[5] D. Roy, A. Krishnamurthy, S.S Heragu, C. Malmborg, Stochastic models for unit-load operations in warehouse systems with autonomous vehicles, Annals of Operational Research, 231(1), 129-155 (2015b).

[6] D. Roy, A. Krishnamurthy, S.S Heragu, C. Malmborg, Blocking Effects on Performance of Warehouse Systems with Autonomous Vehicles. Progress in Material Handling Research, 227-236,(2010)

[7] AutoMod Version 12.6, 2015, AGS Automation Products, Applied Materials
< http://www.appliedmaterials.com/global-services/automation-software/automod>