Deadlock and Collision Prevention Algorithms for Mobile Robots in Automated Warehouses

Ecem Eroglu Turhanlar and Banu Yetkin Ekren
Department of Industrial Engineering
Yasar University
Izmir, Turkey
ecem.eroglu@outlook.com, banu.ekren@yasar.edu.tr

Abstract

Recent supply chain targets have led to decreased response time requests also causing enterprises invest for automation technologies for efficient management for their warehouses. In this paper, we focus on an automated warehousing environment where there are robotic order pickers travelling on ground flexibly throughout aisles. We develop deadlock and collision prevention algorithms for safely travel of those robotic order pickers. The developed algorithms can be utilized by any warehouse system having autonomous vehicles (e.g., automated guided vehicles, mobile robots, or shuttles in a multi-tier automated warehouse) that can travel between aisles freely. While developing those smart algorithms, we also aim to improve some performance metrics such as average flow time and, maximum flow time of a transaction in the system. To test the performance of the algorithms, we model the system as a multi-agent system, where autonomous vehicles are able to communicate with each other under an Internet of Things (IoT) environment. We compare that flexible system design with a dedicated design where a dedicated vehicle is assigned for a pre-defined number of aisles. The experimental results show that under well-designed deadlock and collision prevention control algorithm, flexible travel of vehicles can provide advantage compared to dedicated design.

Keywords
Agent-based simulation, Deadlock prevention, Autonomous vehicle and Mobile robots.

1. Introduction
By the rapid growth of e-commerce that is also a result of COVID-19 outbreaks, an unprecedented rise in the number of online shoppers has revealed. Intralogistics sector is facing new challenges by that effect. For instance, recent e-commerce increase has also shifted order profile towards increased product variability, decreased delivery time and flexible delivery options. To overcome those challenges, investing on warehouse automation technologies providing high transaction rate, decreased errors, under those complicated demand profile requests could be an option. For instance, recently enterprises are very eager to deploy goods with RFID tags helping efficiently implementation of robotic material handling technologies. The growth estimation for the Global Warehouse Automation Market is at a CAGR of 14% between 2020 and 2026 which is estimated to be doubled to $30 billion by 2026 (Research and Markets, 2021).

Increase in e-commerce, has also caused the material handling equipment market growth. For instance, the Material Handling Market is projected to grow at a CAGR of 6.01% to reach US$ 201.057 billion by 2025, from US $141.657 billion in 2019 (Knowledge Sourcing Intelligence, 2020). Fig. 1 shows the current and estimated material handling market growth based on product types.

Based on Fig. 1, automated storage and retrieval systems (AS/RSs) are predicted to witness the fastest growth during the forecast period. The AS/RS technology providers aim to develop system designs providing reduced floor space, labor cost, and inventory levels to enhance productivity while reducing manufacturing cycle time. For instance, a mini-load AS/RS technology that is market by Dematic Group, provides ultra-high-speed load handling referred as shuttle-based storage and retrieval system (SBS/RS) mostly utilized in distribution centers and raw material stores that are specially designed for ultra-high-speed load handling (Dematic Multi-shuttle, 2021). Because there is a dedicated shuttle in each tier of an aisle, this automated warehousing technology is referred as aisle and tier-captive SBS/RS in literature. Note that, the collision and deadlock prevention algorithms provided by this paper can also be utilized by

© IEOM Society International

SBS/RSs where shuttles are allowed to travel between aisles within a tier. This SBS/RS design can be referred as aisle-to-aisle SBS/RS.

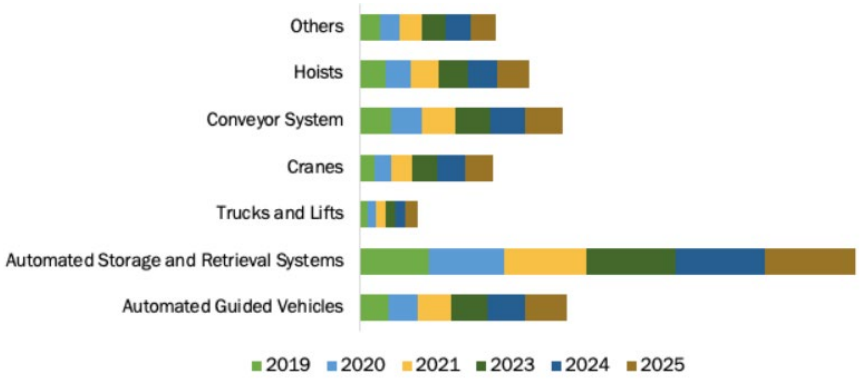

Figure 1. Material handling market growth forecast by product, 2019 to 2025 (source: Knowledge Sourcing Intelligence, 2020)

Mobile robots travelling on ground like automated guided vehicles (AGVs) and automated mobile robots (AMRs) have also huge growth expectation, at a CAGR of 35% by 2026 (Research and Markets, 2020). It is expected that those mobile robots will reach more than 18% of overall warehouse automation market share by 2026. (Research and Markets, 2021).

Our main motivation to the present paper is due to insufficient works on smart collision and deadlock prevention algorithms in literature that can be utilized for warehouses utilizing AMRs and AGVs. Allowing autonomous vehicles flexible travel pattern might also result with decreased number of autonomous vehicles compared to a dedicated system. Decreased number of vehicles would result with decreased investment cost of the system. In practice, in an effort to prevent collisions of AMRs, companies are widely implementing either a fixed travel route (e.g., single direction travel) or zoning (e.g., dedicated area) for those vehicles. In the following section, we provide the related studied works on the subject.

2. Literature Review

Policies to prevent collision and deadlock are mainly studied for AGV systems rather than AS/RSs (e.g., SBS/RSs). An AGV design called EX-AGV allowing load exchange between AGVs is developed by Hsueh (2010). The results of the simulation study show the efficiency and robustness of the proposed system design in terms of the considered performance metrics. A conservative path reservation policy for the collision avoidance between AGVs is presented by Cossentino et al. (2011). Their policy is applied on an agent-based simulation model where they offer it for decision-making in a warehouse. Roy et al. (2013) develop vehicle blocking prevention protocols for three types of blocking case for the warehouses with autonomous vehicles (AVs). In their numerical studies, it is concluded that delays caused by blocking give rise to transaction cycle time output roughly 10-20%.

As mentioned an SBS/RS with aisle-changing shuttle carriers may resemble the studied work. One is studied by Lerher (2018) where analytical travel time models and comparison of those models with their simulation results are presented. However, there is no deadlock and collision prevention algorithm in his models. Lienert and Fottner (2017) present a model utilizing time window routing method for safe shuttle movement in a tier-to-tier and aisle-to-aisle system configuration. Rhazzaf and Masrour (2021) divide the high dimensional warehouses into low dimensional zones and achieve good speed performance. There exist some other works on autonomous vehicle-based storage and retrieval systems (AVS/RSs) including autonomous vehicle travel between aisles (Ekren 2011; Heragu et al. 2011; Ekren and Heragu 2012; Ekren et al. 2014; Roy et al. 2015). And, there are some other works on AVS/RSs including autonomous vehicle travel between tiers (Ekren 2020a, Ekren 2020b). To the best of our knowledge, those works do not provide any algorithm considering preventing of deadlocks. A collision solution approach for horizontal vehicle movement is presented on works include vehicle travel between tiers which is allowing a single shuttle to travel within a tier (Küçükyaşar et al. 2020; Jerman et al. 2021).

The literature includes few papers, examining collision and deadlock avoidance of AVs changing aisles. Instead of flexible travel pattern, the existing works mostly consider dedicated designs where autonomous vehicles cannot travel flexibly. Hence, in those works there is no need for algorithms considering collision and deadlock preventions. In this paper, we propose flexible travel patterns for AVs, where we assume that those vehicles are intelligent agents that are able to communicate with each other to make smart decisions on transaction picking from their queues and their route paths. We apply a multi-agent simulation approach to seek those well-designed control policies.

3. Flexible and Dedicated System Designs

The system proposed in this work is referred as Flexible System Design (FSD) consisting of AMRs travelling flexibly on ground within aisles compared to its alternative design Dedicated System Design (DSD). Figure 2 shows the physical configuration of FSD and DSD.

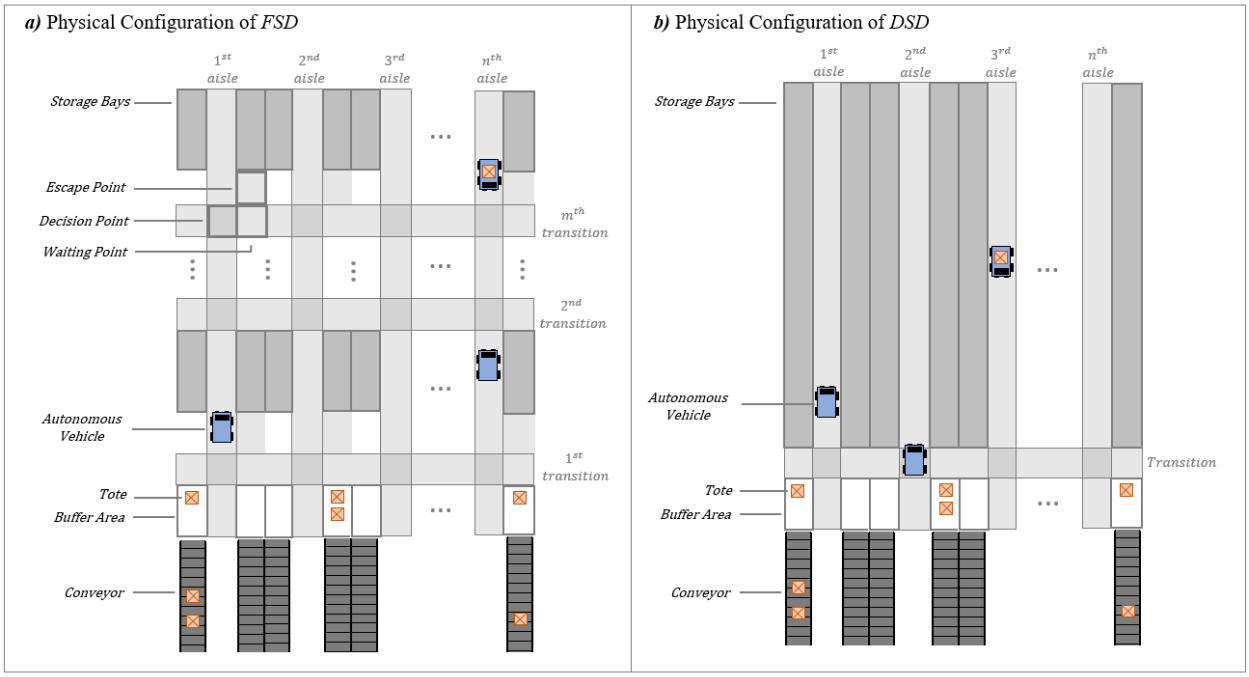

Figure 2. Physical configurations of the studied system designs

In FSD, AMRs can travel on transition roads connecting aisles. Transitions between aisles can be done through roads called "transition" in Figure 2. There is a "decision point" at each intersection point of aisle and transition road. The AMR may stop at those points to make decision on where to direct through its destination address without causing any deadlock and collision. To consider a large warehouse in practice, we can create multiple copies of the pre-designed FSD. In Figure 2, there are "waiting" and "escape" points helping for prevention of AMRs from colliding. The role of those points and the regarding operating principles are explained in Section 5. Basically, the operation of the systems is as follows: FSD includes a large number of transitions so that vehicles can transit between aisles freely. In DSD, an AMR is dedicated to a certain number of aisles so that it cannot transit to the other aisles than those dedicated ones. Hence, in that design, there is only a single transition between those dedicated aisles located at the end (i.e., buffer area) edge of each aisle. Here, our aim is to compare the performance of those designs under the developed control algorithms.

4. Agent-based Simulation Model

To model the proposed FSD, a multi-agent simulation modeling approach is found to be advantageous and appropriate, due to the complexity of the system and desired real-time information tracking purpose from the environment. We treat AMRs as intelligent agents, so that they become dynamic objects in the system which can sense their environment and make autonomous decisions. Except the AMR agents, we also treat the transactions arriving at the system as agents. Those multi-agents have the ability of communicating with each other and tracking real-time information from

their environments. AMR agents act as decision makers in the system to decide on transaction selection waiting in their queue and travel route to their destination storage addresses.

Figure 3 shows the interactions of agents. All agents can provide real-time system status information from their environment by using bidirectional communication. Later, the AMR agents act based on an evaluation procedure of those pieces of information. Here, the evaluation procedure or the decision-making strategy is referred as bidding strategy, which is the basis of agent-based modelling. The right side of the Figure 3 shows the communication between AMR agents which are in a case of deadlock possibility. In such a case, active AMR agent which is to take a decision might send a signal (Trigger Command - TC) to another AMR agent that might cause a deadlock or continue on its way (Movement Permit - MP). Then, the AMR that is to cause a deadlock sends MP signal to the active AMR. Thus, active AMR is an AV triggering the other AMR having the potential of causing deadlock in the system. That triggered AMR is called deadlock AMR, here. The movement of a triggered AMR always starts after a signal transmission, by an active AMR. In Section 5, we give an example from that procedure.

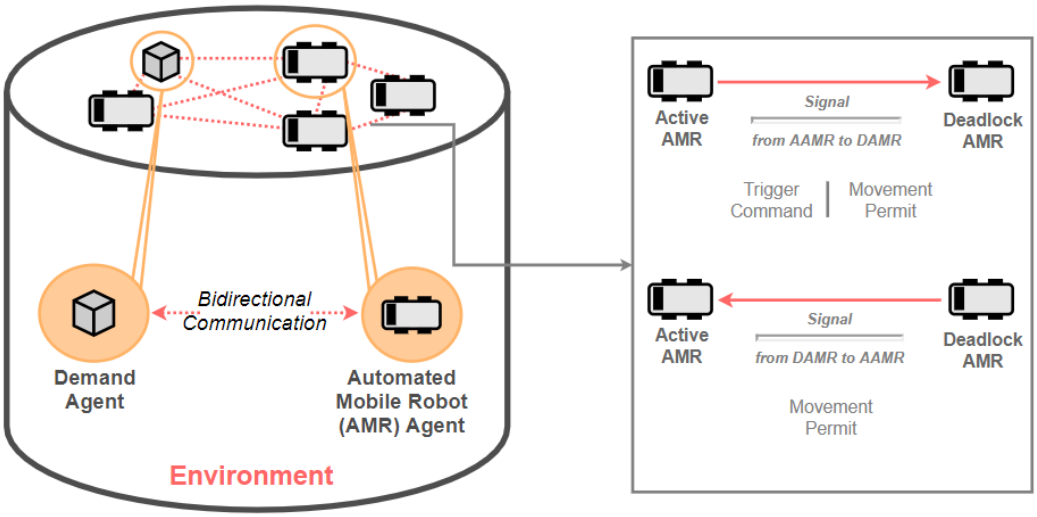

Figure 3. Communication and interaction between agents

The utilized notations for system modelling are given in Table 1.

Table 1. The utilized notations

Notation	Description	Unit
V	The maximum speed that an AMR can reach in long travel distance	m/s
a	Acceleration/deceleration value of AMR	m/s^2
λ	Mean arrival rate of transactions	transactions/month
FT	Average flow time of transactions	s/transaction
MFT	Maximum flow time of transactions	sec.
U	Average utilization of AMRs	%
SD	Standard deviation of flow time of transactions	sec.
WC	Warehouse capacity in terms of number of bays	
NV	Number of AMRs	

The assumptions that are considered in the simulation model are summarized as follows:

- Mean arrival rate (λ) for storage and retrieval transactions follows Poisson distribution with equal rate (Roy et al. 2013, Marchet et al. 2013, Ning et al. 2016, Ha and Chae 2019, Eder 2019, Wu et al. 2020). Here, the mean values are adjusted such that we obtain 95% average utilizations for AMRs.
- Storage or retrieval requests are created randomly for their bay addresses.

- The required time for tote loading and unloading onto/from the AMR is assumed to be three seconds. (Ha and Chae 2019).
- The maximum velocity (V) is assumed to be 2 m/s - 3 m/s based on the pre-defined experiments (Lerher 2018).
- The acceleration and deceleration values for velocities (a) are same and equal to 2 m/s^2 or 3 m/s^2 based on the considered experiment.
- The distance between all adjacent storage bays and points (i.e. buffers, decision, waiting, escape points) is assumed to be 0.5 m (Ning et al. 2016, Eder 2019, Ha and Chae 2019).
- The simulation run length is one month with one-week warm-up period decided by the eye-ball technique.
- The model is run for five independent replications.
- AMRs do not breakdown during the simulation run.

Verification and validation of the simulation model are done by debugging the model by animating the system as well as by the help of an expert working on design and analyses of those systems practically.

5. Intelligent Operating Procedures of FSD and DSD

Remember that in this study, the main aim is to develop smart collision and deadlock prevention algorithms for intelligent AMR agents so that they can travel safely and efficiently in the system. Also, our aim is to let those agents make smart transaction selection decisions taking into consideration both performance metrics improvements as well as safe travels. Here, the AMRs have the goal of minimizing the average flow travel time per transaction (*FT*) as well as minimizing average maximum flow time of a transaction (*MFT*) in long run. By the decreased *MFT*, we aim to contribute on the recent customer oriented supply chain targets, where companies tend to set maximum response time target for their order delivery. Here, *flow time* is measured by the time between when a transaction request is created in the system until it is disposed. Hence, this performance metric also includes waiting times in the system. Besides the transaction selection, an AMR can also make decisions on its travel route through the destination address. Remember that the AMR stops at "decision points" shown in Figure 2, while selecting the correct route through its destination address. Specifically, the AMR stops at a "decision point" to decide which next "decision point" to move, until it reaches the last "decision point" to enter the correct aisle of the destination address.

To remind, before AMR starts its travel, first it selects a transaction from its queue to process. Once an AMR becomes available, first it checks all the waiting transactions' attributes with the current waiting times that are more than the current average waiting time of all transactions. By that we aim to minimize the maximum flow time of a transaction. The checked attributes are: transaction types, storage addresses, etc. Among those long waiting transactions, the one whose address path has the least traffic density and has the closest distance to the available AMR's current location is given priority by the AMR.

Figure 4 shows the algorithmic flow for travel of an AMR agent. According to Figure 4, collision and deadlock problems are altered by triggering policies, shown in the algorithm. These are: Trigger to Waiting Point Policy, Trigger to Escape Point Policy and, Trigger AMR Movement Policy. Note that, those policies are developed for fluent travel of AMRs without any deadlock and collision cases.

We also define an extra rule to prevent an infinite blocking loop of AMRs which may happen when there are four or more AMRs in the system. The existence of that case is followed by AMRs in real-time. If this case happens, then an AMR changes its direction to break that blocking cycle and it continues its way when that blocking loop is broken.

We examine the triggering algorithm through two example cases as shown in Figure 5. In the first case, the orange vehicle is the active AMR and the yellow vehicle is the deadlock AMR. Both vehicles are at the decision points. However, the active vehicle has arrived at the decision point earlier and waited for the collision vehicle to arrive at the decision point. Since the vehicles will direct through the same directions, the orange vehicle sends a "trigger command" to the yellow vehicle to travel to the escape point. The yellow vehicle travels to the escape point. Then, it sends a "movement permit" to the orange vehicle to move. When the orange vehicle in motion decides that the yellow vehicle can continue on its way, then, it sends "movement permit" for the yellow vehicle to leave the escape point. Thus, collision is prevented. In the second case, while the orange vehicle (i.e., active AMR) is to exit the buffer area, the yellow one tends to enter the same buffer area. Here, the buffer area is the point where storage transactions arrive and the retrieval transactions are dropped off by the AMRs. Also, the white vehicle is waiting in idle status. Here,

first, because that point is not on the way of the orange vehicle, the orange vehicle sends a signal to the yellow vehicle to travel to the right-hand side "decision point" (i.e., "trigger command"). Since there is an idle vehicle waiting at the "waiting point" (i.e., the white vehicle), then a signal is sent to that vehicle by the yellow one, to let it travel to the "escape point" (i.e., "trigger command"). Consequently, the orange vehicle follows its route and the yellow vehicle travels to the buffer location, safely.

Algorithm 1 The algorithm for travel decisions of AMR agents

```
1: while an AMR arrives at a decision point do
2:     if the AMR triggered an idle AMR to the waiting point then
3:         the triggered AMR waits at the waiting point until it selects a new job to process
4:     else
5:         if the destination bay address aisle is connected that decision point then
6:             if there is currently any other AMR at the bay address aisle or currently travelling that is
               deadlocking then
7:                 the active AMR waits for the deadlock AMR to arrive at a decision point
8:                 when this condition holds, return to 5
9:             else
10:                the active AMR travels to the destination bay address
11:            end if
12:        else
13:            if there is occupancy on the road of the AMR which is to the next decision point then
14:                if the AMR in the horizontal direction through the destination bay address aisle then
15:                    if there is an available decision point in the other direction to reach the destination
                       then
16:                        that alternative direction is assigned to the AMR
17:                    else
18:                        Trigger to Escape Point Policy is applied
                           the AMR triggers the idle waiting AMR at the decision point through its route to
                           move to the escape point according to the Trigger AMR Movement Policy
19:                    end if
20:                end if
21:            else
22:                if the following horizontal decision point is required to be reached and it is empty then
23:                    the AMR travels to that second decision point without stopping
24:                end if
25:            end if
26:            if there is not occupancy on road through decision point then
27:                the AMR travels to the destination decision point
28:            else
29:                waits for the deadlock AMR to arrive at decision point
30:                when this condition holds, continue
31:            end if
32:            when the deadlock AMR arrives at the decision point, Trigger to Waiting Point Policy
               is applied
               when the triggered AMR arrives at the waiting point, it moves according to the Trigger
               AMR Movement Policy
33:        end if
34:    end if
35: end while
```

Figure 4. Algorithm for travel decisions of AMR agents through destination addresses

Remember that in DSD, a single AMR is dedicated to pre-defined numbers of aisles so that no collision or deadlock can take place. In this system design, there is a transition road across of buffer areas to enable the AMRs to change their aisles. The AMR selects a waiting transaction to process according to the closest address to its current location. Note that, the dwell point of AMRs is assumed to be the last stopping point of those vehicles.

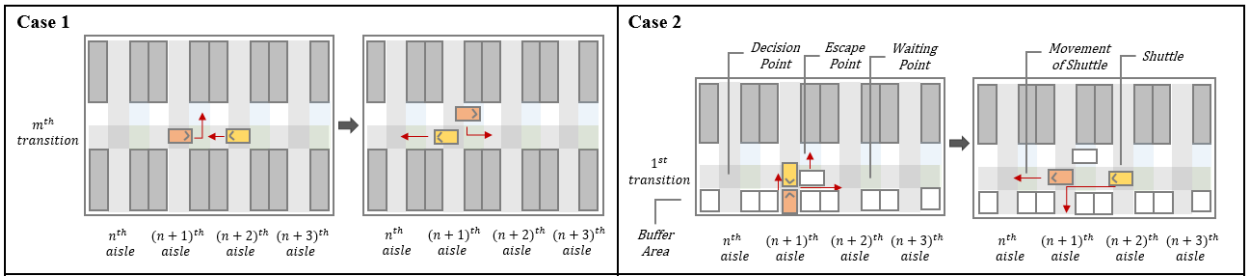

Figure 5. Example deadlock cases

6. Experimentation and Results

In this work, we aim to compare the performance of two systems designs, FSD and DSD. To make a fair comparison, we keep the arrival rate (λ) same for the two systems designs. However, while defining the mean arrival rate, we set it so that we obtain 95% utilization level in a design. Some preliminary determinations have been made in the layout configuration. According to that we assume that the warehouse capacity is 3,600 bays with 12 aisles. In every 16-17 bays, we assume that there is a transition road for transiting to other aisles. The experimentation is done by changing the number of AMRs and velocity profiles of them. Table 2 shows the experimental table with their simulation results. The results are provided under 95% confidence intervals.

Table 2. Simulation results

Design	**Input Variables**				**Performance Metrics**				
#	***System***	***V***	***a***	***NV***	***λ***	***U***	***FT***	***MFT***	***SD***
1	*FSD*	2	2	3	159,150± 471	95.0± 7×10^{-2}	112.0± 1.3	1,636± 52	104.5± 3.0
2	*FSD*	2	2	4	204,260± 317	94.2± 1×10^{-1}	92.5± 0.4	1,189± 96	74.2± 1.0
3	*FSD*	3	3	3	207,320± 372	95.0± 7×10^{-2}	84.6± 0.3	1,318± 84	79.3± 1.0
4	*FSD*	3	3	4	272,350± 441	94.6± 1×10^{-1}	71.7± 0.3	972± 78	58.9± 0.9
5	*DSD*	2	2	3	159,066± 471	83.9± 4×10^{-1}	120.8± 2.0	3,346± 502	139.6± 6.3
6	*DSD*	2	2	4	203,996± 350	82.9± 4×10^{-1}	118.0± 2.0	3,082± 657	130.5± 5.2
7	*DSD*	3	3	3	207,277± 403	81.6± 4×10^{-1}	82.9± 1.2	2,289± 301	91.9± 3.0
8	*DSD*	3	3	4	272,560± 351	81.2± 3×10^{-1}	83.2± 1.0	2,271± 461	89.4± 2.0

Note that in DSD, since there are twelve numbers of aisles in the system, the dedicated numbers of aisles for each AMR is defined by dividing twelve by the number of total AMRs in the system. For instance, when there are three numbers of AMRs in the system, a single AMR will be dedicated for four (12/3) numbers of aisles. In Table 2, we observe the performance of the system in terms four performance metrics: λ, *FT*, *MFT*, *SD*. Figure 6 shows the same results by graphs.

According to Table 2 and Figure 6 results, FSD generally outperforms DSD. Remember that arrival rates are same in both FSD and DSD designs. Therefore, that graph both results overlap.

FSD is generally found to be more efficient than DSD due to its potential of not only reducing *FT* but also *MFT* and *SD*. FSD provides an average of 53% and a maximum of 60% improvement on *FT*. Also, it provides an average of 15% and a maximum 22% improvement on *MFT*. Considering the λ, it is noteworthy that an increase in *V* and *a* values from 2 to 3 can yield the same output rate with less number of AMRs. Those increases in *V* and *a* give slightly better *FT* values but slightly worse *MFT* values.

© IEOM Society International

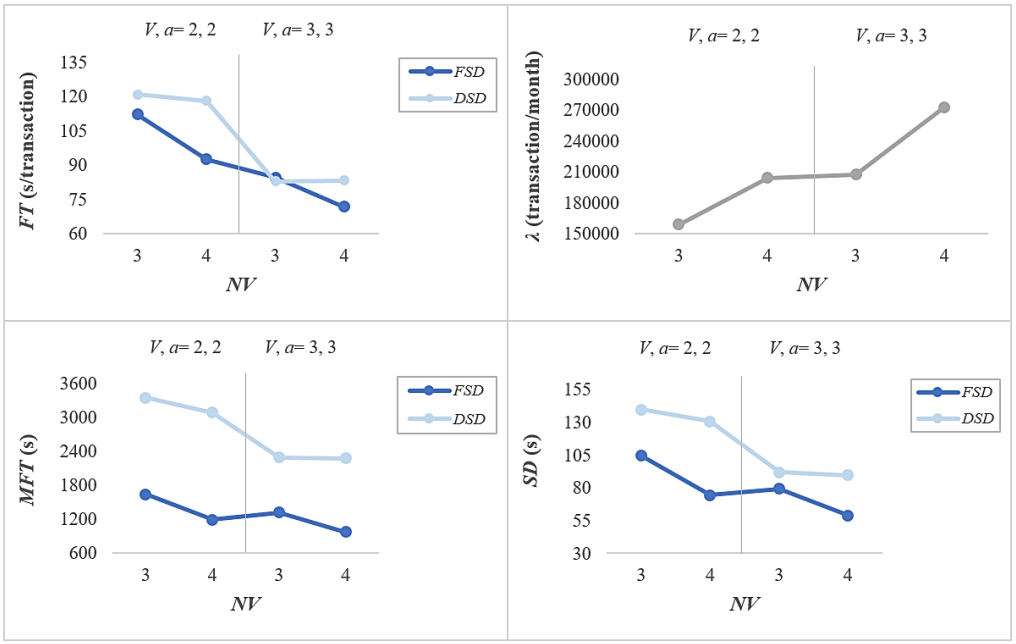

Figure 6. Graph representation of the experimental results

Verification and validation of the models are done by observing consistent performance metric values for models, FSD and DSD as well as by debugging and tracing the models by animating, Also, degenerate tests show the correctness of the developed model. For instance, when the arrival rate of the transactions is increased in the system, U, λ, MFT, and SD values tend to increase. And, an increase in V and a cause a decrease in FT value. As a result, the experiments are found to be suitable for interpretation. Additionally, low variability between simulation replications means that the model is internally valid.

7. Conclusion

In this paper, we focus on an automated warehousing environment where there are robotic order pickers travelling on ground flexibly throughout aisles. We develop flexible travel algorithms for those robotic order pickers throughout the storage racks so that they do not collide with each other. The developed algorithms can be utilized for any warehouse system having autonomous vehicles (i.e., automated guided vehicles, mobile robots, or shuttles in a multi-tier automated warehouse) that can travel between rack aisles freely in a facility. While developing those smart algorithms, we aim to improve some critical performance metrics: average flow time of a transaction, maximum flow time of a transaction as well as variance of travel times in the system. To test the performance of the algorithms, we model the system by a multi-agent simulation model, where autonomous vehicles are able to communicate with each other under an IoT environment. We compare those flexible travel system designs with their dedicated system designs where in the dedicated system design, a dedicated AMR is assigned for a pre-defined number of aisles. The experimental results show under a well-designed deadlock and collision prevention control algorithm the flexible travel of vehicles can provide advantageous compared to dedicated design. And, it is also observed that up to 22% improvement in average flow time value and 60% in maximum flow time value can be achieved by this design.

As future works, more racking designs as well as smart control algorithms can be considered to observe better performance results from the system.

References

Cossentino, M., Lodato, C., Lopes, S., & Ribino, P., Multi agent simulation for decision making in warehouse management, *Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 611-618, 2011.

Dematic Multi-shuttle, Available: https://www.dematic.com/en/products/products-overview/storage-systems/dematic-multishuttle/, Accessed on March 13, 2021

Ekren, B.Y., Performance evaluation AVS/RS under various design scenarios: a case study, International Journal of Advanced Manufacturing Technology, vol. 55, pp. 1253-1261, 2011.

Ekren, B. Y., A simulation-based experimental design for SBS/RS warehouse design by considering energy related performance metrics, *Simulation Modelling Practice and Theory*, vol. 98, pp. 101991, 2020a.

Ekren, B. Y., A multi-objective optimisation study for the design of an AVS/RS warehouse, *International Journal of Production Research*, pp. 1–20, 2020b.

Ekren, B. Y. and Heragu, S. S., Performance comparison of two material handling systems: AVS/RS and CBAS/RS, *International Journal of Production Research*, vol. 50, no. 15, pp. 4061-4074, 2012.

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

Ha, Y. and Chae, J., Free balancing for a shuttle-based storage and retrieval system, *Simulation Modelling Practice and Theory*, vol.82, pp. 12-31, 2018.

Heragu, S. S., Cai, X., Krishnamurthy, A. and Malmborg, C. J., Analytical models for analysis of automated warehouse material handling systems, International Journal of Production Research, vol. 49, no. 22, pp 6833–6861, 2011.

Hsueh, C. F, A simulation study of a bi-directional load-exchangeable automated guided vehicle system, *Computers & Industrial Engineering*, vol.58, no. 4, pp. 594-601, 2010.

Jerman, B., Ekren, B. Y., Küçükyaşar, M., Lerher, T., Simulation-based performance analysis for a novel avs/rs technology with movable lifts, *Applied Sciences*, vol. 11, no.5, pp. 1–16, 2021.

Knowledge Sourcing Intelligence, Available: https://www.knowledge-sourcing.com/report/material-handling-market?gclid=Cj0KCQiA1pyCBhCtARIsAHaY_5dyf2o-SdBwDesZBh4m65EFTEHRVJlpy1vm4OHp067FyUGDNHHPOMsaAoJeEALw_wcB, Accessed on March 13, 2021.

Küçükyaşar, M., Ekren, B., Lerher, T., Cost and performance comparison for tier-captive and tier-to-tier SBS/RS warehouse configurations, *International Transactions in Operational Research*, vol. 28, no. 4, pp. 1847-1863, 2020.

Lerher, T., Aisle changing shuttle carriers in autonomous vehicle storage and retrieval systems, *International Journal of Production Research*, vol.56, no. 11 pp. 3859-3879, 2018.

Lienert, T. and Fottner, J., No more deadlocks – Applying the time window routing method to shuttle systems., *ECMS*, pp. 169-175, 2017.

Rhazzaf, M. and Masrour, T., Smart Autonomous Vehicles in High Dimensional Warehouses Using Deep Reinforcement Learning Approach, *Engineering Letters*, vol. 29, no. 1, 2021.

Research and Markets, Available: https://www.researchandmarkets.com/reports/5174357/warehouse-automation-market-with-post-pandemic?utm_source=BW&utm_medium=PressRelease&utm_code=6fvrb6&utm_campaign=1454615+-+Global+Warehouse+Automation+Market+Report+2020+with+Post-Pandemic+(COVID-19)+Impact+Insights%3a+AGV%2fAMR%2c+ASRS%2c+Conveyors%2c+Sortation%2c+Order+Picking%2c+AIDC%2c+Palletizing+and+WMS&utm_exec=chdo54prd, Accessed on June 5, 2021.

Research and Markets, Available: https://www.researchandmarkets.com/reports/5232383/agv-automated-guided-vehicles-and-amr#src-pos-1, Accessed on June 7, 2021.

Roy, D., Krishnamurthy, A., Heragu, S. S., Malmborg, C. J., Blocking effects in warehouse systems with autonomous vehicles, *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 2, pp. 439-451, 2013.

Roy, D., Krishnamurthy, A., Heragu, S. S., Malmborg, C. J., Stochastic models for unit-load operations in warehouse systems with autonomous vehicles, *Annals of Operations Research*, vol. 231, pp. 129–155, 2015.

Acknowledgements

This work was supported by The Scientific and Technological Research Council of Turkey and Slovenian Research Agency: ARRS [grant number: 118M180].

Biographies

Ecem E. Turhanlar is an M.Sc. student in the Department of Industrial Engineering at Yasar University. She received her B.Sc. degree from the Department of Industrial Engineering at Dokuz Eylul University in 2019 as the high honor

student of the department. She is currently working as a fellow for a research project funded by TUBITAK, Turkey under bilateral project with Slovenian research council whose PI is Dr. Banu Y. Ekren. She studies agent-based simulation modelling of an automated warehousing system under Dr. Banu Y. Ekren's supervision. Her research interests are supply chain management, logistics, and smart warehousing.

Banu Y. Ekren is a full time academic in the department of Industrial Engineering, at Yasar University in Izmir, Turkey. Her research focuses on future of education, warehousing, stochastic and simulation modelling, modelling supply chains, simulation-based optimization, and design and analysis of automated warehousing. Banu Y. Ekren holds associate professor position at her university and teaches simulation, stochastic modelling and facility planning and logistics courses at undergraduate level. She has several journal and book chapter publications.