INTERNATIONAL JOURNAL OF LOGISTICS: RESEARCH AND APPLICATIONS
https://doi.org/10.1080/13675567.2022.2138290

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS Check for updates

Autonomous mobile robot travel under deadlock and collision prevention algorithms by agent-based modelling in warehouses

Ecem Eroglu Turhanlar[a], Banu Y. Ekren[b,c] and Tone Lerher[d]

[a]Vestel Electronic Company, Manisa, Turkey; [b]Department of Industrial Engineering, Yasar University, Izmir, Turkey; [c]School of Management, Cranfield University, Cranfield, UK; [d]Faculty of Mechanical Engineering, University of Maribor, Maribor, Slovenia

ABSTRACT
Recent dramatic increase in e-commerce has also increased the adoption of automation technologies in warehouses. Autonomous mobile robots (AMRs) are from those technologies widely utilized in warehouse operations. It is important to design the operation of those robotic systems in such a way that, they meet the current and future system requirements correctly. In this paper, we study flexible travel of AMRs in warehouses by developing smart deadlock and collision prevention algorithms on agent-based modelling. By that, AMR agents can interact with each other and environment, so that they can make smart decisions maximizing their goals. We compare the performance of the developed flexible travel system with non-flexible designs where there is a single AMR dedicated to a specific zone so that no deadlock or collision possibility takes place. The results show that AMRs may provide up to 39% improvement in the flexible system compared to its non-flexible design.

KEYWORDS
Agent-based simulation; deadlock prevention; autonomous vehicle; autonomous mobile robots; deadlock and collision

1. Introduction

With the rapid growth of e-commerce, which is also a result of the unprecedented rise in the number of online shoppers post the outbreak of COVID-19, the intralogistics sector has been facing new challenges. For instance, from the largest retailer, Amazon's same day delivery strategy, it is well observed that delivery time requests are significantly shrinking. To overcome those challenges, the implementation of Industry 4.0 philosophy, developed on a collaboration of automation technologies within facilities becomes crucial. As in most industries, warehouse managers are also very eager to adopt the automation technologies such as goods with RFID tags, helping implement robotic material handling technologies, effectively. For instance, the Global Warehouse Automation Market growth is estimated at a compound annual growth rate (CAGR) of 14% between 2020 and 2026 which is also estimated to be doubled to $30 billion by 2026 (Research and Markets 2021b).

The increased warehouse automation market has also led to growth in the material handling market, supporting reduced costs in labour and transportation. For instance, the material handling market growth is estimated 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). Figure 1 shows the current and estimated growths in the material handling market based on material handling product types.

CONTACT Banu Y. Ekren banu.yetkinekren@cranfield.ac.uk Department of Industrial Engineering, Yasar University, Bornova, Izmir, Turkey; School of Management, Cranfield University, Cranfield MK43 0AL, UK

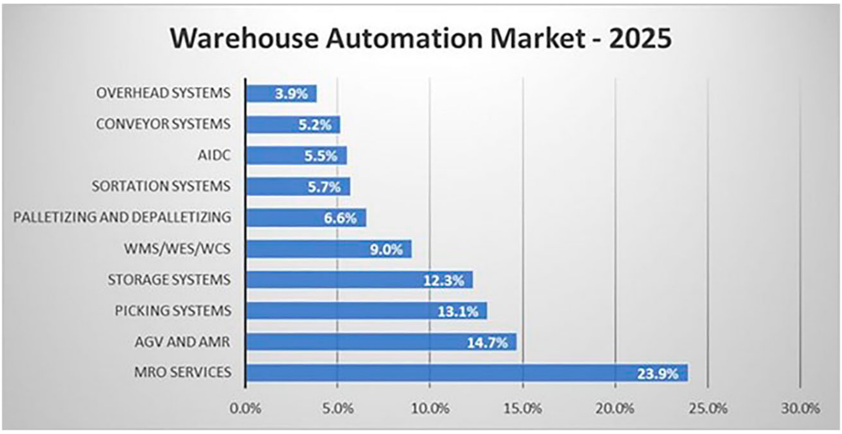

Figure 1. Material handling market growth forecast by product. Source: LogisticsIQ (2022).

Based on Figure 1, the fastest growth among all material handling market products during the forecast period takes place in automated storage and retrieval system (AS/RS) and automated guided vehicle (AGV) technologies. To provide reduced floor space, cycle time, and labour cost in AS/RS technologies, for instance, the Dematic Group provides a mini-load AS/RS technology which is referred to as shuttle-based storage and retrieval system (SBS-RS). SBS-RSs are widely utilised in warehouses and distribution centres to ensure improved system performance such as increased speed, storage density, accuracy, and throughput. Since a dedicated shuttle is assigned to each tier of an aisle, this automated warehousing technology is also referred to as tier-captive SBS-RS in the literature (Lerher, Ekren, Sari, et al. 2015; Lerher, Ekren, Dukic, et al. 2015; Ekren 2017; Ekren and Arslan 2022).

The growth expectation for automated mobile robots (AMRs) is declared to be a CAGR of 35% by 2026 (Research and Markets 2021a). AMR's market size is estimated to reach more than 18% of the overall warehouse automation market share by 2026 (Research and Markets 2021b). While automation is advancing on autonomous vehicle technology rapidly, exploring intelligent operating policies resulting in cost-efficiency, safe and fast process is also becoming an emergent topic. With the recent IT developments, it is possible to equip those robotic technologies with smart algorithms applying dynamic decision-making algorithms utilising real-time data and information from the environment. Our main motivation for the proposed problem is due to the increased utilisation of AMRs in industries as well as the existence of few works in the literature about their flexible travel policies in the system.

The most significant design issue in an AMR problem would be their safe and cost-efficient travel while they complete fast warehouse operations. In another word, efficient travel of those AMRs is important, so that they never collide and cause deadlock within the system while they complete fast transaction operations. The main research questions in this paper are given to be:

RQ1: How AMRs can travel flexibly in a warehouse plant without colliding and being congested while the system also results in high performance?

RQ2: Which travel policies of AMRs, flexible or dedicated, and under which warehouse designs would work better?

In RQ1, we develop smart control policies, for the flexible travel of AMRs by preventing collision and deadlock in the system. We utilise an agent-based simulation modelling approach to model and test the performance of the developed algorithms. Because of the system complexity and necessity of real-time information tracking and the aim of dynamic smart decision making, multi-agent simulation modelling is found to be appropriate. While developing that travel policy, we observe several performance metrics from the system such as the average flow time of a transaction, maximum flow time of transactions, and ratio of average waiting time to average flow time performance metrics. In RQ2, after developing the flexible travel design, we experiment it under different warehouse designs and compare each, with its dedicated (i.e. non-flexible) system design.

From the literature, we observe limited works focusing on developing deadlock and collision prevention algorithms for autonomous vehicles. Table 1 in Section 2, summarises the existing studies from literature. From those, it is observed that the current works mostly focus on the determination of deadlock prevention policies by applying several modelling approaches. Simulation modelling is one of the most widely utilised modelling approaches providing a powerful modelling tool for such complex system designs that also integrates dynamic (e.g. agent-based) decision making algorithms.

Note that from a managerial perspective, an efficient flexible travel design of AMR might also result in a reduced number of autonomous vehicles in the system. By that, the initial investment cost as well as the total operating cost of the system may also decrease. In this paper, by studying smart deadlock and collision prevention control algorithms for flexible travel of AMRs, we aim to enable safe and efficient travel for AMRs, also resulting in increased performance of the system. We are motivated in this work by SBS-RS warehouses, where in their common designs, shuttles (i.e. AMRs) are mostly tier-captive and cannot travel between aisles. In such a system, because there exists a great number of shuttles in the system due to having a dedicated shuttle in each tier, the average utilisation of those shuttles are mostly very low compared to the lifting mechanism dedicated as a single one in each aisle. Allowing those shuttles to travel between multiple aisles within a tier, might also help balance the average utilisation of shuttles and lifts with the decreased total number of shuttles in the system. In practice, technology solution providers may prefer applying a dedicated zone policy for an AMR to ignore development and embedding such complex control algorithms in those robots (Lerher 2018). However, with the help of recent IT and Industry 4.0 developments, integration of such complex control algorithms into those robotic technologies might be easier (Yau et al. 2020;Chen et al. 2021). The main motivation for this work is to explore such smart algorithms.

This paper is organised as follows. In Section 2, we explain the current related literature works about the problem. In Section 3, we describe the studied warehouse designs. In Section 4, we explain the studied flexible system design in detail along with the agent-based simulation modelling and, deadlock and collision prevention control algorithms. In Section 5, to do a sensitivity analysis, we conduct experiments in an experimental design manner. Last, we summarise the work by also providing some potential future studies.

2. Literature review

The smart warehouse and logistics concept and the challenges are presented well by Winkelhaus and Grosse (2022) and Fragapane et al. (2022). The related literature works are summarised in Table 1 by categorising the works based on their studied system types, autonomous vehicle types, applied methods, strategies to prevent deadlock and collisions, and objective of the work. In this section, we explain those studies in detail.

Note that an AGV system might be considered as a single tier warehouse system where AMRs (i.e. AGVs) travel on the ground level. According to the work of Hsueh (2010), an EX-AGV can travel along its shortest path and can change its load to prevent possible collisions. The closest vehicle assignment for a load is suggested as a vehicle dispatching rule among the five proposed rules. Simulation results show that the EX-AGV system is better than the tandem AGV system in terms of system performance, even in cases with a higher number of AGVs.

A system, reserving the path of AGVs for collision avoidance is presented by Cossentino et al. (2011). The proposed conservative policy is applied to an agent-based simulation model to let AGVs make autonomous decisions in the warehouse. Krnjak et al. (2015) propose a decentralised

Table 1. Literature works on AMR.

Literature	System type	Vehicle type	Applied method	Deadlock prevention strategy	Objective
Hsueh (2010)	Automated warehouse	AGV	Mathematical modelling, Simulation	The exchange operation of loads on vehicles moving along their shortest path	Develops an EX-AGV system to prevent deadlocks and conflicts. Find a vehicle dispatching rule resulting with higher system performance
Vivaldini et al. (2010)	Intelligent warehouse	Robotic forklifts	Heuristic, Simulation, Dynamic Programming	Time window routing	Present a router system where smart changes for priority assignment of tasks as well as robotic forklift route definitions in conflict cases
Cossentino et al. (2011)	Automated warehouse	AGV	Simulation	Path reservation policy	Optimise the number of AGVs in the warehouse
Krnjak et al. (2015)	Automated warehouse	AGV	Simulation	An algorithm for decentralised AGV control providing autonomously collision detection for vehicles in communication range and resolving conflict scenarios	Provides algorithms for vehicles that can plan the deadlock-free motions autonomously, by considering nonholonomic vehicle constraints
Draganjac et al. (2016)	Automated warehouse	AGV	Exact method, Private zone mechanism, Simulation	Zone control and vehicle priority policy. Vehicles stop temporarily or it can remove the lower priority vehicles in the system that might cause conflict.	Autonomous route decisions and travel co-ordination
Roy et al. (2016)	Automated warehouse	Autonomous vehicle	Simulation	The cross-aisle protocol. A switching rule is applied where vehicles get alternate use of the cross-aisle. A vehicle need to wait to obtain cross-aisle access.	Evaluate the effects of blocking on system performance Develop protocols to address vehicle blockings
Lienert and Fottner (2017)	SBS-RS	Shuttle	Mixed graph modelling	Time window routing	Present a concept for safe and efficient vehicle movement
Zhou et al. (2017)	Automated warehouse	Mobile Robot	Distributed algorithm	A real-time tracking algorithm is developed for deadlock and collision prevention. AMRs stop and resume through their travels.	Investigate the policy of deadlock and collision avoidance in multi-robot systems, where each robot has a predetermined and intersecting path
Ha and Chae (2018)	SBS-RS	Shuttle	Free balancing/ Simulation modelling	Zone planning	Decrease the number of shuttles and improve the throughput performance
Lienert, Wenzler, and Fottner (2020)	Automated warehouse	Mobile Robot	Simulation	Time window routing	Present and discuss different reservation mechanisms for avoiding deadlocks
Sgarbossa et al. (2020)	Automated warehouse	Mobile Robot	Heuristic, NSGA-II	Zoning	Presents product assignment to two different zones robot picker and human picker.
Rhazzaf and Masrour (2021)	High dimensional warehouses	Autonomous vehicle	Deep Reinforcement Learning	Dividing the warehouse into a grid of low dimension zones	Achieve good performance in terms of speed and number of movements

AGV control algorithm for deadlock prevention in an automated warehouse. The autonomous AGV path planning, and coordination of motion are proposed in the paper of Draganjac et al. (2016). They apply a private zone exact mechanism to prevent deadlock. The concept of prevention includes stopping or removing fewer priority vehicles in conflict. Zhou et al. (2017) study a real-time distributed algorithm for deadlock and collision avoidance by stopping and resuming robots. Lienert, Wenzler, and Fottner (2020) present a deadlock avoidance approach that includes several route reservation mechanisms by time window routing method. They evaluate the results by using the mean values of the simulation results. Vivaldini et al. (2010) present a router system for an intelligent warehouse where smart changes can be made for priority assignment of tasks as well as robotic forklift route definitions in conflict cases. The algorithm is based on Dijkstra's shortest path and the time window approaches. The authors also apply computer simulation tests to validate the efficiency of algorithms under various working conditions.

Other existing studies are considered as multi-tier systems. The deadlock prevention strategy in the work of Ha and Chae (2018) separates systems into non-intersecting zones, limiting each zone to one vehicle in a tier-to-tier SBS-RS. The aim of that work is to show that working with few number of shuttles in the system would still work better than a basic system design. Roy et al. (2016) develop interference protocols and investigate blocking delays at the cross-aisle in AVS/RS. The simulation studies demonstrate the significant contribution of blocking delays on the transaction cycle time. Lienert and Fottner (2017) apply a time window routing approach for the travel of shuttles safely in tier-to-tier and aisle-to-aisle system configurations. Rhazzaf and Masrour (2021) divide the high dimensional warehouses into low dimensional ones to achieve good performance from the system.

Different from those deadlocks and collision focused works, there are tier-to-tier SBS-RS works, where simple deadlock collision and prevention algorithms are implemented by not allowing a second AMR (i.e. shuttle) at the same tier of the current AMR (Eroglu and Yetkin Ekren 2020; Kucukyasar and Ekren 2020a, 2020b; Küçükyaşar, Ekren, and Lerher 2021a, 2021b). Besides, Turhanlar, Ekren, and Lerher (2021) study an initial algorithm development before this current work in an aisle-to-aisle SBS-RS.

In this paper, different from the current literature works, we study a corridor-based warehouse layout, where AMRs can travel freely between any of those aisles without experiencing any deadlock and collision cases. Note that, most of the warehouse systems consist of high-bay shelves under corridor-based designs. The proposed system can also be utilised for both single- and multi-tier warehouse systems where in multi-tier cases, AMRs could travel between multiple aisles within a tier. We present the studied system details in Section 3.

3. Flexible and dedicated warehouse design configurations

This section provides information about the physical warehouse configuration of the proposed flexible system design and the dedicated system design which is developed for comparison purposes. Note that in the dedicated system, it is assumed that a single AMR is dedicated for a specific number of aisles zone so that it never comes across with another AMR in the system. Since in the proposed flexible system design, all AMRs can travel flexibly between aisles, there is a possibility of encountering those AMRs during their travels. Therefore, in this system design, we develop smart anti-collision algorithms based on inter-vehicle communications described in detail in the next section. Once again, since there is no possibility that AMRs can come across in the dedicated system design. Hence, in that system design, we do not consider any deadlock and collision prevention algorithm.

Figure 2 shows the studied two warehouse configurations. While Figure 2(a) shows the design where AMRs can travel flexibly between aisles, Figure 2(b) shows the design where a single AMR is dedicated for a specific zone in terms of a number of aisles. For a fair comparison of those two policies, we assume that there is the same number of AMRs in both system configurations.

In Figure 2(a), as mentioned AMRs may come across, while in Figure 2(b), AMRs never come across. Here, we compare the performance of those two systems designs under the developed deadlock and collision prevention algorithms. Remember that Figure 2(a) design can also be considered as a single tier of a multi-tier warehouse. Hence, both warehouse types with AGVs or multi-tier storage and retrieval (S/R) systems can use the outcomes of this work. In Figure 2(a), to prevent collision and deadlock in the system, we define 'escape', 'decision' and 'waiting' points across each aisle, where AMRs can escape to free the way for another AMR, to make decisions for their next step routes and, wait until a decision point becomes available, respectively. We explain the details of the working principles of those points as well as developed algorithms in the following section. Here, aisles are composed of storage racks on both, left and right sides. The total number of storage bays in the warehouse might differ based-on the capacity requirements. There are buffer locations across of end points, where storage transactions arrive and retrieval transactions are dropped-off. In the dedicated system, the AMRs would solely travel within their dedicated zones. Note that this design is studied for comparison purposes with the flexible design. We detail the flexible system design along with the agent-based modelling in the below section.

4. Flexible system design

In this section, we explain the developed agent-based modelling approach for the flexible system design where AMRs can travel between aisles. Here, we also present the developed deadlock and collision prevention algorithms.

4.1. Agent-based modelling approach

We treat the AMRs as autonomous intelligent agents which can interact with each other and sense real-time information from their environment. We utilise an agent-based simulation modelling approach to model and test the performance of the developed algorithms. Because of the system complexity and necessity of real-time information tracking, and dynamic decision making, multi-agent simulation modelling is found to be appropriate. The decision procedures of the agents, which are the simulated actors in the simulation models are clearly defined at the micro-level. The macro-level structure of the system is formed by these micro-level decisions as a result of

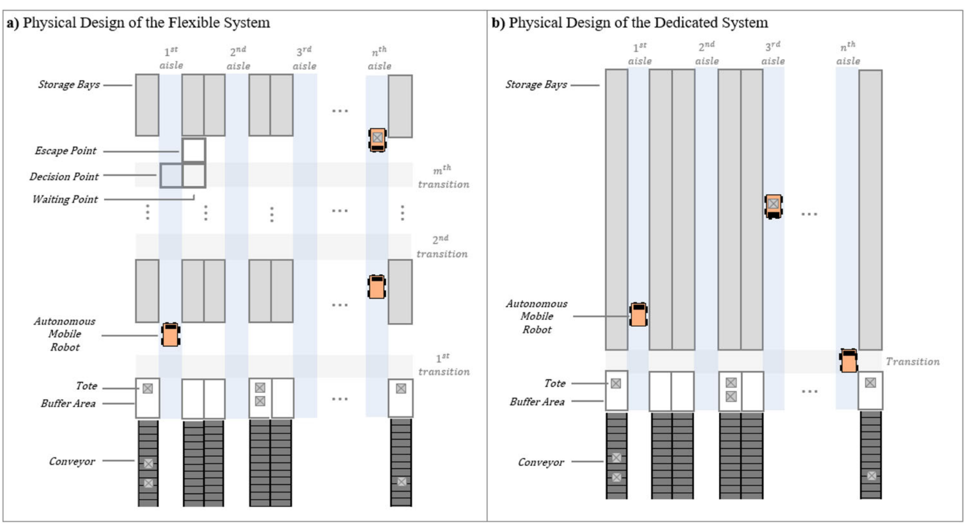

Figure 2. Flexible and dedicated warehouse configurations.

communication in the system. Consequently, we utilise multi-agent simulation modelling to test the performance of the proposed algorithms. The main interactions and connections of agents are shown in Figure 3. According to that, AMR and demand agents can interact with each other. In that figure, AAMR and DAMR represent active AMR and deadlock AMR, respectively, whose details are explained in the sub-section *b*, here.

In the model, two types of agents are defined: AMRs and arriving demand transactions. They are created as intelligent agents in the model, so that they become dynamic objects that can sense their environment in real time and make autonomous decisions towards their goals. Those multi-agents are also able to communicate with each other. The main smart decisions that AMRs can take are the most proper transaction selection waiting in their queue based on their goals, and travel route determination through their destination addresses. Except for the AMR agents, there is a single demand agent, which provides real-time information (i.e. type, address, etc.) about the arriving transaction demands. As shown in Figure 3, the movement of an AMR can take place only after the movement permit signal, is given by another. This rule also applies for the trigger command. The details of those commands are explained in sub-section *b*.

A snapshot from the animation part of the developed simulation model is shown in Figure 4. For instance, in that figure, in every twelve adjacent bays, there is a transition road connecting the aisles. An AMR can transit to another aisle by using that transition road. At each intersection point of an aisle and at each transition road there are that triple-points: decision, waiting, and escape points. An AMR enters an aisle to store or retrieve a load into/from a bay. After completing its process, it travels to the closest decision point to decide its next step. The travel policy is detailed in the following sub-section.

The simulation model is developed in Arena 16.0 commercial software. It is verified and validated by debugging and animating the model. Several degenerate tests are applied on the models to test whether they produce consistent results. For example, the mean arrival rate of transactions is increased, or the velocity of AMRs is decreased in the system, and it is observed that the AMR utilisation levels as well as the average flow time of a transaction performance metrics tend to

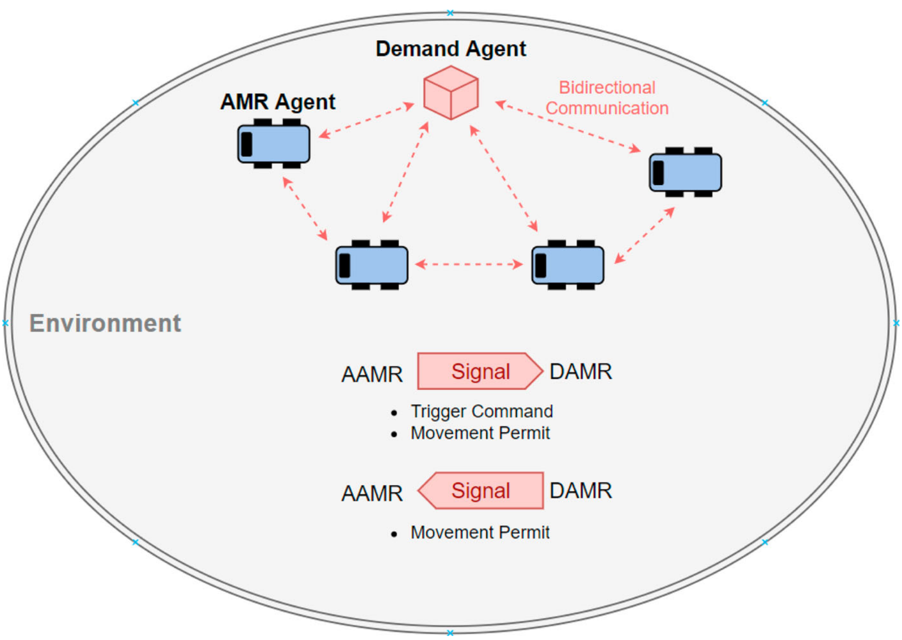

Figure 3. Communication and interaction between agents.

increase. Besides, we also create intended scenarios to check if any collision takes place during the simulation runs. Even if the arrival rate is increased, no collision or deadlock happens during the complete runs. Besides, to check the variability between replications, the half-width values of the simulation results are observed.

The main simulation assumptions that are considered in the model are summarised below:

(1) Poisson distribution is considered for the mean arrival rate (λ) of S/R transactions (Roy et al. 2014; Marchet et al. 2013; Ning et al. 2016; Ha and Chae 2019; Eder 2019; Wu et al. 2020; Ekren 2020a, 2020b). Here, to run the models under highly utilised conditions as most practitioners expect, we adjust the mean values so that we observe 95% average utilisations for the AMRs.
(2) S/R request addresses are created randomly.
(3) The unload and load times of totes are three seconds (Ha and Chae 2018).
(4) The acceleration and deceleration delays are the same (e.g. 2 m/s^2 or 3 m/s^2 depending on the experiment).
(5) Two adjacent bay distances are the same and 0.5 m (Ning et al. 2016; Ha and Chae 2018; Eder 2019).
(6) AMRs do not break down in the model.
(7) An AMR travels to the closest decision point to wait for a new transaction.
(8) There are as many buffer locations as there are aisles in the warehouse, where the storage transactions arrive at, and the retrieval transactions are dropped off.

The simulation model is run for one month with a one-week warm-up period, and five independent replications. The main idea and the significant task in agent-based modelling would be the determination of the best agent behaviours improving the objective performance metrics in the system. In the following sub-section, we explain the defined agent behaviours after a long trial-and-error work.

4.2. Intelligent agent-based behaviours

The main task of an AMR is to select the best transaction from its queue improving its goal. Before selecting a transaction, the AMR gathers and evaluates information from its environment which

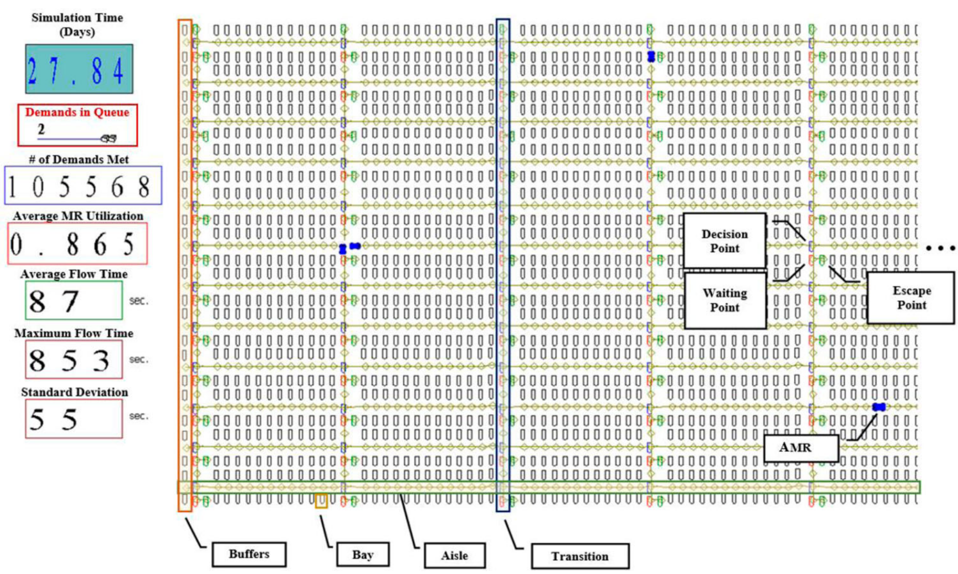

Figure 4. Simulation animation snapshot.

are: transaction demand attributes (e.g. storage or retrieval) waiting in its queue, their storage addresses and, current waiting times of the transactions, traffic density conditions through the bay addresses, the other AMRs' current and target destination addresses, etc. Namely, before an AMR selects a transaction from its queue, it observes and evaluates all those information and, makes a smart decision to improve its goal. The pseudo codes for that transaction selection procedure are given in Figure 5.

According to Figure 5, the algorithm starts with the availability of at least one transaction waiting in the queue ($n(T) \geq 1$) and at least one AMR ($n(A) \geq 1$) in the system. If there is a single transaction waiting to be processed in the queue (e.g. $n\ (T) = 1$), the available AMR (m) closest to the location of that transaction (t) ($min\ D(t,\ m)$) is assigned to that transaction. Otherwise, if there is more than a single transaction in the queue, then the current waiting times of those transactions ($W(t)$) are checked. If there is any transaction waiting longer than the current average waiting for time per transaction performance metric ($W(t) > W^{\text{avg}}$), to decrease the average waiting time in the system, the longest waiting transaction is selected and matched with the closest available AMR. Here, the current average waiting time per transaction performance metric represents the averages of the waiting times of all the transactions so far. Otherwise, if there is no transaction waiting longer than the current average waiting time per transaction, then that match is performed based on the aisle density ($AD(t)$) value through the route of the waiting transaction's storage address. Here, AD (t) represents the number of currently being processed transactions in the aisle of transaction t's address. As a result of that, the transaction having the lowest aisle density ($min\ AD(t)$) and the closest location to an available AMR ($min\ D(t,\ m)$) is matched. Here, first, the priority is assigned to the transaction with the least density parameter (i.e. min $AD(t)$), then the closest available AMR is assigned to that transaction, t. To do that decision correctly, the AMR agent collects real time information from all waiting transactions' aisle address density conditions.

After a transaction is matched with the AMR, AMR starts its travel immediately. If the transaction is a storage process, then the AMR travels to the buffer location to pick up the load. Note that, there are as many buffer locations as there are aisles in the layout. The incoming storage transactions and the outgoing retrieval transactions arrive at those locations. If the transaction is a retrieval process, the AMR travels to the transaction's storage address to pick up the load.

An AMR always travels between two decision points until it reaches at the last decision point connected with its destination aisle. Those decision points are located through the transition roads located across aisles (see, Figure 2(a)). While the AMR is travelling to reach a storage address, after its last decision point, it travels to the destination aisle address called 'storage bay', immediately. When it arrives at the destination address, it stores or retrieves the load, based on the transaction type. If an AMR has already completed a storage process and, there is no other transaction already assigned to that AMR, then it moves to the closest available decision point not to block the current aisle. It waits there until a new transaction is assigned for itself. If the AMR has completed a retrieval process, then it moves to the closest decision point from the buffer area where the load is dropped off. Here, the AMR travels to the buffer location linked with the load's aisle location. For instance, if the retrieval transaction is located at the second aisle, then the AMR drops-off the load at an available buffer location closest to the second aisle location. It is not necessary for an AMR to stop at its following two decision points, if those points are available during its travel and, there is no other AMR heading to any of those points. The AMR understands the availability of those decision points by evaluating the current information from its environment and, it makes this non-stopping decision before it starts its travel.

Note that, we define 'active' and 'deadlock' AMRs in the model, where the former represents the AMR that is currently active to make decisions. The latter represents the AMR tending to cause a deadlock for an active AMR (AAMR). When an AAMR encounters with a deadlock AMR (DAMR), it follows those policies: if the direction of the DAMR is to move through the direction of the AAMR, then the AAMR waits for the DAMR to leave that decision point. This is because, probably that DAMR is already waiting at that decision point to proceed with its route. Otherwise, the AAMR

Data: V: Set of Autonomous Mobile Robots,
A: Set of Available Autonomous Mobile Robots,
T: Set of Waiting Transactions,
$A \subseteq V$, $m \in A$, $t \in T$.
Result: m and t ;
$D(t, m)$: Distance between the address of t and the current location of m,
$W(t)$: Waiting time of t,
W^{avg}: Average waiting time of all t s,
$AD(t)$: Aisle density of the address of t.
while $n(A) \geq 1$ **and** $n(T) \geq 1$ **do**
  if $n(T) = 1$ **then**
    Match: m and t having $min_{D(t,m)}$
  else
    if $W(t) > W^{avg} \quad \exists t \in T$ **then**
      Match: m and t having $min_{D(t,m)}$ where $W(t) > W^{avg}$ for t
    else
      Match: m and t having $min_{D(t,m)}$ where $AD(t) = min_{AD(t)}$ for t
    end
  end
end

Figure 5. The pseudo codes for transaction selection procedure of AMRs.

searches for an alternative route for itself. Here, the alternative route is accepted by the AAMR, if it can arrive at its destination address without increasing its pre-intended distance. In the case that an alternative route does not exist, as well as, if the DAMR is idle at the blocking decision point, then the AAMR triggers that DAMR by sending a signal, to move to the most advantageous point, to free the way for itself. In this case, the trigger policy plays a significant role in the prevention of deadlocks and collisions. The details of the trigger policy in the control algorithms are explained in Figures 6 and 7 flow charts.

The main idea of a trigger policy is to alter any potential deadlock or collision problem in the warehouse. Here, the trigger process starts with an AAMR at a decision point, where it tends to travel to another decision point through its destination address. According to Figure 6, the trigger policy starts when the AAMR is to encounter with an AMR through its intended decision point. Note that since both decision and waiting points are located through transition (AMR's travel) ways, when either the waiting or the decision point is occupied by an AMR through that AAMR's route then, a deadlock is assumed to happen. Then, that AMR is referred to as DAMR. If that DAMR is already triggered to move to a waiting point by another AMR, then the AAMR triggers that DAMR to move to the closest escape point since, since that waiting point would be an obstacle for itself. If the closest escape point is not available, then the DAMR is triggered to the closest decision point not to block that AAMR's travel route. In the trigger policy, the priority is always given to the AMR which is to leave an aisle.

Figure 7 shows the travel procedure of the DAMR triggered by the AAMR. Mainly, a triggered busy DAMR takes an action by helping the AAMR's continuous travel. A triggered idle DAMR waits at its triggered place until it selects a transaction to process. If AAMR is idle while travelling to the decision point and the path through the left aisle is available, then the DAMR triggers the

AAMR to the left side escape point and, it continues its way. If that path is not available, then the DAMR triggers the AAMR to the left side decision point and, it continues its way.

To clarify the Figures 6 and 7 flow charts, we provide some practical deadlock and collision cases in Figure 8 along with the ways how they are resolved. In that figure, in each case, the first left top figure shows how the potential deadlock and collision case tend to happen. The second and the third ones show, how the problem is aimed to be resolved and ultimately resolved, respectively. Under each figure, we also explain, how the solution is produced. Note that in Figure 8, the target destination storage bay address of each AMR can be matched by its equivalent node colour. For instance, in the first case, the blue colour AAMR tends to travel to its storage address, where there is a blue node stored in a bay.

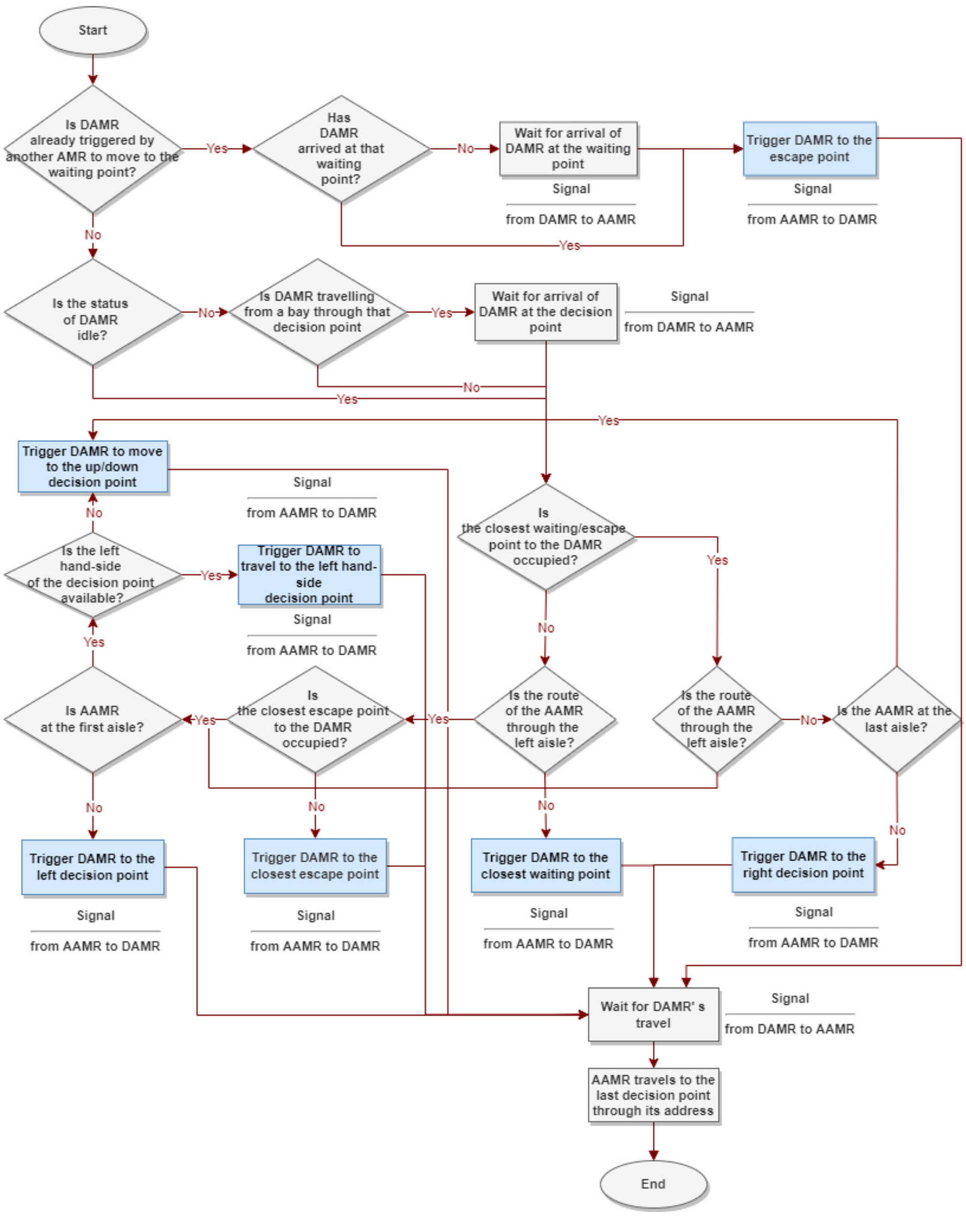

Figure 6. The trigger policy of AMRs.

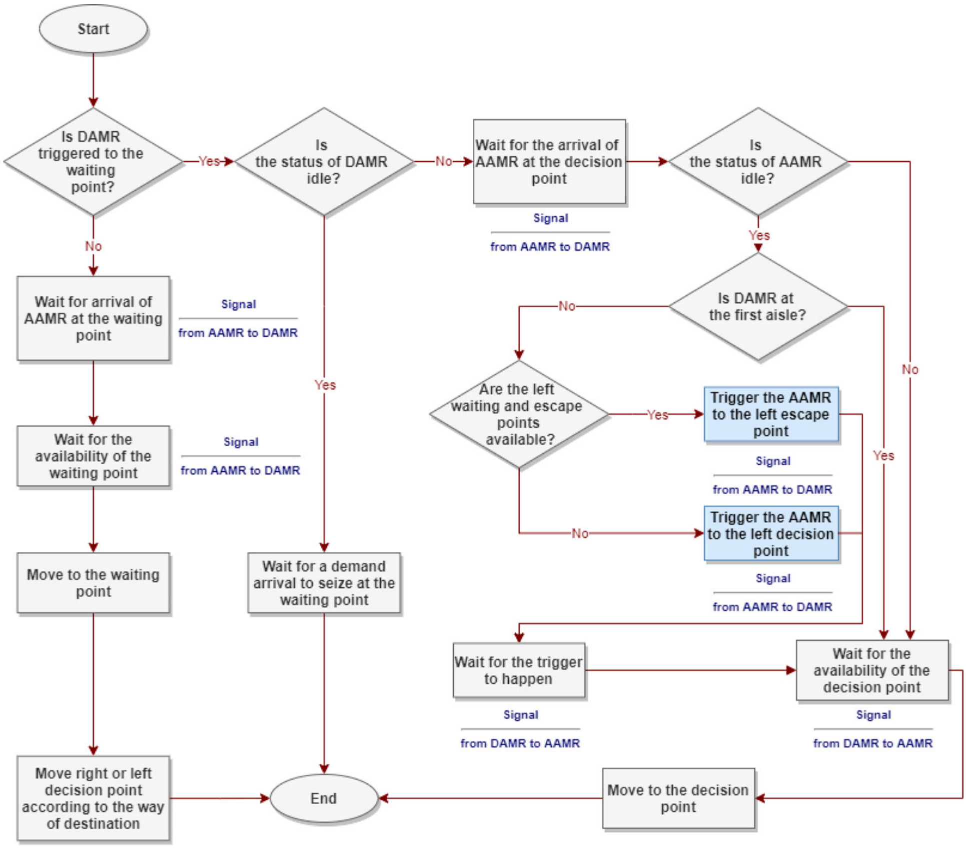

Figure 7. The movement procedure for the triggered DAMR.

To summarise, in the first case, the blue vehicle is the AAMR triggering the green vehicle to the escape point. Later, it continues its travel in the direction of its address. As soon as the blue AMR leaves the decision point, the green AMR travels to the decision point to continue its way. Later, the white AMR enters the area and notices the green AMR which tends to cause a deadlock for itself. The white AMR waits at its current decision point until the green AMR leaves the transition road. In Case 2, the white AMR is waiting idle at a decision point. It is triggered to the waiting point by the blue AMR. The green AMR waits for the blue AMR to leave the decision point, so that it can move to that point. Note that, in the third figure, the white AMR is assigned for a transaction shown in the white colour node. Then, the white AMR exits from the waiting point and moves towards its destination bay address. Last, idle green and blue AMRs that have completed their processes return to the closest decision points and wait for a new transaction to select to process.

To show how the developed control algorithms perform and to do a sensitivity analysis, we conduct experimental design work under different warehouse rack designs in terms of number of bays and aisles and, the AMR designs, in terms of velocity profiles and the number of AMRs. We show the details of that experimental work in the below section.

5. Experimental design

To observe the performance of the developed algorithms, we conduct an experimental design study on the developed models. The considered factors in the experimental design along with their levels are summarised in Table 2. According to that, we mainly define four factors: warehouse capacity

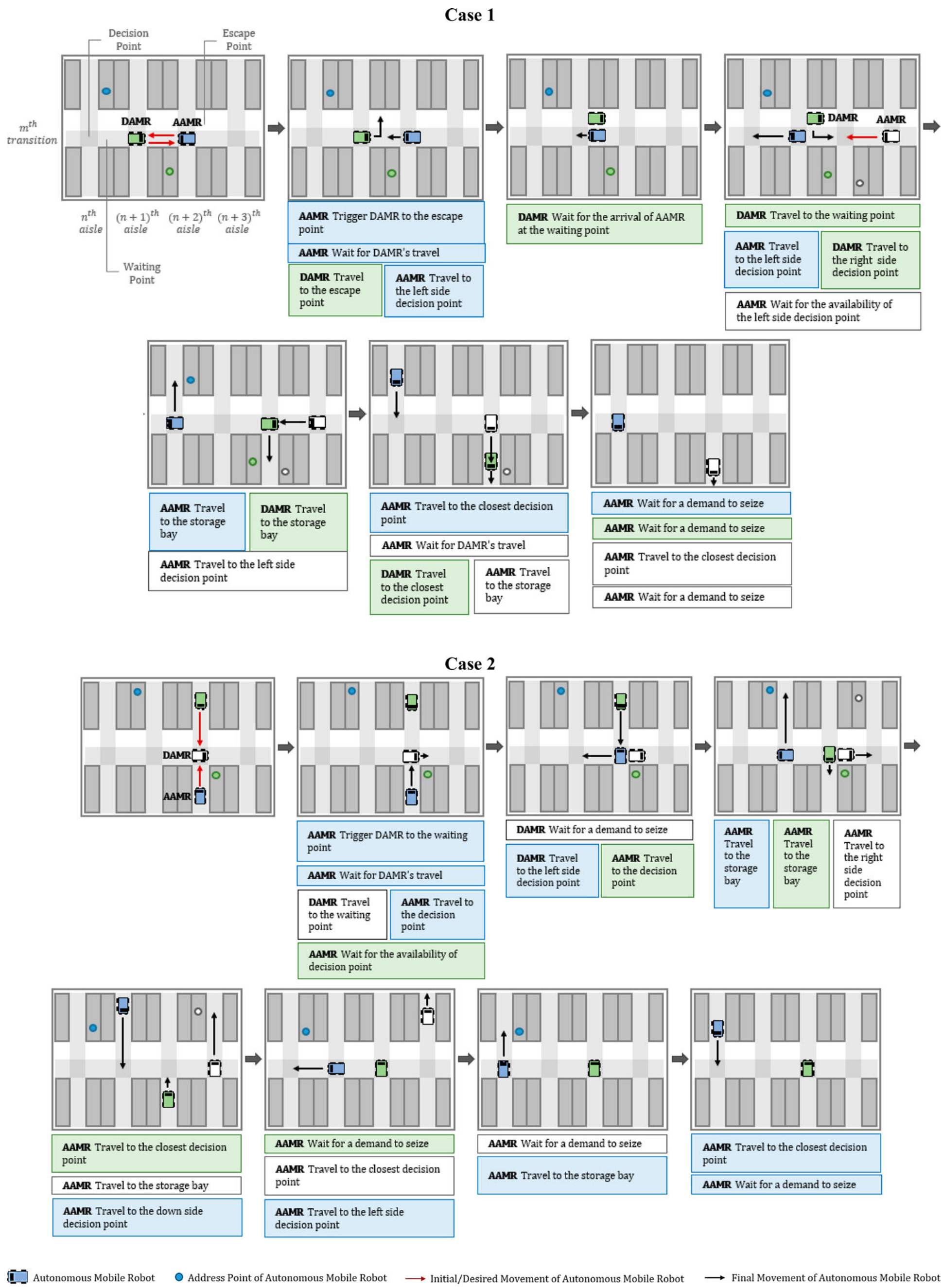

Figure 8. Two deadlock and collision cases were resolved by developed deadlock and collision algorithms.

(C) representing the total number of storage bays, number of aisles (A), velocity profiles of AMR in terms of maximum speed and acceleration/deceleration values, and number of AMRs (N). We apply those values for two systems designs: flexible system design (FSD) and dedicated system design (DSD). Remember that here, the dedicated system design is considered for comparison purpose. From Table 2, by considering each combination of those factor levels in experimenting, we conduct $2^5 = 32$ experiments.

For instance, Figure 9 shows the layout designs for both flexible and dedicated systems, for the experiment $C = 1{,}200$; $A = 6$; $N = 3$. Figure 9(a) shows the layout for the FSD and Figure 9(b) shows the layout for the DSD. Note that in Figure 9(a), there are 17, 16, 17, 16, 17, and 17 numbers of bays in either side of each aisle, in order. That warehouse is divided into seven zones as understood from seven transition roads. In Figure 9(b), there is a dedicated AMR assigned to serve for two following corridors. To be able to make a fair comparison, all the design parameters C, A, and N are designed as same in both warehouses. The selection rule of AMRs in DSD is completed based on the shortest travel distance of AMRs. In DSD, an arriving transaction's bay location is assigned randomly, as well.

Figures 10–12 show the simulation results based on three critical performance metrics: average flow time of a transaction (F_{avg}), maximum flow time of transactions (F_{max}), and the ratio of average waiting time to average flow time (W_R) performance metrics. Those results are the average values of five simulation replications. All the simulation results along with the 95% confidence intervals are given in the Appendix part.

Figures 10 and 11 show the conducted experiment versus F_{avg} and F_{max} performance metrics, respectively. Here, F_{avg} represents the average time a transaction request spends in the system. This value is calculated by summing up each transaction's total time spent in the system, then dividing this value, by the total number of transactions processed in the system. Note that, F_{avg} includes waiting times of transactions in the AMR queue, as well. F_{max} represents the maximum flow time of overall processed transactions in the system. This performance measure is significant with the recent supply chain responsiveness targets. By the decreased F_{max} and F_{avg} values, a company would become more responsible to its customers.

Figure 12 shows the conducted experiment versus the W_R performance metric. W_R is the ratio of the average waiting time of a transaction to the average flow time of a transaction. This performance metric might be significant in observing what ratio of flow time would belong to the waiting time of a transaction. For instance, in Figure 12, in FSD, this ratio is 59% in experiment $C = 3{,}600$; $A = 12$; $N = 3$. Hence, the average waiting time of a transaction can be estimated at 59% of 103.1 sec. where 103.1 sec. represents the average flow time of a transaction in FSD, observed in Figure 10. W_R might be a proper performance metric in understanding, how waiting time could be decreased within the F_{avg}. From Figures 10–12, it is observed that in most of the cases, better performance metrics are obtained in FSD than in DSD. This difference may change based-on the warehouse design where we comment on those results in detail below.

The overall observations from the experiments are summarised below:

- From Figures 9–10, it is observed that when the number of AMRs is high in the system, the performance gap between FSD and DSD increases. As observed in Figure 10, this gap can reach up to 39% for the average flow time in experiment $C = 3{,}600$; $A = 6$; $N = 4$. This is probably because that under that condition, an advantageous condition takes place in FSD compared to DSD. Namely, when the number of AMRs increases in FSD, the possibility of finding a closer transaction to process for an AMR may increase, due to more distributed AMRs throughout the area.
- In higher capacity warehouse designs, the FSD outperforms DSD results. This is probably due to increased travel time in DSD, in the high-capacity case, by the deeper aisles design to meet the required total capacity. Note that, when the number of aisles and the warehouse capacity is high, the FSD's performance is improved drastically. This is probably because the deep shape of the

Table 2. Experimental design table.

System design	Warehouse design			Vehicle scenario
Type	C	A	V	N
FSD	1200	12	2; 3	3
DSD	3600	6	3; 3	4

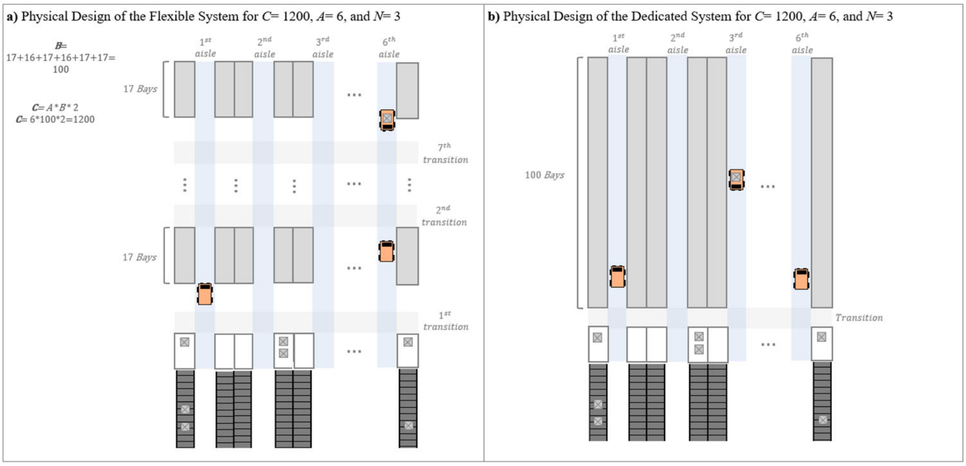

Figure 9. Physical layout of the warehouses when $C = 1{,}200$; $A = 6$; $N = 3$.

footprint decreases when the number of aisles increases. Hence, we may come up with that: the FSD policy plays a significant role under highly capacitated warehouse designs.

- Under the low number of aisles cases, in either warehouse capacity case, FSD tends to perform better than DSD. This is probably because, by the decreased number of aisles, the footprint shape becomes more deeper. This case increases the travel time of AMRs in DSD. However, in FSD, such a layout would require more transition roads, helping AMRs enter aisles efficiently.
- When both, the number of AMRs and the velocity profiles increase in designs, the FSD becomes more advantageous especially, from the F_{max} performance metric.
- Waiting time ratios are typically higher in DSD than in FSD. This is probably due to having a single AMR in a dedicated aisle zone in DSD. Namely, an arriving transaction request waits for that AMR, until it becomes available to be processed.

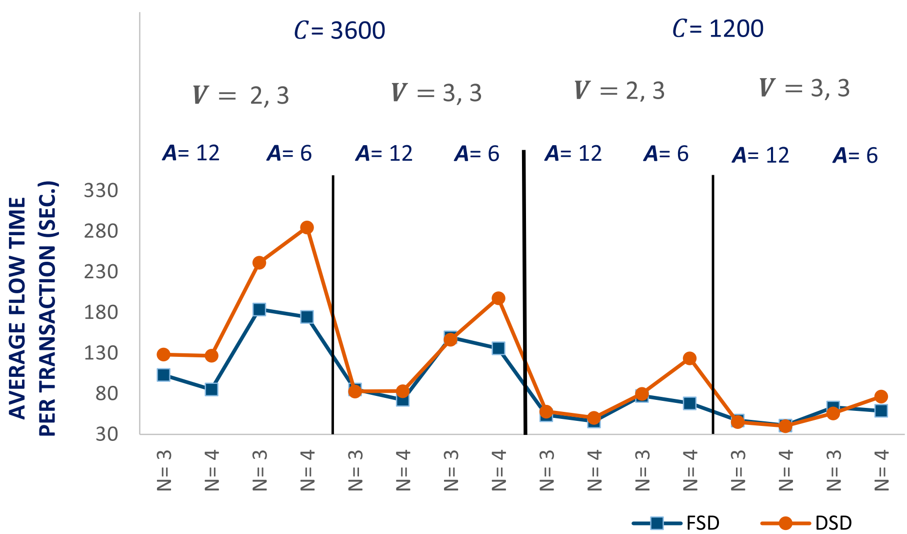

Figure 10. Experiment versus F_{avg}.

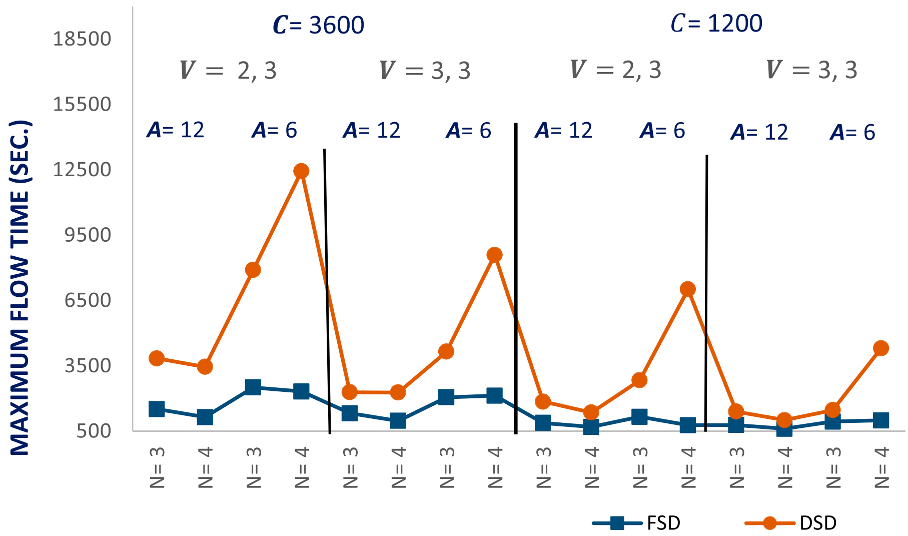

Figure 11. Experiment versus F_{max}.

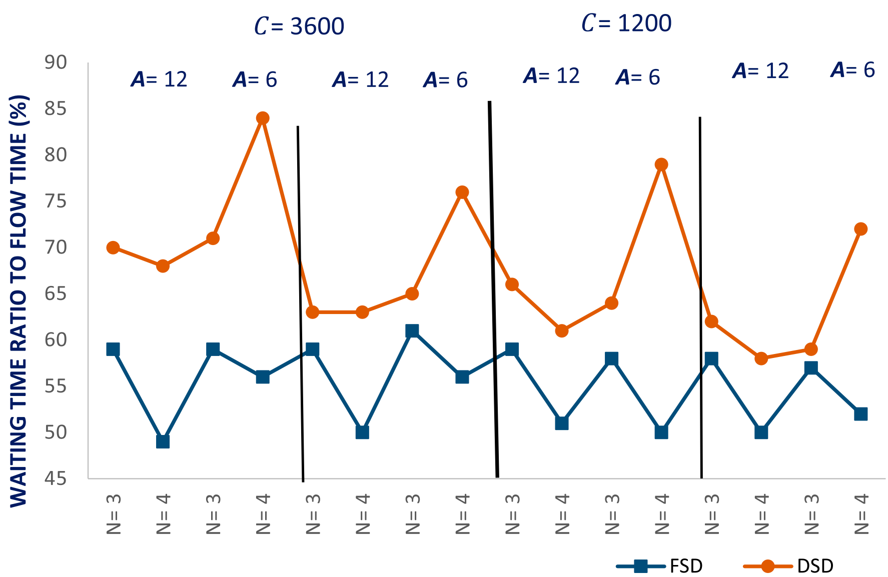

Figure 12. Experiment versus W_R.

- The results show that, from a multi-objective perspective, FSD with an increased number of aisles and AMRs, might be preferable for an AMR warehouse.

In addition to the above findings, an FSD might become advantageous from the decreased number of buffers considered in that system design. Namely, due to the flexible travel property of the AMRs, and their ability to reach any buffer locations in the system, an FSD may be composed of decreased number of buffer locations compared to a DSD. Note that those buffer locations may be considered as increasing the total investment costs because those locations are directly connected with the lifting mechanisms in a multi-tier warehousing system. Hence, by the decreased number of buffers, an FSD may also provide cost efficiency from the system's initial investment cost perspective.

5.1. The potential investment cost reduction by limiting the number of buffer areas

We also simulate the FSD under reduced buffer locations to observe the effect of the reduction of a number of buffer areas on the system performance. We simulate three different buffer scenarios as there is a buffer location at each aisle, (e.g. which we currently completed), there is a buffer location in the middle of every two-aisles, and there is a buffer location in the middle of every three aisles. Even under those reduced buffer locations, the FSD produces better results than the DSD. Hence, a cost reduction might be realised by FSD, when investing in such systems.

5.2. The potential of investment cost reduction by reducing the number of AMRs

By the decreased F_{avg} in FSD, the possibility of working with fewer AMRs is also tested. For instance, we experiment five and six numbers of AMRs for FSD and DSD, respectively. Under the same transaction arrival rates, it is observed that even under decreased numbers of AMRs, the FSD produces better results than DSD (i.e. roughly, 15% better results for F_{avg}, and 53% better results for F_{max}). Therefore, the exploration of more racking design scenarios under the decreased number of AMRs in FSD might be worth exploring.

As a result of this work, more potential future scenarios reducing the initial investment cost by working with fewer AMRs and increased velocity profiles, decreased numbers of lifts, etc. in the developed flexible system can be explored. More experiments with different warehouse rack designs and decision-making rules could be considered and analysed to improve the system's performance.

6. Conclusion

This paper studies deadlock and collision prevention algorithms for flexible travel of automated mobile robots (AMRs) in automated warehousing systems. The main motivation for this work is to explore the existence of efficient travel policies for AMRs, by allowing them to travel flexibly throughout the storage area. By the flexible travel of AMRs, deadlock and collision control policies would be required contributing not only on the performance of the system but also on the initial investment cost of the system by decreasing the number of AMRs as well as buffer locations in the system. To explore the efficient deadlock and collision prevention policy, we utilise an agent-based simulation modelling approach. We treat the AMRs as intelligent agents, where they become autonomous dynamic objects in the system taking autonomous decisions through their goals. They interact with their environment and evaluate real-time information from its environment in selecting the best transaction request from its queue. The AMR agents are also able to define their travel routes through their destination addresses without having any deadlock or collision. We compare the developed flexible system's performance with a dedicated system design under different warehouse designs. The results are promising from the considered performance metrics: F_{avg}, F_{max}, and waiting time of transactions. Up to 39% improvement could be realised by the proposed flexible system design.

The limitation of the work is that the AMR agents are not able to do last minute decision changes while travelling between two decision points. Those limitations can be relaxed in future studies by applying more dynamic rules and last-minute decision changes.

In future works, more deadlock and collision prevention policies could be explored by also allowing AMRs to make last minute decisions such as, it might leave a load and change a pick-up decision address when it is on its route. Also, it may exchange the loads with other AMRs if it is more advantageous, etc. More warehouse layout designs can also be experimented to explore better performance results.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

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

Data availability statement

The data that support the findings of this study are available from the corresponding author, B.Y.E., upon reasonable request.

References

Chen, X., J. Ling, S. Wang, Y. Yang, L. Luo, and Y. Yan. 2021. "Ship Detection from Coastal Surveillance Videos via an Ensemble Canny-Gaussian-Morphology Framework." *Journal of Navigation* 74: 1252–1266.

Cossentino, M., C. Lodato, S. Lopes, and P. Ribino. 2011. "Multi Agent Simulation for Decision Making in Warehouse Management." In *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 611–618. Szczecin, Poland.

Draganjac, I., D. Miklić, Z. Kovačić, G. Vasiljević, and S. Bogdan. 2016. "Decentralized Control of Multi-AGV Systems in Autonomous Warehousing Applications." *IEEE Transactions on Automation Science and Engineering* 13 (4): 1433–1447.

Eder, M. 2019. "An Analytical Approach for a Performance Calculation of Shuttle-Based Storage and Retrieval Systems." *Production & Manufacturing Research* 7 (1): 255–270.

Ekren, B. Y. 2017. "Graph-Based Solution for Performance Evaluation of Shuttle-Based Storage and Retrieval System." *International Journal of Production Research* 55 (21): 6516–6526. https://doi.org/10.1080/00207543.2016.1203076.

Ekren, B. Y. 2020a. "A Simulation-Based Experimental Design for SBS/RS Warehouse Design by Considering Energy Related Performance Metrics." *Simulation Modelling Practice and Theory* 98: 101991.

Ekren, B. Y. 2020b. "A Multi-Objective Optimisation Study for the Design of an AVS/RS Warehouse." *International Journal of Production Research* 59: 1107–1126.

Ekren, B. Y., and B. Arslan. 2022. "A Reinforcement Learning Approach for Transaction Scheduling in a Shuttle-Based Storage and Retrieval System." *International Transactions in Operational Research*, 1–22. https://doi.org/10.1111/itor.13135.

Eroglu, E., and B. Yetkin Ekren. 2020. "An Agent-Based Simulation Model for Deadlock Prevention in an Aisle-to-Aisle SBS-RS." In *Virtual Global Joint Conference on Industrial Engineering and Its Application Areas*, GJCIE 2020, August, 135–146.

Fragapane, G., D. Ivanov, M. Peron, F. Sgarbossa, and J. O. Strandhagen. 2022. "Increasing Flexibility and Productivity in Industry 4.0 Production Networks with Autonomous Mobile Robots and Smart Intralogistics." *Annals of Operations Research* 308: 125–143. https://doi.org/10.1007/s10479-020-03526-7.

Ha, Y., and J. Chae. 2018. "Free Balancing for a Shuttle-Based Storage and Retrieval System." *Simulation Modelling Practice and Theory* 82: 12–31.

Ha, Y., and J. Chae. 2019. "A Decision Model to Determine the Number of Shuttles in a Tier-to-Tier SBS/RS." *International Journal of Production Research* 57 (4): 963–984.

Hsueh, C. F. 2010. "A Simulation Study of a Bi-Directional Load-Exchangeable Automated Guided Vehicle System." *Computers & Industrial Engineering* 58 (4): 594–601.

Knowledge Sourcing Intelligence. 2020. "Material Handling Market Size, Share, Opportunities, COVID 19 Impact, and Trends by Product (Automated Guided Vehicles, Automated Storage and Retrieval Systems, Trucks and Lifts, Cranes, Conveyor System, Hoists, Others), by Application (Warehousing and Distribution, Airport Baggage Handling, Bulk Handling, Postal and Parcel Delivery, Assembly, Packaging), by Industry Vertical (Manufacturing, Healthcare, Chemical, Paper, Food and Beverage, Warehousing, Others) and by Geography – Forecasts from 2020 To 2025." https://www.knowledge-sourcing.com/report/material-handling-market?gclid=Cj0KCQiA1pyCBhCtARIsAHaY_5dyf2o-SdBwDesZBh4m65EFTEHRVJlpy1vm4OHp067FyUGDNHHPOMsaAoJeEALw_wcB.

Krnjak, A., I. Draganjac, S. Bogdan, T. Petrović, D. Miklić, and Z. Kovačić. 2015. "Decentralized Control of Free Ranging AGVs in Warehouse Environments." In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2034–2041. Seattle, WA, USA.

Kucukyasar, M., and B. Y. Ekren. 2020a. "An Experimental Design Study to Identify Significant Factors Affecting Tier-to-Tier SBS-RS Performance." In *Virtual Global Joint Conference on Industrial Engineering and Its Application Areas*, GJCIE 2020, August, 191–201.

Kucukyasar, M., and B. Yetkin Ekren. 2020b. "Two Scheduling Rule Comparisons for Operations of Shuttles in Tier-to-Tier SBS-RS." In *Virtual Global Joint Conference on Industrial Engineering and Its Application Areas*, GJCIE 2020, August, 91–100.

Küçükyaşar, M., B. Y. Ekren, and T. Lerher. 2021a. "Cost and Performance Comparison for Tier-Captive and Tier-to-Tier SBS/RS Warehouse Configurations." *International Transactions in Operational Research* 28 (4): 1847–1863.

Kucukyasar, M., B. Y. Ekren, and T. Lerher. 2021b. "Energy Efficient Automated Warehouse Design." In *Solving Urban Infrastructure Problems Using Smart City Technologies Handbook on Planning, Design, Development, and Regulation*, edited by John R. Vacca, 269–292.

Lerher, T. 2018. "Aisle Changing Shuttle Carriers in Autonomous Vehicle Storage and Retrieval Systems." *International Journal of Production Research* 56 (11): 3859–3879.

Lerher, T., B. Ekren, G. Dukic, and B. Rosi. 2015. "Travel Time Model for Shuttle-Based Storage and Retrieval Systems." *The International Journal of Advanced Manufacturing Technology* 78: 1705–1725. https://doi.org/10.1007/s00170-014-6726-2.

Lerher, T., B. Y. Ekren, Z. Sari, and B. Rosi. 2015. "Simulation Analysis of Shuttle Based Storage and Retrieval Systems." *International Journal of Simulation Modelling* 14 (1): 48–59. https://doi.org/10.2507/IJSIMM14(1)5.281.

Lienert, T., and J. Fottner. 2017. "No More Deadlocks-Applying the Time Window Routing Method to Shuttle Systems." In *Proceedings of the 31st European Conference on Modelling and Simulation (ECMS)*, Budapest, Hungary, 169–175.

Lienert, T., F. Wenzler, and J. Fottner. 2020. "Simulation-Based Evaluation of Reservation Mechanisms for the Time Window Routing Method." *Anwendungen und Konzepte der Wirtschaftsinformatik* 12: 7.

LogisticsIQ. 2022. "Warehouse Automation Market." Accessed October 28 2022. https://www.thelogisticsiq.com/research/warehouse-automation-market/

Marchet, G., M. Melacini, S. Perotti, and E. Tappia. 2013. "Development of a Framework for the Design of Autonomous Vehicle Storage and Retrieval Systems." *International Journal of Production Research* 51 (14): 4365–4387.

Ning, Z., L. Lei, Z. Saipeng, and G. Lodewijks. 2016. "An Efficient Simulation Model for Rack Design in Multi-Elevator Shuttle-Based Storage and Retrieval System." *Simulation Modelling Practice and Theory* 67: 100–116.

Research and Markets. 2021a. "AGV (Automated Guided Vehicles) and AMR (Autonomous Mobile Robots) Market for Logistics, Manufacturing, Healthcare, Disinfection, Retail, Security, Agriculture, Hospital, Indoor and Outdoor Delivery, Cleaning, Tele-operation - Forecast to 2026." https://www.researchandmarkets.com/reports/5232383/agv-automated-guided-vehicles-and-aAV#src-pos-1.

Research and Markets. 2021b. "Warehouse Automation Market with Post-Pandemic (COVID-19) Impact by Technology (AGV/AAV, ASRS, Conveyors, Sortation, Order Picking, AIDC, Palletizing and WMS), by Industry (E-Commerce, Grocery, Apparel, F&B, Pharma), by Geography – Forecast to 2026." 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%2fAAV%2c(ASRS%2c(Conveyors%2c(Sortation%2c(Order(Picking%2c(AIDC%2c(Palletizing(and(WMS&utm_exec=chdo54prd.

Rhazzaf, M., and T. Masrour. 2021. "Smart Autonomous Vehicles in High Dimensional Warehouses Using Deep Reinforcement Learning Approach." *Engineering Letters* 29: 1.

Roy, D., A. Krishnamurthy, S. S. Heragu, and C. J. Malmborg. 2014. "Blocking Effects in Warehouse Systems with Autonomous Vehicles." *IEEE Transactions on Automation Science and Engineering* 11 (2): 439–451.

Roy, D., A. Krishnamurthy, S. S. Heragu, and C. J. Malmborg. 2016. "A Simulation Framework for Studying Blocking Effects in Warehouse Systems with Autonomous Vehicles." *European Journal of Industrial Engineering* 10 (1): 51–80.

Sgarbossa, F., A. Romsdal, F. H. Johannson, and T. Krogen. 2020. "Robot Picker Solution in Order Picking Systems: An Ergo-Zoning Approach." *IFAC-PapersOnLine* 53 (2): 10597–10602.

Turhanlar, E. E., B. Y. Ekren, and T. Lerher. 2021. "Aisle-to-Aisle Design for SBS-RS Under Smart Deadlock Control Policies." In *Proceedings of the 11th Annual International Conference on Industrial Engineering and Operations Management Singapore*.

Vivaldini, K. C., J. P. Galdames, T. S. Bueno, R. C. Araújo, R. M. Sobral, M. Becker, and G. A. Caurin. 2010. "Robotic Forklifts for Intelligent Warehouses: Routing, Path Planning, and Auto-Localization." In *2010 IEEE International Conference on Industrial Technology*: *ICIT 2010*; Viña del Mar, Chile, 14–17 March 2010, 1463–1468.

Winkelhaus, S., and E. H. Grosse. 2022. "Chapter 3 – Smart Warehouses – a Sociotechnical Perspective." In *The Digital Supply Chain*, edited by Bart L. MacCarthy and Dmitry Ivanov, 47–60. Elsevier. doi:10.1016/B978-0-323-91614-1.00003-4. ISBN 9780323916141.

Wu, Y., C. Zhou, W. Ma, and X. T. Kong. 2020. "Modelling and Design for a Shuttle-Based Storage and Retrieval System." *International Journal of Production Research* 58 (16): 4808–4828.

Yau, K.-L. A., S. Peng, J. Qadir, Y.-C. Low, and M. H. Ling. 2020. "Towards Smart Port Infrastructures: Enhancing Port Activities Using Information and Communications Technology." *IEEE Access* 8: 83387–83404. doi:10.1109/ACCESS.2020.2990961.

Zhou, Y., H. Hu, Y. Liu, and Z. Ding. 2017. "Collision and Deadlock Avoidance in Multirobot Systems: A Distributed Approach." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47 (7): 1712–1726.

Appendices

Appendix 1. Experimental design simulation results for FSD

Warehouse design		Vehicle scenario		Performance measures			
C	A	V	N	λ (monthly)	F_{avg}	F_{max}	W_R
1200	6	2; 3	3	225,590 ± 278	77.4 ± 0.3	1155 ± 102	58%
1200	6	2; 3	4	297,860 ± 312	68.3 ± 0.5	772 ± 31	50%
1200	6	3; 3	3	264,820 ± 424	63.1 ± 0.2	940 ± 61	57%
1200	6	3; 3	4	345,940 ± 542	59.0 ± 0.3	991 ± 83	52%
1200	12	2; 3	3	328,420 ± 472	53.7 ± 0.3	876 ± 29	59%
1200	12	2; 3	4	428,820 ± 524	46.0 ± 0.4	691 ± 25	51%
1200	12	3; 3	3	365,370 ± 530	47.1 ± 0.2	780 ± 30	58%
1200	12	3; 3	4	480,170 ± 565	40.9 ± 0.4	609 ± 16	50%
3600	6	2; 3	3	97,896 ± 248	183.9 ± 2.0	2508 ± 140	59%
3600	6	2; 3	4	128,370 ± 219	174.6 ± 1.4	2320 ± 60	56%
3600	6	3; 3	3	124,690 ± 289	149.5 ± 1.2	2057 ± 60	61%
3600	6	3; 3	4	161,560 ± 355	136.0 ± 1.0	2127 ± 276	56%
3600	12	2; 3	3	173,040 ± 428	103.1 ± 0.4	1511 ± 115	59%
3600	12	2; 3	4	222,720 ± 375	85.2 ± 0.4	1145 ± 90	49%
3600	12	3; 3	3	207,520 ± 372	85.4 ± 0.3	1318 ± 84	59%
3600	12	3; 3	4	270,350 ± 441	72.4 ± 0.3	972 ± 78	50%

Appendix 2. Experimental design simulation results for DSD

Warehouse design		Vehicle scenario		Performance Measures			
C	A	V	N	λ (monthly)	F_{avg}	F_{max}	W_R
1200	6	2; 3	3	225,590 ± 278	80.0 ± 0.2	2385 ± 254	64%
1200	6	2; 3	4	298,260 ± 578	123.7 ± 1.4	7016 ± 853	79%
1200	6	3; 3	3	264,820 ± 424	55.9 ± 0.2	1468 ± 276	59%
1200	6	3; 3	4	345,940 ± 543	76.5 ± 0.5	4303 ± 1333	72%
1200	12	2; 3	3	328,290 ± 279	58.0 ± 0.6	1852 ± 290	66%
1200	12	2; 3	4	428,640 ± 279	50.4 ± 0.6	1354 ± 204	61%
1200	12	3; 3	3	365,280 ± 199	45.1 ± 0.3	1398 ± 322	62%
1200	12	3; 3	4	480,040 ± 219	40.2 ± 0.3	1007 ± 199	58%
3600	6	2; 3	3	97,897 ± 247	241.3 ± 1.3	7902 ± 2,317	71%
3600	6	2; 3	4	128,380 ± 217	285.8 ± 6.1	12,442 ± 4935	84%
3600	6	3; 3	3	124,690 ± 287	146.5 ± 0.5	4147 ± 736	65%
3600	6	3; 3	4	162,130 ± 451	197.7 ± 1.5	8598 ± 703	76%
3600	12	2; 3	3	172,833 ± 345	128.2 ± 2.2	3839 ± 1125	70%
3600	12	2; 3	4	222,514 ± 375	126.9 ± 2.4	3451 ± 425	68%
3600	12	3; 3	3	207,777 ± 403	82.9 ± 1.2	2289 ± 301	63%
3600	12	3; 3	4	270,560 ± 351	83.2 ± 1.0	2271 ± 461	63%