Proceedings of the 2022 Winter Simulation Conference
B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C.G. Corlu, L.H. Lee, E.P. Chew, T. Roeder, and P. Lendermann, eds.

ORDER RELEASE STRATEGIES FOR A COLLABORATIVE ORDER PICKING SYSTEM

Quang-Vinh Dang
Tugce Martagan
Ivo Adan

Department of Industrial Engineering and Innovation Sciences
Eindhoven University of Technology
Groene Loper 3
Eindhoven, 5612 AE, THE NETHERLANDS

Jan Kleinlugtenbeld

Vanderlande Industries B.V.
Vanderlandelaan 2
Veghel, 5466 RB, THE NETHERLANDS

ABSTRACT

A collaborative order picking system (COPS) enables human-robot collaboration by using order pickers for picking and autonomous mobile robots (AMRs) for transporting load carriers. Owing to the potential performance enhancement compared to a traditional manual order picking system, COPSs are gaining momentum in the retail warehousing sector. This paper proposes order release strategies based on priority and dispatching rules to achieve the best pick rate performance per AMR. A discrete event simulation model is developed to facilitate the evaluation of the proposed strategies. Their effectiveness is demonstrated with the use of real-world data from a case study warehouse. Our computational results show that a COPS using proposed strategies significantly improves the pick rate performance compared to the current practice.

1 INTRODUCTION

In today's constantly changing global supply chains, demands for products and services are ever-increasing. Both customers and companies have higher expectations in terms of the availability of products. Consequently, warehousing plays a more vital role in companies' supply chain networks. A major activity in warehouse operations is order picking (Bartholdi and Hackman 2019). Order picking, the process of retrieving items from their storage locations, has both important labor-intensive and cost-intensive aspects (Ho et al. 2008; de Koster et al. 2007). de Koster et al. (2007) report that over 80 % of all orders processed by warehouses worldwide are picked manually. Due to the flexibility and agility of human order pickers, manual order picking is still often preferred above full automation. However, traditional picker-to-parts order picking systems are relatively inefficient since order pickers only spend approximately 15 % of their time on picking. In comparison, around 50 % of the time is spent on traveling (Tompkins et al. 2003). As human order pickers are especially of added value during picking and not during other processes, alternatives employing automated equipment are introduced to optimize the system. Traditionally, automation in warehousing only included automated storage and retrieval systems, carousels, or shuttle systems (Azadeh et al. 2019). More recently, automated guided vehicles (AGV) are being used to automate transportation tasks, such as repetitive movement of racks in a parts-to-picker system, in highly structured and static warehouses (Azadeh et al. 2019). However, a more technologically sophisticated approach is needed to deploy such automation in collaboration with human order pickers in a picker-to-part system (Azadeh et al. 2019; Boysen et al. 2019). The recent design of a more dynamic vehicle, the autonomous mobile robot (AMR), has opened up opportunities for collaborative order picking systems (Meller et al. 2018).

Collaborative order picking systems (COPS) are picker-to-part systems enabling human-robot collaboration by using human order pickers for picking and AMRs to transport load carriers. This system enables

978-1-6654-7661-4/22/$31.00 ©2022 IEEE

order pickers to exclusively focus on picking cases, while AMRs take care of all the transportation activities. This system may significantly increase picker productivity by decreasing the picker traveling distance and handling time. Although AMRs allow for flexible deployment without significant changes to the existing warehouse infrastructure, especially for traditional manual order picking systems, employing AMRs in a warehousing environment creates new design and control challenges.

One of the difficulties faced when implementing such a human-robot system is efficiently controlling the collaborative order picking that consists of order batching and batch sequencing (releasing) decisions (Žulj, Salewski, Goeke, and Schneider 2022). The three types of human-robot collaborative systems, introduced by Azadeh et al. (2020), illustrate that different operational policies can be adopted since system behavior can be dynamically adjusted, making these systems very flexible. To show the flexibility of the system, Azadeh et al. (2020) provide a model to analyze the effect of dynamic zoning strategies in a human-robot collaborative picking environment, using queuing models and Markov Decision Processes. Ghelichi and Kilaru (2021) provide analytical models to analyze last-mile delivery and meet-in-aisle concepts using AMRs. Fragapane et al. (2021) provide a review and research agenda for planning and control of AMRs in intralogistics. Recently, Žulj et al. (2022) propose a two-stage heuristic to deal with order batching and sequencing on an AMR-assisted picking system. Nevertheless, their paper considers a deterministic and static system, with specific handover points where order pickers put picked items in AMRs, which differs from our COPS. Although human-robot collaborative order picking systems are gaining momentum, the literature in this domain is still very scarce. It is widely discussed that employing robot solutions in warehouse order picking offers multiple advantages. Nevertheless, there is still a lack of research on its applicability in traditional warehousing concepts. Hence, this paper considers a COPS in a retail warehouse domain, focusing on the batching and releasing decisions in the system.

In this paper, the following contributions are made: (i) we introduce various order release strategies that outperform the current practice; (ii) we develop a simulation model of a COPS for testing the proposed strategies; (iii) we provide managerial insights for practitioners based on results of a real-world case study; (iv) we provide data for the real-world case study used for the experiments in this paper. This research is conducted in close collaboration with our industry partner, Vanderlande Industries B.V., a company located in the Netherlands. Vanderlande is the global market leader for end-to-end value-added logistic process automation. The remainder of the paper is organized as follows. Section 2 provides a problem description. Next, order release strategies are presented in Section 3, while a simulation model of a COPS is developed in Section 4. Experimental results are given in Section 5. Finally, conclusions and interesting future research directions are discussed in Section 6.

2 PROBLEM DESCRIPTION

In this section, we describe the key components of a COPS, the control of order pickers, and the collaborative order picking problem. The COPS employs AMR to support order pickers by taking over all the transportation tasks, enabling pickers to focus on picking. An illustration of the COPS is provided in Figure 1. Each AMR in the system follows a specific routine. This routine starts when an AMR does not have a task to execute, so idle. The idle state is interrupted by the Warehouse Management System (WMS), assigning a collection of order lines, or pickrun, to an AMR. After receiving a pickrun, the AMR collects empty load carriers to start the picking operation. Next, it moves towards the first pick location. When an AMR arrives at a pick location, it waits to get served by an order picker. Once it is served, the AMR moves towards the following pick location. This process continues until all items of a pickrun are collected. Once a pickrun is complete, the AMR drives from the last pick location towards the drop-off location to drop off the full load carriers. From there, it will move towards its starting point, ending its routine.

As shown in Figure 1, order pickers move freely in the picking area during the picking process, picking items for AMRs waiting at specific pick locations. Order pickers follow a free-floating policy, which means that they are not explicitly committed to a certain pickrun or AMR but are guided to available pick locations by a central controller. It also means that order pickers are not committed to any zone, or there is no fixed

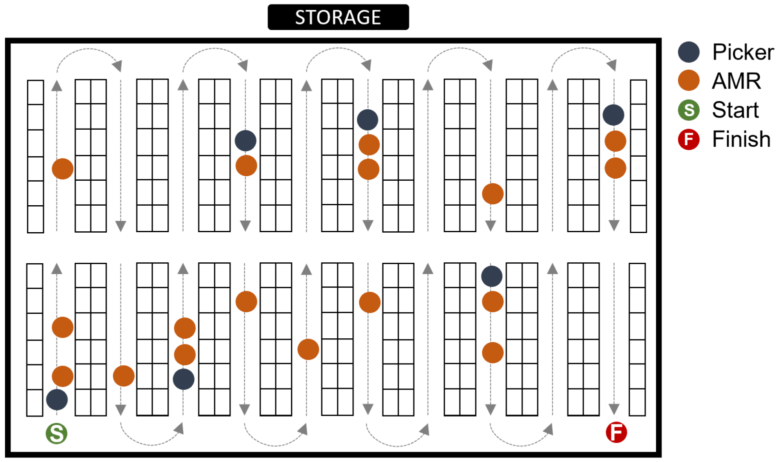


Figure 1: COPS illustration.

number of pickers per zone, where a zone is defined as an area that is a combination of a number of aisles (de Koster et al. 2007). If multiple AMRs are nearby waiting to be served, an order picker can serve them simultaneously. The principle of picking multiple AMRs, or a 'wave' of AMRs, at once gives the COPS the potential to improve the pick rate performance (number of order lines picked per hour).

The problem is how to create waves of AMRs to achieve a high pick rate. The creation of waves of AMRs is influenced by the composition of pickruns (batching decision) and the sequence of assigning these pickruns to AMRs (sequencing or releasing decision). For instance, if a couple of pickruns have similar pick locations in a particular zone, they could be released to create a cluster with a high density of potential pick locations. This enables an order picker to pick a lot of products without having to spend a lot of time traveling, which increases the performance of the order picker and thus the overall pick rate performance of the system. The combination of the batching and releasing decisions in the COPS is referred to as the collaborative order picking problem (COPP). The former decision determines the composition, or a 'batch', of picklists in a pickrun, where each picklist consists of a predetermined set of products that needs to be picked to fill a single load carrier. The latter determines which pickrun, from the available pickruns, to release to an idle AMR. The combinations of different batching and releasing rules result in alternative order release strategies (ORSs).

From a business perspective, applying ORS to COPS is of added value if a higher pick rate performance can be achieved with fewer resources. The investment costs of a COPS are roughly linear to the required number of AMRs. Therefore, we aim to develop an ORS that provides the best pick rate performance per AMR. In addition, the impact of an ORS on a COPS can be assessed using other performance measures, i.e., the time AMR and pickers spend on each activity, such as traveling, picking, and waiting, can be used as performance indicators. A decrease in traveling and waiting time is needed to increase the pick rate performance of a COPS while increasing the time spent on picking. If order pickers devote more time to picking, the probability of picking more order lines increases, but encountering disturbance (capturing all breaks and disruptions) also increases. The disturbance may increase the waiting time to be served of AMRs, thus the likelihood of AMR congestion, which in turn affects the waves of AMRs and the pick rate performance per AMR ultimately. All these trade-offs need to be considered when designing ORSs. In the next section, we describe various batching and releasing approaches for ORSs.

3 ORDER RELEASE STRATEGIES

This section presents different approaches for the batching and releasing decisions. While batching methods are described in Section 3.1, releasing methods are discussed in Section 3.4, followed by their combinations.

3.1 Batching Methods

The batching decision is responsible for creating pickruns from a set of picklists. A picklist consists of order lines with specific pick locations and several cases that need to be picked at these locations. Picklists are created by the WMS and will not be subject to change. The capacity of each AMR is restricted to two load carriers in our paper. In other words, each pickrun can consist of a combination of two picklists, each one of them belonging to a single load carrier. For batching picklists, we present two methods: regular heuristic (RH) and savings heuristic (SH). The RH applies the current way of creating pickruns in the case study warehouse. On the other hand, the SH adapts the concept of the Clarke & Wright Savings Algorithm (CWSA) (Clarke and Wright 1964) to create batches that 'save' travel distance. These methods will be discussed in Sections 3.2 and 3.3, respectively.

3.2 Regular Heuristic

Our case study warehouse employs a single discrete-order-picking policy. It means order pickers pick per customer order, and all picklists of a pickrun are picked sequentially. The same concept is applied in the regular batching approach. First, all available picklists will be grouped based on customer orders. Next, each group of picklists will be ordered based on their pick locations. The first two picklists are then batched in a pickrun. This continues until all picklists of a group have been batched in pickruns. Afterwards, the same procedure is applied to the next group until all groups are considered. If a customer order contains an uneven number of picklists, there will be a pickrun consisting of only one picklist.

3.3 Savings Heuristic

For the SH batching method, the CWSA is adapted to fit the batching of two picklists in a pickrun. The savings of a pair is calculated by adding the total distance of the first picklist to the total distance of the second picklist and then subtracting the total distance of the combination of the two picklists. A positive saving indicates that combining the two picklists is favorable in terms of travel distance. In contrast, a negative saving suggests that picking both picklists separately would be more advantageous, assuming a fixed starting location. The procedure of the SH is presented in Algorithm 1.

Algorithm 1: Savings heuristic.

Result: Maximized distance savings for solution S
Set initial solution S_0 as list of individual picklists
for $(q \wedge r) \in S_0$ *and* $(q \neq r)$ **do**
    Calculate d_q, d_r, and d_{qr}
end
Create empty savings list SL
for $(q \wedge r) \in S_0$ *and* $(q \neq r)$ **do**
    $s_{qr} = d_q + d_r - d_{qr}$
    Append s_{qr} to SL
end
Sort SL in descending order of s_{qr}
Create empty solution S
while $SL \neq \emptyset$ **do**
    for $s_{qr} \in SL$ **do**
        if $(q \notin S) \wedge (r \notin S)$ **then**
            Combine $q \wedge r$ and add to S
            Eliminate s_{qr} from SL
        else
            Eliminate s_{qr} from SL
        end
    end
end
return solution S

First, an initial solution (S_0) is initiated as the set of picklists (line 1). Then, the total distance of every single picklist and every pair of two picklists in the initial solution is calculated (lines 2–4). The total distance (d_q) of a single picklist (q) is calculated by iterating over the pick locations in the list and summing the distances between the pick locations. The two individual picklists (q and r) are merged into a single list (qr) to calculate the combined distance. A distance model calculating these distances is designed based on the dimensions of the case study warehouse and the s-shape routing policy (Žunić et al. 2018).

Afterwards, the SH computes a list of savings (*SL*) of all possible combinations (lines 6–9). The saving of a combination of two picklists is determined by subtracting the combined total distance from the sum of the individual total distances. Then, the SH constructs solution S iteratively (lines 12–21). It checks each pair in the savings list whether it is feasible to combine. The feasibility is based on whether either of the two picklists has already been batched or not (line 14). Each time a savings pair has been combined or discarded, the element is removed from the savings list (lines 16 & 18) until the savings list is empty.

3.4 Releasing Methods

The releasing decision is in charge of assigning a pickrun to an idle AMR. The goal is to release pickruns to create a high pick density of released work, thus enabling the formation of waves of AMRs. In addition to the current way of working, this paper proposes a release rule based on zone information of the available pickruns. The release rules are explained in detail in the following sections.

3.5 Random Release Rule

This first release rule can be seen as a benchmark based on the current working method in the case study warehouse. It allows for comparing the current way of working against one of the designed release rules. This rule uses a random draw from the list of available pickruns to release a pickrun to an idle AMR. We note that the complete warehouse is considered a single zone for this rule. In this way, the workload can be spread equally across the warehouse. As each customer order generally contains SKU from across the whole warehouse, all pickruns together should cover approximately the complete warehouse. By giving each pickrun an equal probability of being chosen, there is a high probability of achieving an equal spread throughout the warehouse.

3.6 Zone-Based Release Rule

The basis of the second release rule is illustrated in Figure 2. This release rule splits up the warehouse into several zones to control the workload in each of them (de Koster et al. 2007). Releasing pickruns per zone creates 'clusters' of potential pick locations, which is favorable for creating waves of AMRs, thus achieving a high pick rate performance.

The rule is based on pickrun- and zone-specific information. For each pickrun in the available pickruns, a zone score is determined for each zone. The zone score is calculated by the aisle number of each pick location of a pickrun. If an aisle number aligns with a zone, the zone score is increased by one. When performing this calculation for a complete pickrun, it can be decided which zone the pickrun aligns with by taking the maximum zone score. By aligning all pickruns of the available set with their respective zones, a zone-based sequence is created. Next, the pickruns in the zone need to be sequenced, for which two types of rules have been applied. These rules will be discussed in the following paragraphs. The complete flow chart of the rule can be created by inserting a specific type in the blue process in Figure 2.

The first type of rule provides an ordering based on the lowest pick location number of each pickrun. The sequence of pickruns is ordered based on their zones, from a low aisle number to a high aisle number. Then, the pickruns per zone are ordered on the pick location number of their first pick location. Overall, this creates a sequence from the left-hand side of the warehouse to the right-hand side of the warehouse. It aims to create a picking 'wave' across the zones, moving gradually from left to right, allowing order pickers to move along with the released work. This rule will be referred to as the zone-based ordering rule.

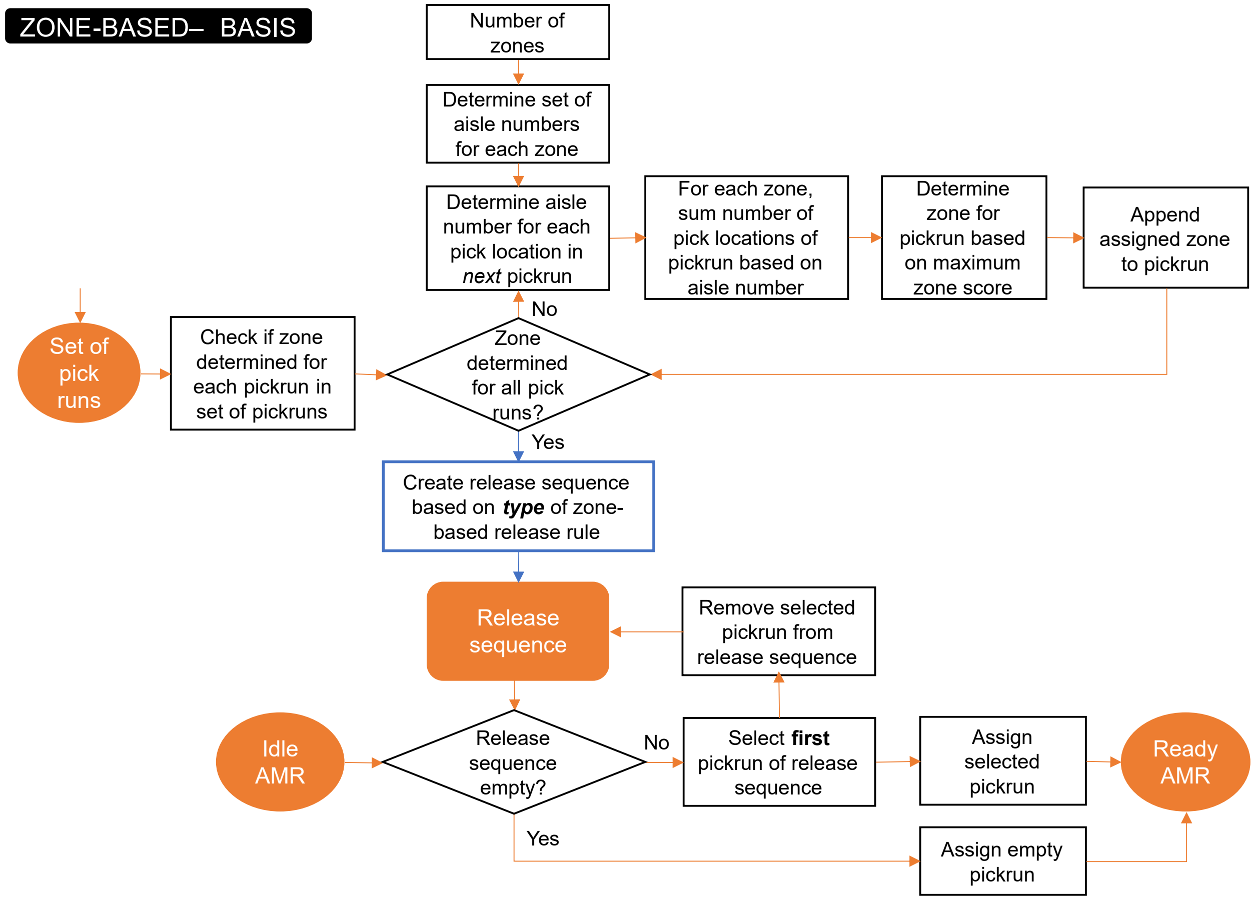


Figure 2: Zone-based release rule.

In the second variant, the sequence is first determined by the zone a pickrun is assigned to (from left to right) and then by randomizing the pickruns belonging to a particular zone. It still clusters work by zones but spreads the workload within zones. While zone-based ordering aims to create a high pick density in a specific area, using random sequencing within zones potentially decreases the effect of congestion. This variant will be referred to as the zone-based random rule.

Lastly, each combination of batching and releasing methods creates an ORS. With the two batching and three releasing methods, there are six combinations in total. We denote the codes of the batching methods by R (for regular heuristic) and S (for savings heuristic). Similarly, the codes of the releasing methods are R (for random release), ZO (for zone-based ordering), and ZR (zone-based random). Then, an ORS will be referred to by a code for the batching and releasing method, e.g., the current practice with *regular* batching and *random* release will be referred to as ORS R-R.

4 SIMULATION MODEL

This section presents a discrete-event simulation model for a COPS in which the introduced batching and releasing methods are implemented as an input function. Simulation is chosen because it allows capturing system uncertainties in our problem, for example, the AMR congestion and disturbance of order pickers. The simulation model is coded in Python 3.8.11 (Van Rossum and Drake 2009). The SimPy package has been used for the specific implementation of the DES simulation engine (Team SimPy 2020).

The AMR process is responsible for the behavior of an individual AMR, while the picker process controls the behavior of an individual picker, including traveling to pick locations, disturbance, and picking.

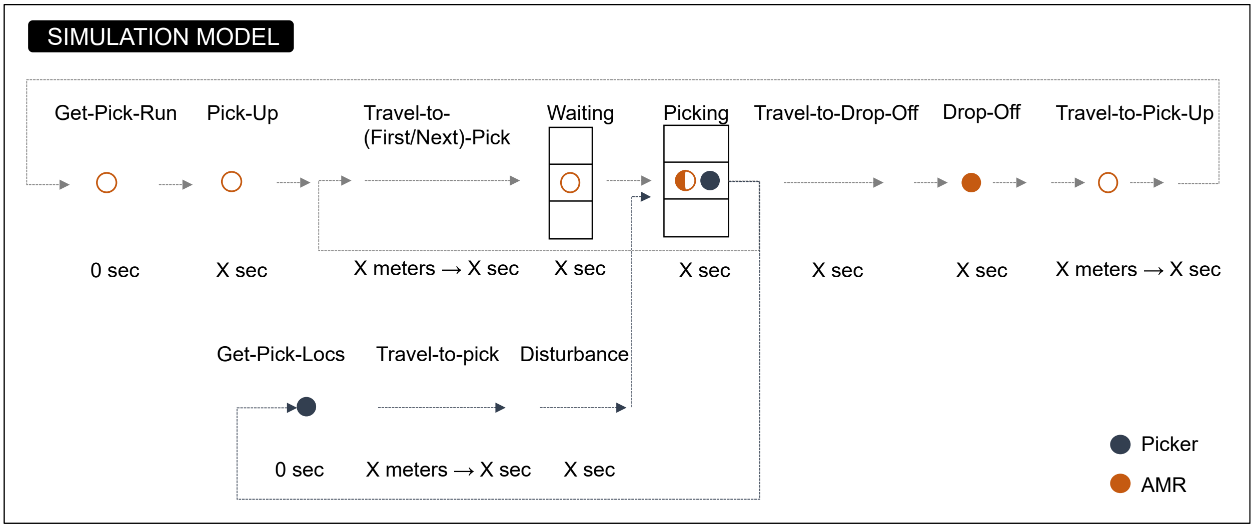


Figure 3: Overview simulation model.

The picker optimizer process allocates pickers to pick locations by checking whether there is any AMR available for picking from the current location of the picker until the end of the current half-aisle and on the next half-aisle in the direction of work (upstream of the driving/walking direction). This continues until the pickruns of two or more AMRs are assigned to the picker. The picker optimizer process is triggered each time a new event occurs in the SimPy Environment since, after each event, it could be needed to send a picker to a new pick location.

An interaction between an AMR and picker process occurs whenever a picker serves an AMR at a specific pick location. A ‘rack’ is used to model the interaction at such a location. A SimPy *FilterStore* represents a ‘rack’ and forms a queue for the *FilterStore.Put* and *FilterStore.Get* actions of requests. For each type of interaction of processes, a separate list of racks is initialized. Every rack in the list represents a pick location, so the length of the list is determined by the number of locations in the warehouse. Next to the AMR-picker interaction, there are two other types of interactions. The interaction between the picker- and picker optimizer process triggers whenever a picker does not have a task to execute. The interaction between AMR processes is triggered whenever AMRs overtake each other (see Figure 4). The overtaking times can be determined by using the AMR speed and the distance an AMR needs to go ‘around’ another AMR in an aisle. This distance is calculated based on the dimensions of the warehouse (see the distance model). For an interaction to occur, one process must *put* a request in a rack, and the other must *get* this specific request out of the rack. As there are multiple AMR and picker processes, each process has its identifier. Every request in a rack inherits the identifier from its sender to make sure that the receiver gets the valid request from the rack. This way, multiple processes may put a request in the same rack and can be served by various other processes.

For the simulator to start, several elements need to be initialized. First, all the main input parameters need to be instantiated, including the set of pickruns that needs to be processed, which is the output of the batching decision (see Section 3.1). Second, the AMR and picker processes are initialized for a defined number of AMR and pickers, respectively. Third, each AMR process is assigned a random pickrun with a random amount of remaining pick locations. These pickruns are randomly drawn from the set of pickruns. Each picker process is given a random start location in the warehouse. This initialization procedure ensures that the simulation model is not empty at the start and thus does not use the actual pickruns during its warm-up period. Fourth, the racking for each type of process interaction is constructed. Finally, the picker optimizer process is initiated. After all these elements are complete, the simulation model can start.

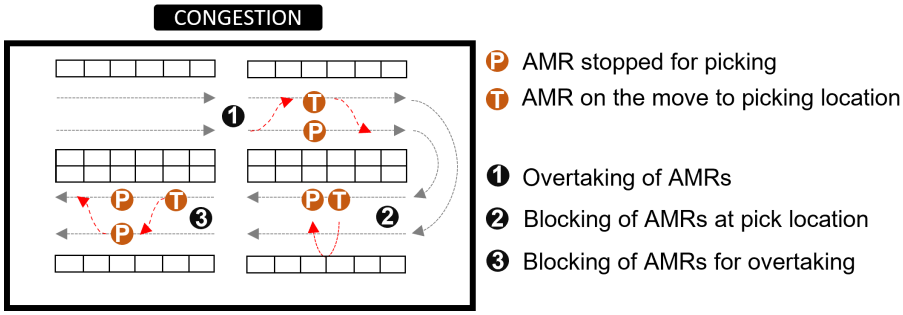


Figure 4: AMR congestion scenarios.

As the simulation model needs to capture the real-world aspects of a COPS as closely as possible, the model should be detailed while maintaining an acceptable run time. To achieve this objective, the model is designed under several assumptions:

- A data sample of the order picking data is used as model input. This data is assumed to be complete and available at the beginning of the operation. Hence, no continuous arrivals will occur. Also, the allocations of picked items are given in the data, and the item reallocation is not considered.
- Disturbance of pickers is modeled as a timeout event initiated after a number of order lines are picked. This event is assumed to capture all breaks, case failures, re-stacking, and other disruptions.
- The speed of AMRs and pickers are constants based on average real-life speeds. The picking time is assumed to be constant without loss of generality because the picking action in the COPS limits itself to only taking an item out of a storage location and putting it on an AMR.
- Pick-up and drop-off locations for AMRs are fixed at a specific pick location to enable exact distance calculations. It is assumed that individual AMRs do not travel to shipping docks but to a dedicated drop-off location.
- In the pick-up and drop-off state, it is assumed that AMRs do not encounter congestion.

5 RESULTS AND DISCUSSION

We first provide an analysis of the current way of working using an order picking data sample in Section 5.1. We then study and compare the performance of various ORSs in Section 5.2. To provide realistic insights on the design of ORS for a COPS, real-world order picking data will be used from a case study warehouse. The realistic dimensions of the case study warehouse can also be found in the distance model.

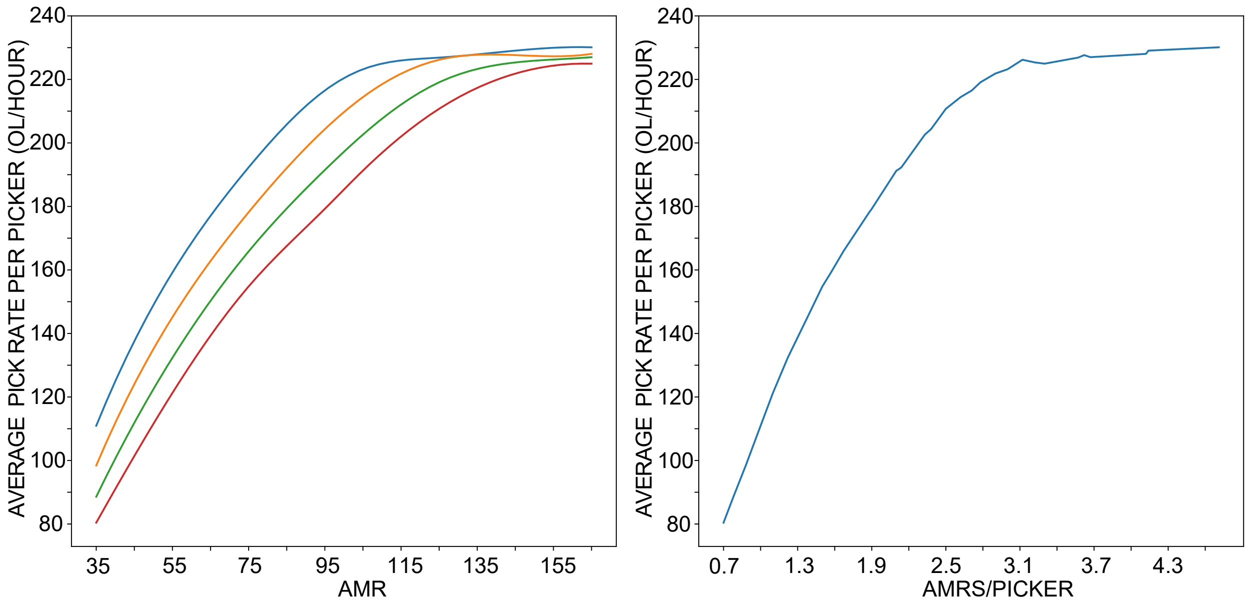
5.1 Analysis of Current Practice

An overview of the analysis of the current practice (ORS R-R) is presented in Table 1. The parameter values in Table 1 are derived from discussions with experts from Vanderlande. For the picker disturbance, after a normally distributed number of order lines with $\mathcal{N}(50,5)$ is picked, a normally distributed timeout with $\mathcal{N}(45,3)$ is triggered. In addition, at a specific pick location, picking the first product takes 6 seconds, and picking each remaining product takes 3 seconds. Also, the warehouse is divided into 4 zones. In this analysis, the impact of the number of AMRs will be first tested against the number of pickers. The result of this analysis is shown in Figure 5.

In general, adding more AMRs leads to higher system performance until the maximum utilization of the pickers is reached. At some point, the pickers will not pick more items, and their pick rate performance stabilizes. In this case, adding more AMRs does not have an effect anymore. From the perspective of manual picking, adding more pickers would seem a logical step to increase the pick rate performance. However, as shown in Figure 5(a), for a COPS, a high number of pickers requires a large number of AMRs to achieve a high pick rate performance. This is caused by the fact that the number of AMRs determines

Table 1: Analysis scenarios.

Scenario	Parameter	Value Range
1	Number of AMRs	[35, 55, ..., 145, 165]
	Number of Pickers	[35, 40, 45, 50]
2	Picker speed (m/s)	[0.4, 0.6, ..., 1.4, 1.6]
	Number of AMRs	[55, 95, 125, 165]
3	AMRs speed (m/s)	[0.4, 0.6, ..., 1.4, 1.6]
	Number of Pickers	[15, 25, 35, 45]

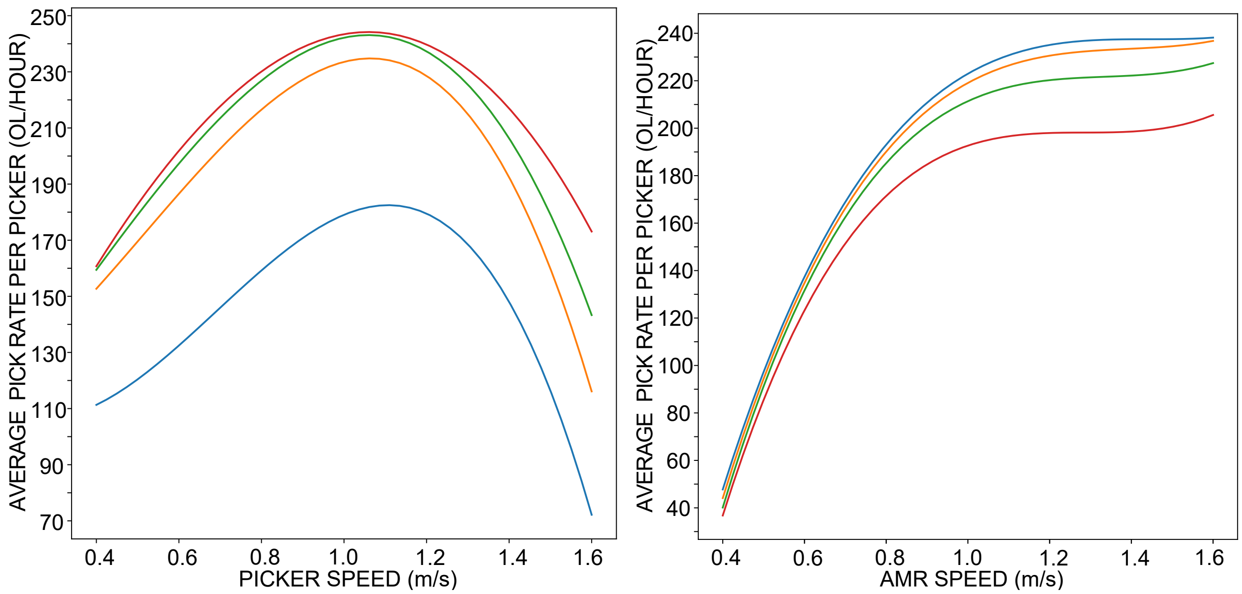


(a) (b)

Figure 5: (a) Average pick rate per AMR: 35 pickers (—), 40 pickers (—), 45 pickers (—), 50 pickers (—); (b) Average pick rate per ratio AMRs/picker.

the number of potential pick locations. With more pickers in the system, potential pick locations should increase to achieve similar performance. Hence, more AMRs are required. When looking at the curve for 35 pickers, the peak of the curve is reached between 95 and 115 AMRs. Looking at the relative increase in performance, the increase from 95 to 100 AMRs adds 5 OL/hr, while the increase from 100 to 115 AMRs also adds 5 OL/hr. Therefore, when considering the return on investment, using 100 AMRs for 35 pickers seems to be reasonable. This suggestion is confirmed by Figure 5(b), where the ratio of AMR per picker is plotted against the pick rate performance. In this plot, it can be seen that the curve reached its maximum around 2.9 - 3.1 AMRs/picker. Increasing this ratio further provides minimal additional returns in terms of performance.

The second analysis focuses on picker speed and AMR speed. The results of this analysis are presented in Figures 6(a) and 6(b), respectively. In general, with a higher picker speed, the picker spends less time traveling, and thus has more time left for picking, assuming that the pick time stays equal. As a consequence, the pick rate performance would increase. However, in a COPS, the number of AMRs is a limiting factor. More AMRs in the system lead to more potential pick locations. In combination with a higher picker speed, this leads to increased performance. The pick rate performance declines when the speed of the



(a) (b)

Figure 6: (a) Average pick rate per picker speed (with 35 pickers and 1.2 m/s AMR speed): 55 AMRs (—), 95 AMRs (—), 125 AMRs (—), 165 AMRs (—); (b) Average pick rate per AMR speed (with 100 AMRs and 0.8 m/s picker speed): 15 pickers (—), 25 pickers (—), 35 pickers (—), 45 pickers (—).

pickers grows above the speed of the AMR. This is because the picker optimizer reallocates a picker to another location once it detects the picker and AMR will not reach their pick location simultaneously. This logic is based on the constraint that a picker will never wait for an AMR to arrive. Reallocation of pickers leads to additional travel time, which reduces the average pick rate performance. A potential solution for this specific problem would be either increasing the speed of the AMR or reducing the number of pickers. However, it is questionable whether a picker speed above 1.2 m/s is representative in a real-life case. Following our industry partner, it seems more reasonable to assume a picker speed of around 0.8 m/s. Figures 6(a) and 6(b) shows that the reachable and reasonable option for 100 AMRs is about 1.0 m/s picker speed, 35 pickers, and 1.2 m/s AMR speed.

5.2 Comparison of Different ORS

This section studies the performance of the ORSs presented in Section 3. The input parameters of the simulation model are set to the reasonable values determined in Section 5.1. In addition, ORS R-R is considered the current practice, thus used as a benchmark. To assess whether an ORS is significantly different from the benchmark, a statistical test, i.e., Mann-Whitney U test (Mann and Whitney 1947), will be used. For this assessment, the average pick rate per picker is used as a performance indicator.

Table 2 shows that the performance in the number of order lines increases when using an ORS different from the current practice. Our statistical tests result in the p-values smaller than 0.05, and hence they are statistically different compared to ORS R-R. Table 2 also reveals that the zone-based rules (under the same batching method) improve the average pick rate by approximately 3–4 %. With the same release rule, the SH considerably improves the performance by around 15–16 %. Especially, their combinations (S-ZO and S-ZR) gain the most with about 20 % increase in the average pick rate. This improvement also implies the possibility of decreasing the number of AMRs with the proposed saving and zone-based heuristics while still achieving the same pick rate performance as in the current practice.

Table 2: Results of various ORSs.

	Measure	R-R		R-ZO		R-ZR		S-R		S-ZO		S-ZR	
		μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
	OL/hr	219.9	9.6	227.3	9.1	224.3	8.7	253.6	10.7	264.8	8.6	261.8	8.1
	% improvement vs. R-R	-		3.4		2.0		15.3		20.4		19.1	
AMR	Pick-Up	10.9	1.1	11.1	1.4	11.0	1.2	11.8	1.1	12.2	1.5	12.1	1.2
	Travel to pick	14.4	1.8	14.7	2.6	14.6	1.8	10.5	1.4	10.9	1.8	10.8	1.3
	Waiting	43.9	3.5	39.3	8.1	42.1	5.5	43.6	3.9	37.9	7.6	41.4	5.1
	Picking	14.3	1.5	14.6	2.3	14.4	1.9	16.5	1.8	17.0	2.4	16.9	2.1
	Drop-Off	8.4	0.9	8.6	1.2	8.5	1.0	9.3	1.0	9.7	1.2	9.6	1.0
	Congestion	2.6	0.4	6.1	0.9	3.8	0.5	1.8	0.4	5.7	0.8	2.7	0.4
Picker	Travel to location	16.1	2.2	18.5	1.4	16.4	1.7	16.0	2.2	17.9	1.4	14.9	1.7
	Travel to pick	38.7	1.7	33.7	1.5	36.9	1.2	31.3	1.6	25.7	1.3	30.0	1.2
	Picking	39.6	2.0	41.5	1.8	40.8	1.7	45.6	2.2	48.3	1.7	47.5	1.7
	Disturbance	5.6	0.3	6.4	0.3	5.9	0.3	7.2	0.4	8.1	0.3	7.6	0.3

It can also be noticed that the AMR division of time spent on each activity remains relatively stable across all ORSs, except for AMR congestion and waiting. The time spent on congestion is directly influenced by the method of releasing pickruns. In contrast, the time spent on waiting is influenced by the behavior of the pickers and indirectly affected by the method of releasing pickruns. Across all ORSs, a pattern can be distinguished when comparing the time AMRs spent waiting to the time pickers spent picking. If AMR waiting time is on the high side, then the total picking time is on the low side, and vice versa. It is because when pickers spend more time picking, it will directly impact the waiting time of AMRs. Apart from this, a more surprising pattern is visible between the congestion time and waiting time of AMRs. If the congestion percentage is high, then the time spent on waiting is low, and vice versa. The ORS with a high AMR congestion and a low AMR waiting time generates a higher average pick rate.

Furthermore, the differences in performance between the two batching methods mainly result from AMR congestion, picker traveling time, and picker picking time. It can be seen that the AMR congestion percentage and the time of the pickers traveling to picks are lower for the ORSs using the SH. Also, the total picking time increased by approximately 6–8 %. This can be explained by the fact that the SH enables combinations of picklists in close proximity to each other, whereas in regular batching, picklists are combined based on their belongings to customer order. The SH allows two picklists to (partially) overlap, creating a dense pickrun with locations close to each other. In most cases, regular batching will create pickruns with two picklists that need to be picked sequentially since two picklists from the same customer order can not have any overlap. It means that regular batching covers more travel distance and thus has a higher probability of creating AMR overtakes. If the total distance of a pickrun is smaller, picking can be conducted more efficiently. This causes an AMR to spend less time in the warehouse, preventing AMR congestion. It can be concluded that the performance of a COPS is highly dependent on the efficacy of a batching method.

6 CONCLUSIONS

In this paper, we study a collaborative order picking system with pickers and autonomous mobile robots. Several order release strategies, combining batching and releasing rules, are proposed to improve the pick rate performance per AMR compared to the current practice. We assess the effect of the proposed ORSs on a COPS through a developed simulation model. The computational results reveal that the saving heuristic for the batching decision has a more significant impact on the system performance than the zone-based rules for the releasing decision (i.e., 16 % vs. 4 % improvement). In addition, their combinations can yield up to about 20 % increase in the average pick rate. This yield implies a considerable potential investment saving for future users of the system. In other words, an efficient ORS can help achieve the same performance target with fewer AMRs or pickers. Interesting future research works include an integrated method for the batching and releasing decisions, more complex ORSs using reinforcement learning techniques, and an improved picker optimizer that would benefit designed ORSs, and the willingness of human order pickers to accept the new concept and trust this new way of working.

REFERENCES

Azadeh, K., R. De Koster, and D. Roy. 2019. "Robotized and Automated Warehouse Systems: Review and Recent Developments". *Transportation Science* 53(4):917–945.

Azadeh, K., D. Roy, and M. R. de Koster. 2020. "Dynamic Human-Robot Collaborative Picking Strategies". *SSRN Electronic Journal*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3585396, accessed 16th June 2022.

Bartholdi, J.J. and Hackman, S.T. 2019. "Warehouse & Distribution Science". https://www2.isye.gatech.edu/~jjb/wh/book/editions/wh-sci-0.96.pdf, accessed 16th June 2022.

Boysen, N., R. de Koster, and F. Weidinger. 2019. "Warehousing in the E-commerce Era: A Survey". *European Journal of Operational Research* 277(2):396–411.

Clarke, G., and J. W. Wright. 1964. "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points". *Operations Research* 12(4):568–581.

de Koster, R., T. Le-Duc, and K. J. Roodbergen. 2007. "Design and Control of Warehouse Order Picking: A Literature Review". *European Journal of Operational Research* 182(2):481–501.

Fragapane, G., R. de Koster, F. Sgarbossa, and J. O. Strandhagen. 2021. "Planning and Control of Autonomous Mobile Robots for Intralogistics: Literature Review and Research Agenda". *European Journal of Operational Research* 294(2):405–426.

Ghelichi, Z., and S. Kilaru. 2021. "Analytical Models for Collaborative Autonomous Mobile Robot Solutions in Fulfillment Centers". *Applied Mathematical Modelling* 91:438–457.

Ho, Y.-C., T.-S. Su, and Z.-B. Shi. 2008. "Order-batching Methods for an Order-picking Warehouse with Two Cross Aisles". *Computers & Industrial Engineering* 55(2):321–347.

Mann, H. B., and D. R. Whitney. 1947. "On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other". *The Annals of Mathematical Statistics* 18(1):50 – 60.

Meller, R. D., D. Nazzal, and L. M. Thomas. 2018. "Collaborative Bots in Distribution Centers". In *Proceedings of 15th International Material Handling Research Colloquium*, edited by E. Ballot, A. L. Carrano, K. Ellis, W. Ferrell, J. D. Porter, and D. Reed, 1–6. Savannah, Georgia: DigitalCommons@Georgia Southern.

Team SimPy 2020. "SimPy 4.0.2. DES for Python". https://simpy.readthedocs.io/en/latest/, accessed 1st March 2021.

Tompkins, J. A., J. A. White, Y. A. Bozer, and J. M. A. Tanchoco. 2003. *Facilities Planning*. New Jersey: John Wiley & Sons.

Van Rossum, G., and F. L. Drake. 2009. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Žunić, E., A. Beširević, S. Delalić, K. Hodžić, and H. Hasić. 2018. "A Generic Approach for Order Picking Optimization Process in Different Warehouse Layouts". In *Proceedings of the 41st International Convention on Information and Communication Technology, Electronics and Microelectronics*, edited by K. Skala, M. Koricic, T. G. Grbac, M. Cicin-Sain, V. Sruk, S. Ribaric, S. Gros, B. Vrdoljak, M. Mauher, E. Tijan, P. Pale, and M. Janjic, 1000–1005. Opatija, Croatia: Institute of Electrical and Electronics Engineers, Inc.

Žulj, I., H. Salewski, D. Goeke, and M. Schneider. 2022. "Order Batching and Batch Sequencing in an AMR-assisted Picker-to-parts System". *European Journal of Operational Research* 298(1):182–201.

AUTHOR BIOGRAPHIES

Q.V. DANG is an assistant professor in the Department of Industrial Engineering and Innovation Sciences at the Eindhoven University of Technology. His research interests include planning and scheduling automated manufacturing or services systems and collaboration between autonomous mobile robots and human operators. His email address is q.v.dang@tue.nl.

T.G. MARTAGAN is an associate professor in the Department of Industrial Engineering and Innovation Sciences at the Eindhoven University of Technology. Her research interests include stochastic modeling and optimization of manufacturing systems. Her email address is t.g.martagan@tue.nl.

I.J.B.F. ADAN is a Professor at the Department of Industrial Engineering of the Eindhoven University of Technology. His current research interests are in the modeling and design of manufacturing systems, warehousing systems and transportation systems, and more specifically, in the analysis of multi-dimensional Markov processes and queueing models. His email address is i.adan@tue.nl.

J.H.B. KLEINLUGTENBELD has a Master of Science in Manufacturing Systems Engineering from the Eindhoven University of Technology. His interests are manufacturing/warehouse systems design and supply chain optimization. After graduation, he is now a Supply Chain Analyst Manufacturing at Vanderlande Industries. His email address is jan.kleinlugtenbeld@vanderlande.com.