



COMPARISON OF ROUTING ALGORITHMS FOR STORAGE AND RETRIEVAL MECHANISM IN CYLINDRICAL AS/RS

Vytautas V. Janilionis¹, Žilvinas Bazaras², Vytautas Janilionis³

^{1,2}*Dept of Transport Engineering, Kaunas University of Technology, Kaunas, Lithuania*

³*Dept of Applied Mathematics, Kaunas University of Technology, Lithuania*

Submitted 7 May 2014; resubmitted 19 October 2014; accepted 23 October 2014;
first published online 14 January 2015

Abstract. The objective of this research is to propose new routing algorithms for the Storage and Retrieval Mechanism (SRM) in the Cylindrical Automated Storage and Retrieval System (C-AS/RS) and contribute to the system conceptualization by investigating the maximum achievable retrieval request rates for different routing algorithms and system parameters. For this purpose, flexible and detailed simulation model was developed and investigated for 2 SRM types, 3 routing algorithms and a feasible set of system movement and load transfer time parameters. Based on the simulation output, the regression models for different SRM types and routing algorithms were developed for predicting the maximum retrieval request rate. The differences of the average maximum retrieval request rate were evaluated for various system configurations and routing algorithms. The alternative to optimal routing algorithm was proposed, reducing the system performance only by 1.4÷2.4% on average, but requiring significantly less calculations when planning the SRM tour. In addition, the system analysis indicated that SRM vertical velocity and load transfer time have the highest impact on the system performance and for different SRM types the average maximum retrieval request rates differ by 22.2÷31.8%.

Keywords: cylindrical automated storage and retrieval system; routing algorithm; branch and bound algorithm; automated warehouse; simulation; regression analysis.

Introduction

Warehouses play a major role in modern supply chain by consolidating a bulk production and distributing it among the customers. Today's warehouses require to handle complex flows and provide a wide range of different services, so they should be equipped with intelligent and reliable technologies, which would solve the transportation and distribution problems of loads efficiently. The most common technology for the automated transportation of loads in the warehouse is Automated Storage and Retrieval Systems (AS/RS), which have been integrated into many warehouses and distribution centres so far. The main purpose of the AS/RS is to automate the load storage and retrieval process in the warehouse, reduce labour costs, human error rates, increase system performance and storage capacity. Different industry requirements have created the demand for the development of various types of AS/RS such as traditional crane based, autonomous vehicle, person on

board, split platform, carousel and vertical lift systems (Gu *et al.* 2010). Each type of AS/RS can be customized for the required system design and will benefit the material handling process by reduced labour costs, storage footprint, time required to store and retrieve items from the storage locations. In addition, AS/RS allows to achieve high inventory optimization and tracking level and increase overall system performance. However, significant system design issues should be solved when considering the AS/RS application, because it requires high initial investment, usually contains fixed layout with limited capacity and the equipment is highly utilized. The complex system design decisions should be made in order to define physical and control parameters when selecting the AS/RS for a specific material handling system (Roodbergen, Vis 2009).

The AS/RS physical design mainly includes the configuration of storage rack, Storage/Retrieval Machine (SRM) and the Input/Output (I/O) locations. The rack

Current original article has been corrected since first publishing online. Please see the statement of correction (doi:10.3846/16484142.2016.1161935 of the corrigendum).

Corresponding author: Vytautas Janilionis
E-mail: vytautas.janilionis@gmail.com

Copyright © 2015 Vilnius Gediminas Technical University (VGTU) Press
<http://www.tandfonline.com/TRAN>



Taylor & Francis
Taylor & Francis Group

parameters such as storage location dimensions, number of levels, aisles and bays are system design specific and their values depend on the required storage space and building restrictions. For any type of AS/RS it is very important to find the optimal rack configuration, because it directly affects the SRM cycles and the overall system performance (Hu *et al.* 2005). The SRM design requires to define the structure specifying its type, number and layout of Load Handling Devices (LHDs) on it and select the load handling equipment with appropriate movement velocities, accelerations and load transfer times. Increasing the number of LHDs on the SRM will improve the system performance significantly and allow to achieve better results for the same rack configuration compared to the SRM with less LHDs (Lerher *et al.* 2011). Since having more LHDs on the SRM will result in complex equipment cycles, the operation and implementation costs of various SRMs should be taken into account. The SRM movement parameters are always expected to have an impact on the system performance and it seems that faster SRMs should be more preferable. However, the rack configuration defines the SRM movement boundaries, so the velocities and accelerations should be adjusted to the rack parameters, because it is not efficient to have extremely fast SRM with powerful engine, which never achieves the maximum velocity defined and consumes lots of energy (Lerher *et al.* 2014). In addition, the properties of the unit load, which will be transported by the SRM, should be also considered when selecting movement parameters, e.g. high loads might not be compatible with high equipment accelerations. The system I/O locations connect the AS/RS to the other systems in the warehouse, so the number of those locations and their layout should be optimized and ensure the minimum performance losses during the load exchange between the different systems (Sari *et al.* 2007).

The determination of the optimal system physical parameter set will not result in the efficient system performance, because it can only be achieved with appropriate control policies. Therefore, having the same system physical parameter set, the performance can be improved significantly by applying efficient equipment control policies (Gagliardi *et al.* 2014a). Typical control design decisions for the AS/RS include storage assignment, scheduling and dwell-point selection policies. The storage assignment policies define the rules for storing the arriving loads in the storage rack and the most commonly used policies are: random, class-based and full-turnover (Zaerpour *et al.* 2013). The random policy is the most common and straightforward because the arriving loads are stored and retrieved from the random locations in the rack. Since, many system details are usually not known at the early design phase, the random policy is the most appropriate assumption. The other two policies are more sophisticated and could improve the system performance significantly, but they are strongly dependent on the specific design case, load and product types. The scheduling policies define the equipment routing and scheduling rules, which define the sequence of SRM operations required to process the target requests. There are many heuristics available for

optimal scheduling, but most of them are system design specific and require custom adjustments when applied to other systems. After examining a number of investigations on the control policies, the general guidelines for efficient SRM scheduling can be defined (Carlo, Vis 2012; Yin, Rau 2006; Ekren *et al.* 2010):

- equally balanced in time equipment workload.
- combined storage and retrieval operation cycles, e.g. after completing the storage request, the SRM should process the next retrieval request while it is still in the rack and before it travels to the I/O location.
- dynamic scheduling and look ahead policies for selecting the next request to process. The look ahead policies consider a defined number of subsequent system requests and arrange them in specific sequence, which minimizes a certain criterion. Due to the system dynamics, the request process sequence have to be adjusted as soon as new requests are available.

The dwell point selection policy mainly defines the location, where the SRM should travel when it have processed all available requests and become idle. However, the dwell point policy only has a significant impact on the performance of low utilised systems, because the SRM becomes idle quite often. Highly utilised systems can hardly use the benefits of dwell-point policy, and the rule for the SRM to remain at the location of the last operation turns out to be efficient enough (Meller, Mungwattana 2005).

This paper continues the research on the cylindrical automated storage and retrieval system (C-AS/RS), which contains a new type of the SRM with a number of vertically moving and rotating LHDs, placed in the centre of the octagonal shape rack. The design process of C-AS/RS should consider all issues mentioned previously and specifically focus on the control policies, which have high complexity level due to the simultaneous movement and rotation of the SRM.

The previous researches on C-AS/RS investigated the impact of I/O positions to the C-AS/RS performance with independent vertically moving LHDs (Janilionis, Bazaras 2012a), the impact of I/O positions together with a number of LHDs on the SRM to the system performance with interconnected vertically moving LHDs (Janilionis, Bazaras 2012b) and compared two SRM types for different combinations of retrieval rates, I/O number and positions (Janilionis, Bazaras 2013).

Since the routing algorithms for the SRM in C-AS/RS have not been investigated in any research so far, the objective of this research is to propose the SRM routing algorithms, considering optimal case, which requires full permutation and exponential computation time, and alternative algorithms, which are less accurate but require less calculations. For this purpose, the C-AS/RS simulation model is created and regression models developed from the simulation results for investigating the influence of routing algorithms and system parameters to the system performance. This research also contributes to the C-AS/RS conceptualization by investigating the maximum achievable retrieval request rates λ_{out}^{max} for different routing algorithms and system parameters,

because the analysis of maximum system capabilities for different parameters provides a valuable support to the system designers helping them to understand better the system behaviour (Foley *et al.* 2002).

The complexity of the SRM operations limits the development of analytical performance estimation models for C-AS/RS and does not allow to make reasonable assumptions and simplifications without losing accuracy when investigating such systems. Hence, the simulation model for the system investigation is necessary, which can estimate the interactions between the system parameters (Gagliardi *et al.* 2014b). A discrete event simulation software *AutoMod* (LeBaron, Jacobsen 2007) is used for the model development and integrated with the statistical analysis software SAS (Khattree, Naik 2000) for the simulation experiment result analysis.

1. C-AS/RS Integration to Order Picking System

The major application field of the proposed C-AS/RS is automated Order Picking Systems (OPS), where customer orders are assigned to operators at the work stations and required products are transported on conveyor system from the storage area operated by AS/RS to the workstations. The C-AS/RS is designed to solve the load-sequencing problem and improve the overall OPS performance.

Typical OPS layout and flows are showed in Fig. 1. The multi-aisle storage structure is linked to workstations via transport conveyor system (Andriansyah *et al.* 2011). Each operator at the workstation receives a sequence of orders to process and has to put the required number of product items from arriving product loads to the active customer order load. After the customer order is completed, the order load leaves the station and

travels to consolidation or despatch zone and operator can start on the next order. The arriving product loads from the storage must match the order picking sequence at the station. Sequencing of the product totes is usually maintained on the AS/RS front zone conveyor, so that the arriving product loads match the order picking sequence at the workstation. However, handling small orders such as e-commerce in this system will result in a high sequencing requirement, because each order requires a small number of product items, but the range of products is very high in the storage resulting in completely random distribution of stock in the storage. Retrieving products from the storage locations and sequencing on the front zone conveyor, will create additional delay times in product travel times to workstations and reduce the workload of the operators and AS/RS equipment significantly. Therefore, even though the system has enough resources, they will be not used efficiently to achieve the target flow rates.

The integration of C-AS/RS into OPS (Fig. 2) will improve the overall system performance, because it operates as a temporary storage buffer locally at the workstations, consolidates the arriving product totes and transports them to the operator in required sequence for pending orders. The design of C-AS/RS should allow the fast handling of products in the buffer and creation of a steady flow between the storage area and workstations, ensuring the efficient utilization of SRMs and operators.

The C-AS/RS will remove the sequencing restriction from the AS/RS front zone conveyors, so that SRMs could operate at optimal rates and deliver as many products as possible to the workstations, where they will be rapidly re-sorted by the C-AS/RS and keep the steady workflow to the operator.

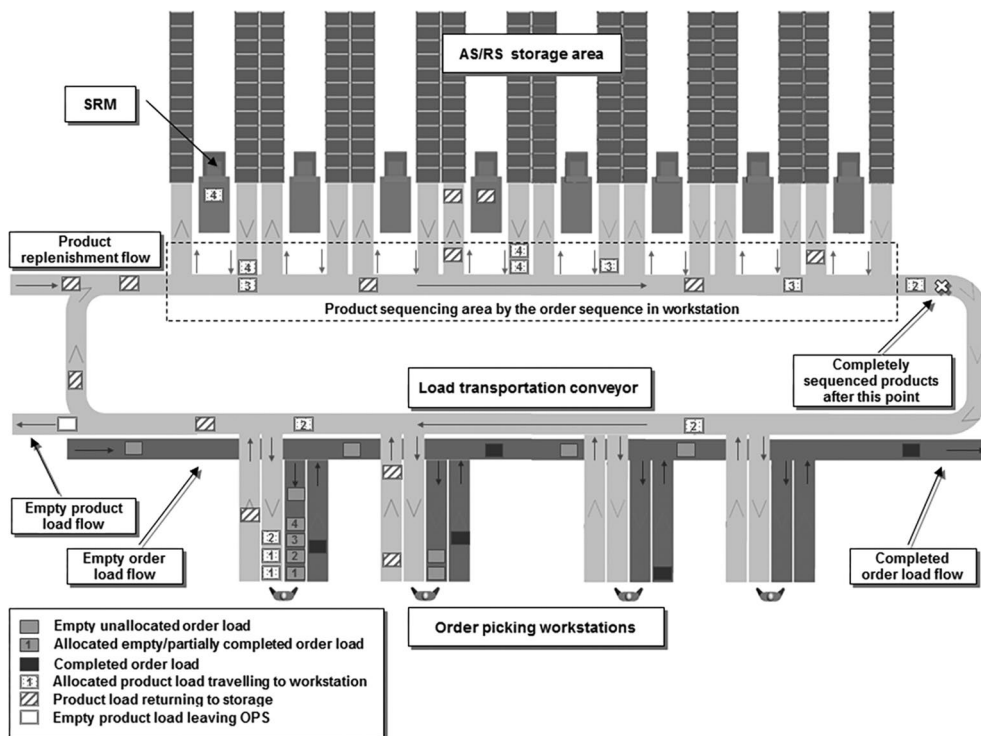


Fig. 1. Automated OPS

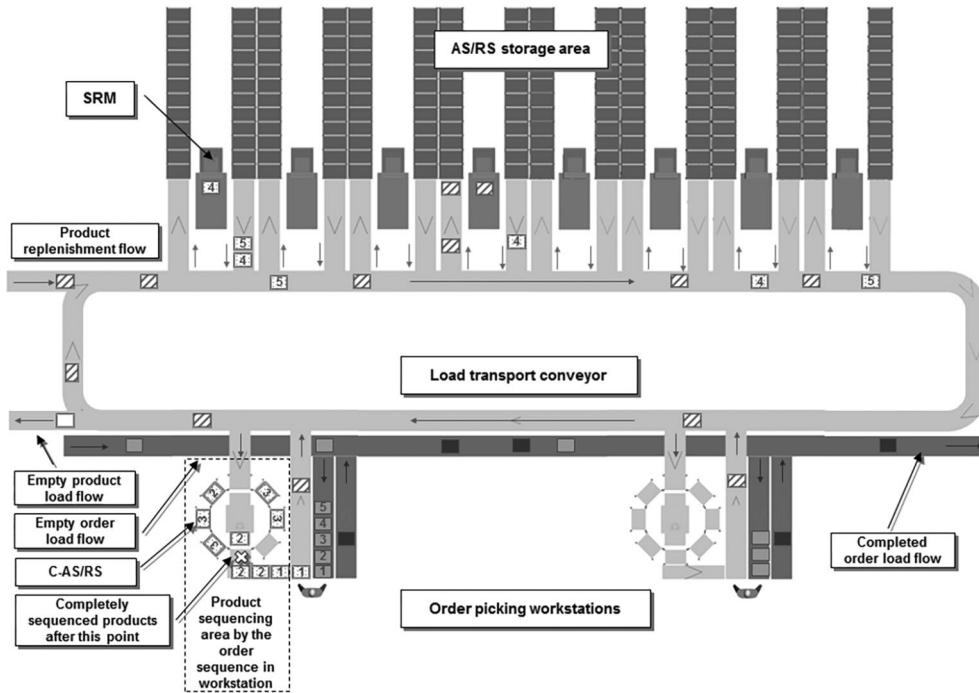


Fig. 2. C-AS/RS integration to OPS

2. C-AS/RS Components and Parameters

The C-AS/RS structural components include octagonal shape storage rack, I/O locations and SRM with LHDs. Fig. 3 shows the example system considered in this research with the major system components defined and dimensions proposed.

The C-AS/RS requires intelligent control hardware and software, which would ensure not only the optimal system functioning, but also the reliable connection to other warehouse systems. The rack layout has columns and levels with storage locations arranged in a cylindrical octagonal shape and an integrated SRM, which transports loads from/to rack locations. The SRM has a

number of LHDs attached to it and can move between the levels and rotate between the columns simultaneously, so that each LHD can access any rack location. The LHD transfers the load on/off when it arrives to the destination location. The SRM types SRM₁ and SRM₂ are considered in this research and the key difference between them is that LHDs can move vertically independently for SRM₂ and move as a single unit only for SRM₁, but all LHDs are rotated as a single unit for both types. Since the major C-AS/RS application is its integration into warehouse OPS, the interface conveyors are installed into I/O locations and linked to the required systems in the warehouse. System design is flexible enough and allows the required number of I/O locations to be

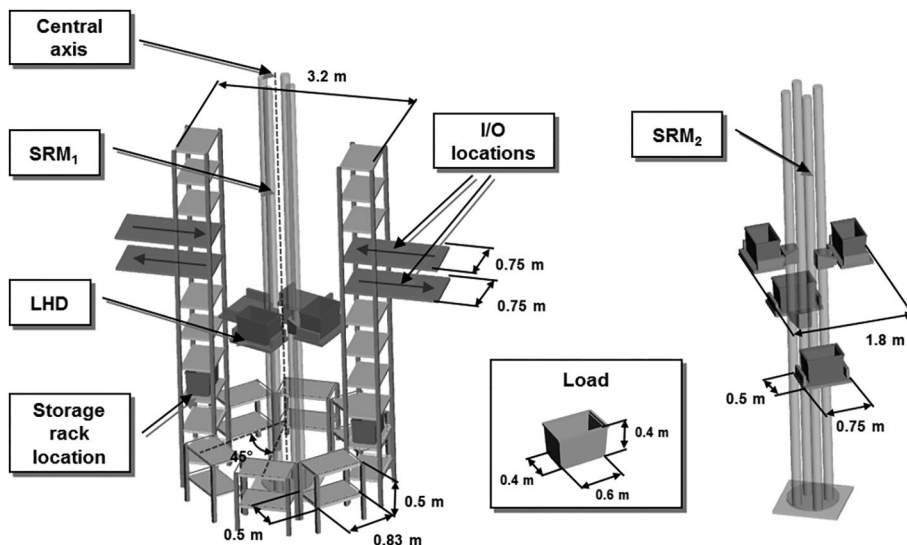


Fig. 3. C-AS/RS structure

positioned in any rack column and level. The layout of I/O locations should optimize the movements of the SRM so that it picks or drops the destination loads in a single transfer. The system is also scalable level and LHD wise, so the number of storage levels and LHDs on the SRM can be selected depending on the required system design.

The C-AS/RS performance and integration is highly dependent on the system control hardware and software selection. The hardware solutions are beyond the scope of this research, because it will focus on defining the efficient system control policies, which will provide the basic recommendations for the system designers. One of the most important problems to be solved when defining the C-AS/RS control policies is the SRM routing problem. Before defining the routing problem in more details, the key system parameters will be specified.

The SRM has vertical velocity v_{vert} [m/s] with acceleration, rotation velocity v_{rot} [m/s] with acceleration, a number of LHDs N_{LHD} , where each of them transfers a load in time [s] to/from rack location, and each location is positioned in level $l_i \in L$, where $L = \{l_i | i = \overline{1, N_L}\}$, and column $c_j \in C$, where $C = \{c_j | j = \overline{1, N_C}\}$. The total number of rack locations is $N_L \cdot N_C$, where N_L defines the total number of storage levels and the number of rack columns $N_C = 8$ is fixed due to the proposed system design. Some rack locations must be blocked from storage and used as I/O locations with the load handling conveyors installed. In order to access any rack location, each LHD has to make a number of rotations and vertical movements. Since the SRM moves all LHDs simultaneously, the feasible rotation set $R = \{r_k | r_k = 45^\circ \cdot \alpha, \alpha = \overline{-3, 4}, k = \overline{1, 8}\}$ of size N_R is defined, which specifies all possible rotations, measured in degrees, the SRM can make relatively to its central axis. When SRM rotates, each LHD rotates simultaneously by degrees between columns c_{j_1} and c_{j_2} ($j_1, j_2 = \overline{1, N_C}$) in time $t_{c_{j_1}c_{j_2}}$ and each LHD moves vertical distance between levels l_{i_1} and l_{i_2} ($i_1, i_2 = \overline{1, N_L}$) in time. Therefore, if two empty storage locations $s_{i_1c_{j_1}}$ and $s_{i_2c_{j_2}}$ are considered, the LHD travel time between the locations is $\text{MAX}(t_{c_{j_1}c_{j_2}}, t_{l_{i_1}l_{i_2}})$, because of the SRM feature to move and rotate simultaneously.

All C-AS/RS rack locations can be classified into four location sets by the type: input locations $S_I = \{I_{l_i c_j} | l_i \in L, c_j \in C\}$, output locations $S_O = \{O_{l_i c_j} | l_i \in L, c_j \in C\}$, empty storage locations $S_{ST} = \{s_{l_i c_j} | l_i \in L, c_j \in C\}$ and retrieval locations with loads in them $S_{RT} = \{z_{l_i c_j} | l_i \in L, c_j \in C\}$, so the union $S_I \cup S_O \cup S_{ST} \cup S_{RT}$ defines the total rack capacity of size $N_L \cdot N_C$.

3. SRM Cycle

Similarly to any other type of AS/RS, the SRM in C-AS/RS operates in cycles and the full cycle consists of the Storage Pick (I), Storage Drop (ST), Retrieval

Pick (RT) and Retrieval Drop (O) tours. The individual tour planning is done by the SRM routing procedure, which is called every time after the SRM completes a tour and before it goes idle. Single tour planning is considered in this research, so the routing procedure plans only the next tour, although the system design allows planning more tours in advance. The core part of the routing procedure is the SRM routing algorithm, which estimates the time required to execute different routing scenarios to complete a tour, and selects the scenario with minimum total travelling time. Before starting each SRM tour, the routing procedure input requires initial $S_{tour}^{init} \subseteq S_I \cup S_O \cup S_{ST} \cup S_{RT}$ and destination $S_{tour}^{dest} \subseteq S_{tour}$ SRM location sets, where $tour \in \{I, ST, RT, O\}$, because the routing procedure needs to know where the SRM currently is and where it needs to travel, so it could plan the tour between the initial and destination locations. After the tour planning is completed, the routing procedure provides the following output:

- SRM rotation tuple $R_{tour} = (r_1, r_2, \dots, r_n)$ of variable length and fixed sequence, where $1 \leq n \leq N_{LHD}$, defines the tour rotations the SRM has to make so that each LHD could access its destination location.
- Active LHDs tuple $H_{tour} = (H_1, H_2, \dots, H_n)$, where $1 \leq n \leq N_{LHD}$ and each sub-tuple H_k contains active LHDs h_i ($1 < i < n$) in each rotation r_k .
- Destination location tuple $S_{tour}^* = (S_1, S_2, \dots, S_n)$, where $1 \leq n \leq N_{LHD}$ and each sub-tuple S_k contains destination locations matched to active LHDs in H_k in each rotation r_k .

During the application of the routing procedure output, the SRM strictly follows rotation sequence defined by R_{tour} and routes active LHDs H_{tour} to destination locations S_{tour} in each rotation.

Fig. 4 shows the C-AS/RS rack mapped to the 2D coordinate system defined by rack levels $l_i \in L$ and columns $c_j \in C$ and illustrates the routing procedure output application for the full SRM₂ cycle of the system with $N_{LHD} = 4$ and the following tour specification:

- **Storage pick.** The SRM initial location is defined by the tuple $S_{SP}^{init} = (s_{l_{i+3}c_7}, s_{l_{i+2}c_1}, s_{l_{i+4}c_3}, s_{l_{i+5}c_5})$, where each element corresponds to the initial location of LHDs (h_1, h_2, h_3, h_4) . The initial SRM location might be the destination location of the last tour or the dwell point location, where the SRM travels in the case of being idle. Before the SRM starts travelling from the initial location, the destination location set $S_{SP}^{dest} = \{I_{l_{i+1}c_7}, I_{l_{i+1}c_1}, I_{l_{i+1}c_3}, I_{l_{i+1}c_5}\}$ is created based on the set S_I . The rotations $R_{SP} = (r_1) = (0^\circ)$ are required for active LHDs $H_{SP} = (H_1) = (h_1, h_2, h_3, h_4)$ to arrive to the system input locations $S_{SP}^* = (S_1) = (I_{l_{i+1}c_7}, I_{l_{i+1}c_1}, I_{l_{i+1}c_3}, I_{l_{i+1}c_5})$ and pick up the target loads. The I/O layout should be designed in the optimal way, so that the SRM requires minimum travelling time between the I/O locations and all target loads could be picked up in a single transfer time t_{tr} .

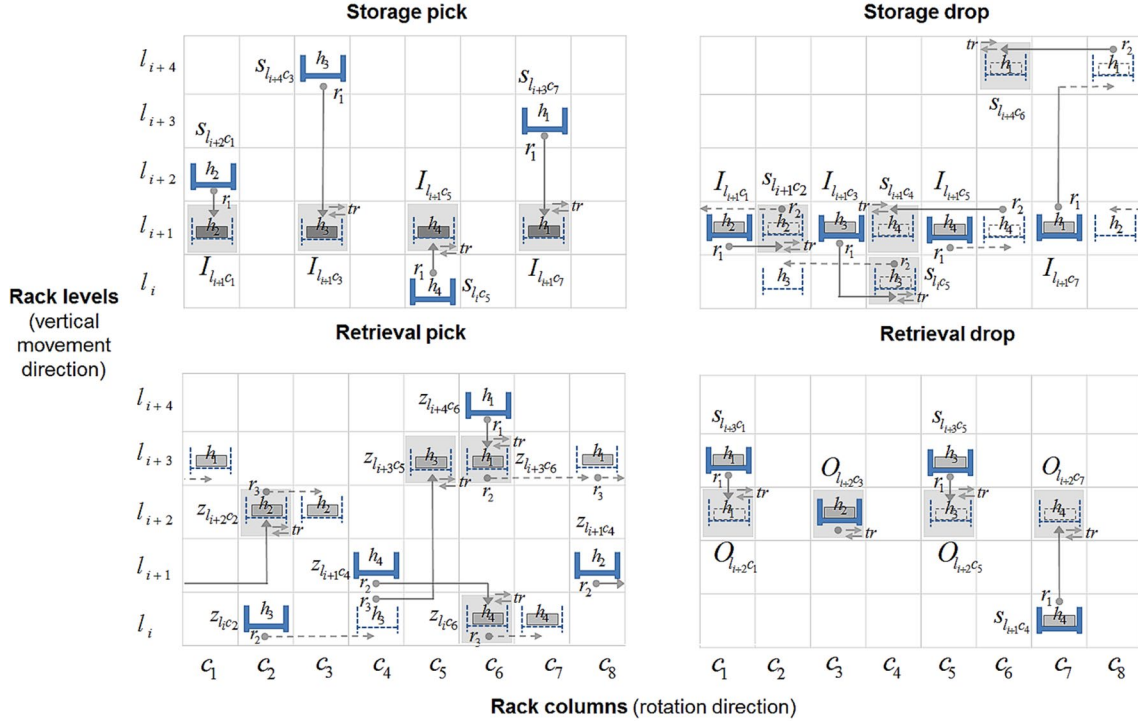


Fig. 4. Full SRM 2 cycle

- **Storage drop.** The SRM travels from the system input location $S_{SD}^{init} = S_{SP}^*$ to the empty storage locations $S_{SD}^{dest} = \{s_{l_{i+4}c_6}, s_{l_{i+1}c_2}, s_{l_i c_5}, s_{l_{i+1}c_4}\}$. The rotations $R_{SD} = (r_1, r_2) = (45^\circ, -90^\circ)$ are required for active LHDs $H_{SD} = (H_1, H_2) = ((h_2, h_3), (h_1, h_4))$ to arrive to the empty storage locations $S_{SD}^* = (S_1, S_2) = ((s_{l_{i+1}c_2}, s_{l_i c_5}), (s_{l_{i+4}c_6}, s_{l_{i+1}c_4}))$ and drop off the on board loads.
- **Retrieval pick.** The SRM travels from the final storage drop tour location $S_{RP}^{init} = S_{SD}^*$ to the retrieval locations $S_{RP}^{dest} = \{z_{l_{i+3}c_6}, z_{l_{i+2}c_2}, z_{l_{i+3}c_5}, z_{l_i c_6}\}$. The rotations $R_{RP} = (r_1, r_2, r_3) = (0^\circ, 90^\circ, 45^\circ)$ are required for active LHDs $H_{RP} = (H_1, H_2, H_3) = (h_1, (h_2, h_4), h_3)$ to arrive to the retrieval locations $S_{RP}^* = (S_1, S_2, S_3) = (z_{l_{i+3}c_6}, (z_{l_{i+2}c_2}, z_{l_i c_6}), z_{l_{i+3}c_5})$ and pick up the target loads.
- **Retrieval drop.** The SRM travels from the final retrieval pick tour location $S_{RD}^{init} = S_{RP}^*$ to the system output locations $\{O_{l_{i+2}c_1}, O_{l_{i+2}c_3}, O_{l_{i+2}c_5}, O_{l_{i+2}c_7}\}$. The rotations $R_{RD} = (r_1) = (0^\circ)$ are required for active LHDs $H_{RD} = (H_1) = (h_1, h_2, h_3, h_4)$ to arrive to the system output locations $S_{RD}^* = (S_1) = (O_{l_{i+2}c_1}, O_{l_{i+2}c_3}, O_{l_{i+2}c_5}, O_{l_{i+2}c_7})$ and drop off the on board loads. Similarly to the storage pick tour, the I/O layout should be optimal and minimize the transfer time and SRM travelling time between the I/O locations.

After completing the retrieval drop tour, the SRM remains at the location of the last operation or starts a new cycle.

4. SRM Routing Algorithm

The routing algorithm makes a core part of the SRM routing procedure and directly plans the SRM tour, which specifies the movement steps the SRM has to follow in order to process the target loads. According to the C-AS/RS design, the SRM rotates on the central axis and transports active LHDs to the destination locations in each rotation, so rotations r_k ($k=1, n$) divide the tour into sub-tours, so $W_{tour} = (W_1, W_2, \dots, W_n)$, where $1 \leq n \leq N_{LHD}$. Therefore, each sub-tour W_k can be defined by the tuple $W_k = (r_k, H_k, S_k)$ with the following elements: rotation $r_k \in R_{tour}$, active LHDs tuple $H_k \in H_{tour}$ and destination location tuple $S_k \in S_{tour}^*$ in rotation r_k . The time required to complete a sub-tour is T_{W_k} and the total time to complete a tour is $T_{tour} = \sum_{k=1}^n T_{W_k}$ ($1 \leq n \leq N_{LHD}$), which should be minimized by the routing algorithm. Since rack locations are defined by columns and levels, the optimal algorithm should check all possible routing scenarios for SRM rotations $r_k \in R$ and locations accessed in each level $l_i \in L$ and column $c_j \in C$ by the LHDs. Therefore, a complete evaluation of all routing scenarios would find a global minimum of total tour time T_{tour} , but would result in exponentially large number of $(N_C \cdot N_L)^{N_{LHD}}$ scenarios in total, which might not be possible to evaluate com-

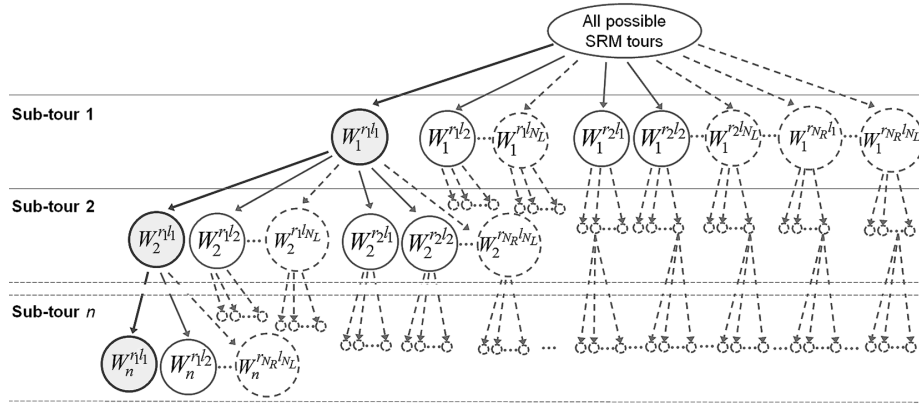


Fig. 5. Optimal routing algorithm solution tree

pletely. The optimal routing algorithm (A_1), based on general branch and bound framework, which is one of the main tools to solve the full enumeration problems and find the optimal solution, is proposed in this research (Farahani, Tari 2002). Thus, the following components will be defined for the SRM routing algorithm to be compatible with the proposed framework: solution tree with nodes (partial or complete solutions), bounding function, branching, pruning, node selection and termination rules.

Fig. 5 shows the sketch of the solution tree for the optimal routing algorithm where each node represents a sub-tour $W_k^{r_i l_j}$ ($k = \overline{1, n}, 1 \leq n \leq N_{LHD}$) in rotation $r_i \in R$ ($i = \overline{1, N_R}$) and level $l_j \in L$ ($j = \overline{1, N_L}$). The practical meaning of the node is that the SRM is rotated by r_i degrees and all available LHDs are transported to level l_j and corresponding column c_m ($m = \overline{1, N_C}$), where each LHD checks for eligible target load to process. Root node defines the set of all solutions and each subsequent node is expanded to $N_R \cdot N_L$ child nodes. Fig. 5 shows an example SRM tour marked in bold.

The fundamental principle of the branch and bound framework is to branch the tree only with feasible and promising nodes and limit the size of it as much as possible. The objective of the routing algorithm is to find the SRM tour, which processes all target loads in time T_{tour}^{min} , which is the global minimum in the solution space.

Fig. 6 represents the optimal SRM routing algorithm proposed in this research. The bounding function evaluates the current total tour time at each node and prunes on it, stopping further branching, if the node is not feasible or promising. The algorithm uses the depth-first node selection policy, which always tries to branch the tree further and find the incumbent solution early – complete tour which process all target loads in minimum time T_{tour}^{min} . For the feasible nodes, the SRM should find at least one target load to process in rotation r_i and level l_j . If the node is feasible, it should be also promising, so that the current total tour time T_{tour}^* at selected node does not exceed the current incumbent value, i.e. $T_{tour}^* < T_{tour}^{min}$. The algorithm terminates when all nodes have been explored and returns the last incumbent solution found.

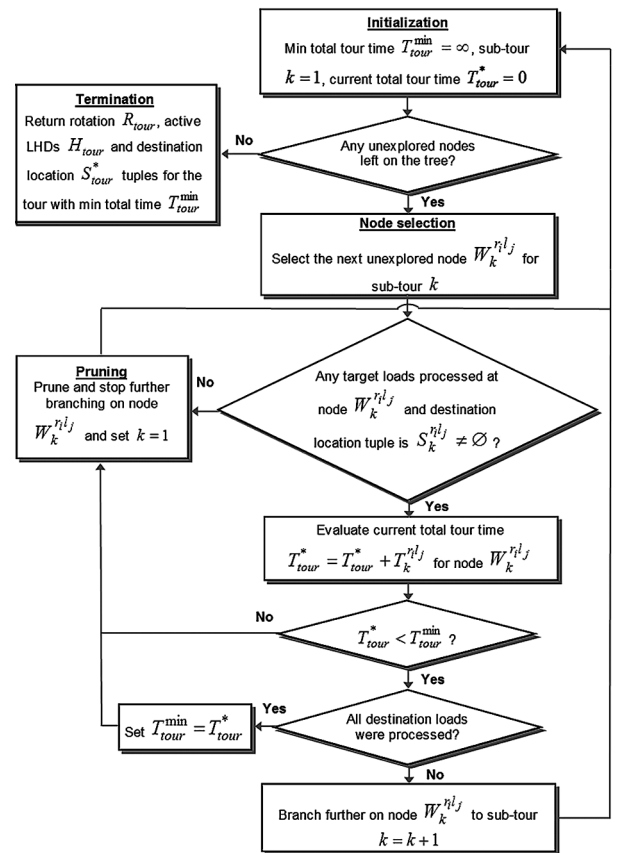


Fig. 6. Optimal routing algorithm

In addition to the optimal algorithm A_1 , this research proposes two alternative routing algorithms – A_2 and A_3 , which are predicted to be less accurate but reduce the calculations significantly.

The routing algorithm A_2 estimates all different SRM rotation $r_i \in R$ ($i = \overline{1, N_R}$) combinations and in each sub-tour W_k ($k = \overline{1, n}, 1 \leq n \leq N_{LHD}$) searches for the closest level $l_j \in L$ ($j = \overline{1, N_L}$) from the current SRM location with the maximum number of destination loads to process in a single transfer. This algorithm will require to evaluate maximum of $(N_C)^{N_{LHD}} \cdot N_L$ rout-

ing scenarios, which is a significantly smaller number compared to the optimal algorithm A_1 . Moreover, the algorithm A_2 is also based on the branch and bound framework, so the required calculations are reduced even more.

The algorithm A_3 is based on a simple optimization approach, because for each sub-tour W_k it searches for the closest rotation r_i and level l_j with the maximum number of destination loads to process in a single transfer. This algorithm only optimizes the next sub-tour, but does not guarantee the optimization of the whole tour, however, it requires only maximum of $N_C \cdot N_L \cdot N_{LHD}$ routing scenarios to evaluate in order to plan the SRM tour.

All afore mentioned routing algorithms are completely compatible with both SRM types, but some minor modifications will be required during realization due to the feature that LHDs can move vertically independently for SRM_2 and move as a single unit only for SRM_1 .

5. Results

The purpose of this research is to provide the comparative analysis of the SRM routing algorithms A_1 , A_2 and A_3 , evaluating the maximum retrieval request rate λ_{out}^{max} [loads/hour] for the system parameters such as SRM type (SRM), vertical velocity (v_{vert}) and acceleration, rotation velocity (v_{rot}) and acceleration and load transfer time. The detailed and parameterized simulation model of the C-AS/RS was developed in the *AutoMod* environment, which allowed to investigate the system performance for various sets of parameters and all proposed routing algorithms. Moreover, *AutoMod* statistical analysis module *AutoStat* was used to generate simulation results from multiple model runs and combined with the advanced statistical analysis software SAS for result analysis.

The investigated model parameters can be classified into fixed and variable parameter sets. The fixed parameters include: rack structure and size – $N_L \cdot N_C = 15 \cdot 8$, rack filling level – 95%, number of active LHDs on SRM $N_{LHD} = 4$, number of I/O locations $N_{IO} = 4$ and I/O positions $l_{IO} = 0708$ meaning that the C-AS/RS has 4 input conveyors installed at rack level 7 and 4 output conveyors installed at level 8. It has been already shown by Janilionis and Bazaras (2013) that the I/O locations should be positioned as close as possible to the middle of the rack and the I/O layout should be optimal so that all LHDs can drop off/pick up all loads simultaneously in a single transfer at the I/O level.

The variable model parameters include: routing algorithm $A \in \{A_1, A_2, A_3\}$, SRM type $SRM \in \{SRM_1, SRM_2\}$, vertical velocity $v_{vert} \in \{1, 1.25, 1.5, 1.75, 2, 2.25, 2.5, 3\}$ [m/s] and acceleration [m/s²], rotation velocity $v_{rot} \in \{33.75, 39.375, 45, 50.625, 56.25\}$ [deg/s] and acceleration $a_{rot} \in \{67.5, 78.75, 90, 101.25, 112.5, 123.75, 135\}$ [deg/s²] and load transfer time $t_{tr} \in \{1, 1.5, 2, 1.25, 1.75\}$ [s].

Since the system structure was defined, the maximum required calculations for the routing algorithms can be evaluated: optimal algorithm A_1 requires a large number of $(N_C \cdot N_L)^{N_{LHD}} = (8 \cdot 15)^4$ scenarios to evaluate compared to $(N_C)^{N_{LHD}} \cdot N_L = (8)^4 \cdot 15$ scenarios re-

quired for A_2 and $N_C \cdot N_L \cdot N_{LHD} = 8 \cdot 15 \cdot 4$ scenarios for A_3 to plan the SRM tour.

Using the simulation results and statistical analysis software SAS, 6 regression models were developed for SRM types 1 and 2 and routing algorithms A_1 , A_2 and A_3 , which predict the average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$ for all combinations of movement and load transfer time parameters. All regression equations have the same structure with the estimates of parameters $\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_{22}$ presented in Table 1.

$$\begin{aligned} \hat{\lambda}_{out}^{max} = & \hat{\beta}_0 + \hat{\beta}_1 \cdot v_{vert} + \hat{\beta}_2 \cdot v_{vert}^2 + \hat{\beta}_3 \cdot a_{vert} + \\ & \hat{\beta}_4 \cdot a_{vert}^2 + \hat{\beta}_5 \cdot v_{rot} + \hat{\beta}_6 \cdot v_{rot}^2 + \hat{\beta}_7 \cdot a_{rot} + \\ & \hat{\beta}_8 \cdot a_{rot}^2 + \hat{\beta}_9 \cdot v_{vert} \cdot a_{vert} + \hat{\beta}_{10} \cdot v_{rot} \cdot a_{rot} + \\ & \hat{\beta}_{11} \cdot a_{vert} \cdot a_{rot} + \hat{\beta}_{12} \cdot v_{vert} \cdot v_{rot} + \hat{\beta}_{13} \cdot t_{tr} + \\ & \hat{\beta}_{14} \cdot t_{tr}^2 + \hat{\beta}_{15} \cdot t_{tr} \cdot v_{vert} + \hat{\beta}_{16} \cdot t_{tr} \cdot a_{vert} + \\ & \hat{\beta}_{17} \cdot t_{tr} \cdot v_{rot} + \hat{\beta}_{18} \cdot t_{tr} \cdot a_{rot} + \hat{\beta}_{19} \cdot t_{tr} \cdot v_{vert}^2 + \\ & \hat{\beta}_{20} \cdot t_{tr} \cdot a_{vert}^2 + \hat{\beta}_{21} \cdot t_{tr} \cdot v_{rot}^2 + \hat{\beta}_{22} \cdot t_{tr} \cdot a_{rot}^2. \end{aligned}$$

Regression models include only statistically significant regressors ($p < 0.01$) for predicting the average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$. All models have very high values of adjusted R-Square (R_{adj}^2) greater than 0.99 and satisfy regression assumptions. The signs of all parameter estimates are the same in all regression models, so the trends of the average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$ are also the same for all models.

The results showed that the $\hat{\lambda}_{out}^{max}$ significantly depends on the SRM type and routing algorithm. The analysis of standardized regression parameter estimates (Khattree, Naik 2000) indicated that vertical velocity v_{vert} , load transfer time t_{tr} and the interactions of v_{vert} and t_{tr} have the most significant impact on the $\hat{\lambda}_{out}^{max}$. Fig. 7 represents 6 regression functions on each plot, which show the dependencies of $\hat{\lambda}_{out}^{max}$ on v_{vert} , t_{tr} , SRM types, and routing algorithms A_1 , A_2 and A_3 , when other system parameters v_{rot} , a_{rot} and a_{vert} are fixed. The top row of plots in Fig. 7 represents the ‘fast’ system case with the highest values of parameters v_{rot} , a_{rot} and a_{vert} while the bottom row represents the ‘slow’ system case with the lowest values of the same parameters.

Both SRM types of the C-AS/RS were investigated for the same values of the system parameters and the result analysis showed that higher values of the average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$ were achieved for systems with SRM_2 in all cases. The highest average maximum retrieval request rate $\hat{\lambda}_{out}^{max} = 1043$ was achieved with routing algorithm A_1 , SRM_2 , $v_{vert} = 3$ m/s, $t_{tr} = 1$ s and the ‘fast’ system parameter set compared to $\hat{\lambda}_{out}^{max} = 768$ achieved with SRM_1 , A_1 and the same parameter set ($\hat{\lambda}_{out}^{max}$ reduced by 26.3%). In general, any system with SRM_1 achieved 22.2÷31.6% smaller values of $\hat{\lambda}_{out}^{max}$ compared to system with SRM_2 . The $\hat{\lambda}_{out}^{max}$ achieved for systems with SRM_1 compared to systems with SRM_2 was 26.51±1.02% smaller when systems operated with routing algorithm A_1 , 25.85±1.11% smaller with A_2 and 28.42±0.98% smaller with A_3 (Table 2).

Table 1. Estimates of regression parameters for models predicting the average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$

Regression model number		1	2	3	4	5	6
SRM type		SRM_1	SRM_1	SRM_1	SRM_2	SRM_2	SRM_2
Routing algorithm		A_1	A_2	A_3	A_1	A_2	A_3
Regressor	Parameter	Parameter estimates					
$Intercept$	$\hat{\beta}_0$	163.93	145.66	19.23	131.56	79.45	37.68
v_{vert}	$\hat{\beta}_1$	288.04	295.35	331.32	372.45	367.98	391.34
v_{vert}^2	$\hat{\beta}_2$	-57.02	-57.85	-65.10	-73.42	-72.12	-75.85
a_{vert}	$\hat{\beta}_3$	33.93	33.48	31.31	36.01	34.48	33.34
a_{vert}^2	$\hat{\beta}_4$	-2.48	-2.50	-2.40	-2.73	-2.65	-2.62
v_{rot}	$\hat{\beta}_5$	7.80	7.57	7.40	13.14	13.86	11.97
v_{rot}^2	$\hat{\beta}_6$	-0.0740	-0.0714	-0.0716	-0.1207	-0.1265	-0.1111
a_{rot}	$\hat{\beta}_7$	1.43	1.33	0.93	2.14	2.0454	1.6120
a_{rot}^2	$\hat{\beta}_8$	$-6.06 \cdot 10^{-3}$	$-5.59 \cdot 10^{-3}$	$-3.98 \cdot 10^{-3}$	$-9.2010 \cdot 10^{-3}$	$-8.95 \cdot 10^{-3}$	$-7.08 \cdot 10^{-3}$
$v_{vert} \cdot a_{vert}$	$\hat{\beta}_9$	4.100	4.187	4.226	4.440	4.401	4.727
$v_{rot} \cdot a_{rot}$	$\hat{\beta}_{10}$	$7.45 \cdot 10^{-3}$	$7.06 \cdot 10^{-3}$	$3.72 \cdot 10^{-3}$	$1.34 \cdot 10^{-3}$	$1.28 \cdot 10^{-3}$	$9.25 \cdot 10^{-3}$
$a_{vert} \cdot a_{rot}$	$\hat{\beta}_{11}$	0.01202	0.01007	0.01187	0.01750	0.01712	0.0156
$v_{vert} \cdot v_{rot}$	$\hat{\beta}_{12}$	0.2088	0.2134	0.4897	0.3892	0.4010	0.5708
t_{tr}	$\hat{\beta}_{13}$	-77.81	-69.09	-5.37	-56.42	-24.60	-9.61
t_{tr}^2	$\hat{\beta}_{14}$	61.49	59.79	47.27	74.35	68.24	58.71
$t_{tr} \cdot v_{vert}$	$\hat{\beta}_{15}$	-103.34	-106.69	-114.51	-129.05	-126.81	-125.04
$t_{tr} \cdot a_{vert}$	$\hat{\beta}_{16}$	-13.97	-13.97	-13.048	-14.70	-13.97	-13.28
$t_{tr} \cdot v_{rot}$	$\hat{\beta}_{17}$	-2.89	-2.76	-2.789	-4.612	-4.975	-4.219
$t_{tr} \cdot a_{rot}$	$\hat{\beta}_{18}$	-0.6198	-0.5511	-0.3734	-0.8969	-0.8541	-0.6470
$t_{tr} \cdot v_{vert}^2$	$\hat{\beta}_{19}$	18.32	18.71	19.39	22.63	22.05	20.82
$t_{tr} \cdot a_{vert}^2$	$\hat{\beta}_{20}$	0.7948	0.8229	0.7669	0.8570	0.8221	0.7823
$t_{tr} \cdot v_{rot}^2$	$\hat{\beta}_{21}$	0.0239	0.0227	0.0230	0.0361	0.0393	0.0333
$t_{tr} \cdot a_{rot}^2$	$\hat{\beta}_{22}$	$2.03 \cdot 10^{-3}$	$1.80 \cdot 10^{-3}$	$1.28 \cdot 10^{-3}$	$2.87 \cdot 10^{-3}$	$2.80 \cdot 10^{-3}$	$2.17 \cdot 10^{-3}$
Model summary statistics							
R_{adj}^2		0.996	0.996	0.993	0.992	0.993	0.992
\sqrt{MSE}		4.79	4.52	5.91	8.67	7.98	8.35

Moreover, only a small difference was observed comparing the λ_{out}^{max} values for systems operating with routing algorithm A_1 to systems operating with A_2 . Systems operating with SRM_1 and routing algorithm A_2 achieve only $1.39 \pm 0.7\%$ smaller values on average of $\hat{\lambda}_{out}^{max}$ compared to A_1 and $2.38 \pm 0.46\%$ smaller values for systems with SRM_2 . However, the algorithm A_2 requires $(N_L)^{N_{LHD}-1}$ times less routing scenarios to evaluate in order to plan the SRM tour, i.e. 15^3 times less calculations

for A_2 are required for the C-AS/RS considered in this research compared to A_1 . For example, the ‘fast’ system with SRM_1 , $t_{tr} = 1$, $v_{vert} = 1$ and operating with routing algorithm A_1 (Fig. 7) achieves $\hat{\lambda}_{out}^{max} = 825$ compared to only 2.4% smaller value of $\hat{\lambda}_{out}^{max} = 805$ achieved with A_2 . In addition, the analysis indicated that for systems operating with routing algorithm A_3 , the $\hat{\lambda}_{out}^{max}$ reduced by $9.96 \pm 2.64\%$ for systems with SRM_1 and by $7.57 \pm 1.93\%$ for SRM_2 compared to optimal algorithm A_1 .

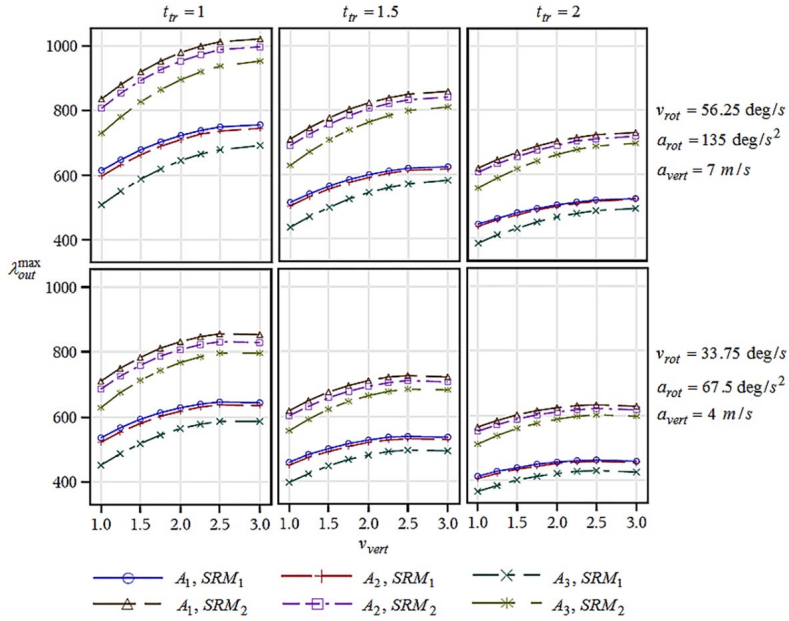


Fig. 7. Average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$ dependence on SRM types, routing algorithms, v_{vert} and t_{tr} , when other parameters are fixed

Table 2. Comparison of percentage differences of $\hat{\lambda}_{out}^{max}$ between systems with different SRM types and routing algorithms

Subgroup for calculation percentage difference		Percentage difference of $\hat{\lambda}_{out}^{max}$ descriptive statistics			
SRM type	Routing algorithm	Min [%]	Max [%]	Mean [%]	SD [%]
Diff 1-2	A_1	23.2	29.1	26.51	1.02
Diff 1-2	A_2	22.2	28.5	25.85	1.11
Diff 1-2	A_3	25.5	31.6	28.42	0.98
SRM_1	Diff A_2-A_1	0.0	3.2	1.39	0.70
SRM_1	Diff A_3-A_1	4.8	18.1	9.96	2.64
SRM_1	Diff A_3-A_2	4.0	16.1	8.59	2.33
SRM_2	Diff A_2-A_1	0.0	4.2	2.38	0.46
SRM_2	Diff A_3-A_1	3.7	13.7	7.57	1.93
SRM_2	Diff A_3-A_2	2.1	10.4	5.32	1.67

Conclusions

This research continued the previous research (Janilionis, Bazaras 2013) on the C-AS/RS, proposing and formalizing 3 routing algorithms for the SRM, which use different route planning strategies and require different number of calculations depending on the complexity. The optimal algorithm A_1 evaluates $(N_C \cdot N_L)^{N_{LHD}}$ routing scenarios in order to plan the tour for the SRM, compared to $(N_C)^{N_{LHD}} \cdot N_L$ scenarios required for A_2 and only $N_C \cdot N_L \cdot N_{LHD}$ scenarios required for A_3 . The detailed simulation model of the C-AS/RS was developed in the *AutoMod* environment, which allowed to investigate the maximum retrieval request rate λ_{out}^{max} for various sets of system parameters and routing algorithms. The developed simulation model is parameterized and flexible, so that various system configurations can be easily investigated.

Based on the experimental simulation data, 6 regression models were developed for 2 SRM types and

3 routing algorithms, which predict the average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$ in the feasible range of system movement and load transfer time parameters. All regression models showed a very good fit to the simulation results with high values of adjusted R-Square ($R_{adj}^2 > 0.99$) and satisfied regression assumptions.

The analysis showed that systems with SRM_1 achieved 22.2÷31.8% smaller values of $\hat{\lambda}_{out}^{max}$ in all cases compared to systems with SRM_2 . The highest impact on the average maximum retrieval request rate $\hat{\lambda}_{out}^{max}$ despite the SRM type, was observed for vertical velocity v_{vert} and load transfer time t_{tr} . The values of $\hat{\lambda}_{out}^{max}$ increased more rapidly for the smaller values of v_{vert} when other system parameters were fixed and the most rapid increase was observed for systems operating with routing algorithm A_3 .

The comparison of the SRM routing algorithms indicated that systems operating with routing algorithm A_2 achieved smaller values of $\hat{\lambda}_{out}^{max}$ compared to systems

operating with A_1 . Systems with SRM_1 and routing algorithm A_2 achieved only $1.39 \pm 0.7\%$ smaller values on average of $\hat{\lambda}_{out}^{max}$ compared to A_1 and $2.38 \pm 0.46\%$ smaller values for systems with SRM_2 . Routing algorithm A_3 reduces the $\hat{\lambda}_{out}^{max}$ by $9.96 \pm 2.64\%$ for systems with SRM_1 and $7.57 \pm 1.93\%$ for systems with SRM_2 compared to A_1 . The recommendation of this research is to use the new algorithm A_2 for the SRM routing, because it reduces $\hat{\lambda}_{out}^{max}$ only by $1.4 \div 2.4\%$ on average, but requires $(N_L)^{N_{LHD}-1}$ times less routing scenarios to evaluate when planning the SRM tour compared to optimal algorithm A_1 .

The C-AS/RS analysis presented in this research showed the significance of the considered system parameters to the maximum retrieval request rate and indicated that further investigations on the C-AS/RS are required, considering various combinations of I/O layouts and number of LHDs on the SRM, and more advanced routing algorithms with extended tour planning, evaluating more than one tour in advance.

Acknowledgements

This work has been supported by the European Social Fund within the project ‘Development and application of innovative research methods and solutions for traffic structures, vehicles and their flows’, project code VP1-3.1-ŠMM-08-K-01-020.

References

- Andriansyah, R.; De Koning, W. W. H.; Jordan, R. M. E.; Etman, L. F. P.; Rooda, J. E. 2011. A process algebra based simulation model of a miniload-workstation order picking system, *Computers in Industry* 62(3): 292–300. <http://dx.doi.org/10.1016/j.compind.2010.09.005>
- Carlo, H. J.; Vis, F. A. V. 2012. Sequencing dynamic storage systems with multiple lifts and shuttles, *International Journal of Production Economics* 140(2): 844–853. <http://dx.doi.org/10.1016/j.ijpe.2012.06.035>
- Ekren, B. Y.; Heragu, S. S.; Krishnamurthy, A.; Malmborg, C. J. 2010. Simulation based experimental design to identify factors affecting performance of AVS/RS, *Computers & Industrial Engineering* 58(1): 175–185. <http://dx.doi.org/10.1016/j.cie.2009.10.004>
- Farahani, R. Z.; Tari, F. G. 2002. A branch-and-bound method for finding flow-path designing of AGV systems, *International Journal of Engineering* 15(1): 81–90.
- Foley, R. D.; Frazelle, E. H.; Park, B. C. 2002. Throughput bounds for miniload automated storage/retrieval systems, *IIE Transactions* 34(10): 915–920. <http://dx.doi.org/10.1080/07408170208928922>
- Gagliardi, J.-P.; Renaud, J.; Ruiz, A. 2014a. On sequencing policies for unit-load automated storage and retrieval systems, *International Journal of Production Research* 52(4): 1090–1099. <http://dx.doi.org/10.1080/00207543.2013.838331>
- Gagliardi, J.-P.; Renaud, J.; Ruiz, A. 2014b. A simulation modeling framework for multiple-aisle automated storage and retrieval systems, *Journal of Intelligent Manufacturing* 25(1): 193–207. <http://dx.doi.org/10.1007/s10845-012-0686-x>
- Gu, J.; Goetschalckx, M.; McGinnis, L. F. 2010. Research on warehouse design and performance evaluation: a comprehensive review, *European Journal of Operation Research* 203(3): 539–549. <http://dx.doi.org/10.1016/j.ejor.2009.07.031>
- Hu, Y.-H.; Huang, S. Y.; Chen, C.; Hsu, W.-J.; Toh, A. C.; Loh, C. K.; Song, T. 2005. Travel time analysis of a new automated storage and retrieval system, *Computers & Operations Research* 32(6): 1515–1544. <http://dx.doi.org/10.1016/j.cor.2003.11.020>
- Janilionis, V.; Bazaras, Ž. 2013. The design and investigation of two storage/retrieval mechanisms of the cylindrical automated storage and retrieval system, *Transport* 28(4): 352–360. <http://dx.doi.org/10.3846/16484142.2013.864700>
- Janilionis, V.; Bazaras, Ž. 2012a. Impact of input/output position locations to the performance of C-AS/RS with autonomous load handling devices, in *Proceedings of 7th International Conference Intelligent Technologies in Logistics and Mechatronics Systems*, 3–4 May 2012, Panevėžys, Lithuania, 86–90.
- Janilionis, V.; Bazaras, Ž. 2012b. Performance analysis of the cylindrical AS/RS with multi load handling devices, in *Proceedings of IV International Conference Transport Problems*, 27–29 June 2012, Katowice–Slemien, Poland, 231–237.
- LeBaron, T.; Jacobsen, C. 2007. The simulation power of AutoMod, in *2007 Winter Simulation Conference (WSC)*, 9–12 December 2007, Washington, USA, 210–218. <http://dx.doi.org/10.1109/WSC.2007.4419603>
- Lerher, T.; Edl, M.; Rosi, B. 2014. Energy efficiency model for the mini-load automated storage and retrieval systems, *The International Journal of Advanced Manufacturing Technology* 70(1–4): 97–115. <http://dx.doi.org/10.1007/s00170-013-5253-x>
- Lerher, T.; Šraml, M.; Potrč, I. 2011. Simulation analysis of mini-load multi-shuttle automated storage and retrieval systems, *The International Journal of Advanced Manufacturing Technology* 54(1–4): 337–348. <http://dx.doi.org/10.1007/s00170-010-2916-8>
- Meller, R. D.; Mungwattana, A. 2005. AS/RS dwell-point strategy selection at high system utilization: a simulation study to investigate the magnitude of the benefit, *International Journal of Production Research* 43(24): 5217–5227. <http://dx.doi.org/10.1080/00207540500215617>
- Khattree, R.; Naik, D. N. 2000. *Applied Multivariate Statistics with SAS Software*. 2nd edition. SAS Press/John Wiley and Sons. 152 p.
- Roodbergen, K. J.; Vis, I. F. A. 2009. A survey of literature on automated storage and retrieval systems, *European Journal of Operational Research* 194(2): 343–362. <http://dx.doi.org/10.1016/j.ejor.2008.01.038>
- Sari, Z.; Grasman, S. E.; Ghouali, N. 2007. Impact of pickup/delivery stations and restoring conveyor locations on retrieval time models of flow-rack automated storage and retrieval systems, *Production Planning & Control: The Management of Operations* 18(2): 105–116. <http://dx.doi.org/10.1080/09537280600909494>
- Zaerpour, N.; De Koster, R. B. M.; Yu, Y. 2013. Storage policies and optimal shape of a storage system, *International Journal of Production Research* 51(23–24): 6891–6899. <http://dx.doi.org/10.1080/00207543.2013.774502>
- Yin, Y.-L.; Rau, H. 2006. Dynamic selection of sequencing rules for a class-based unit-load automated storage and retrieval system, *The International Journal of Advanced Manufacturing Technology* 29(11–12): 1259–1266. <http://dx.doi.org/10.1007/s00170-005-0005-1>