

Joint Retrieval Position Selection and Picking Station Assignment for Order Picking in Autonomous Cube Storage Systems

Aya Ounissi
Sruthi Mangala Suresh,
Robert Schulz,
Funing Li

*Institute of Mechanical Handling and Logistics
University of Stuttgart*

Autonomous cube storage systems require coordinated decisions on where requested items should be retrieved and to which picking station an order should be assigned. This paper proposes an integrated decision approach that jointly considers retrieval position selection and picking station assignment based on transport effort, station congestion, and vertical access effort. The approach is evaluated in a discrete-event simulation against two sequential strategies. Under baseline conditions, the integrated strategy reduces mean order processing time and increases throughput, outperforming both reference strategies. The performance advantage persists and increases with system scale, reaching 5.1% lower processing time and 5.2% higher throughput in the largest scenario. The results show that these gains are associated with lower realized robot travel time and higher station utilization, while differences in vertical digging effort are comparatively small. However, the integrated strategy also produces a more uneven distribution of stack heights across the grid, revealing a trade-off between immediate operational efficiency and the longer-term spatial configuration of inventory. Overall, the findings indicate that jointly considering retrieval positions and station assignment improves efficiency and scalability, while pointing to grid-configuration effects that merit further investigation over longer operating horizons.

[autonomous cube storage; retrieval position selection; picking station assignment; integrated optimization; order picking]

1 INTRODUCTION

1.1 BACKGROUND AND PROBLEM CONTEXT

Autonomous cube storage systems represent one of the most significant contemporary solutions for high-density, robot-assisted warehousing. These systems combine high space efficiency with scalable material provisioning and are therefore deployed particularly in e-commerce, spare

parts, and small parts environments. Characteristic features include a grid-based storage layout and vertically stacked bins [1].

Material transport within the system is performed by autonomous robots operating on the storage surface. These robots possess both transport and lifting capabilities, enabling them to selectively pick up, move, and lower individual bins. Customer orders are assigned to one or more robots, each of which retrieves the appropriate bins. If the required bins are not located at the topmost position of a stack, the overlying bins must first be temporarily relocated to gain access to the target bin. Subsequently, the robots transport the relevant bins to a picking station (port), typically located at the lower section of the system and operated by human workers. Upon completion of the picking process, the bins are returned to the storage system by the robots.

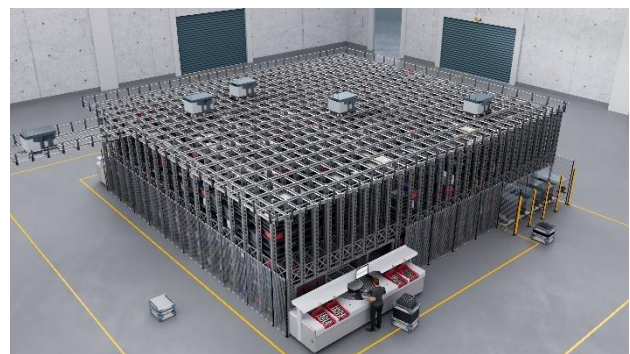


Figure 1: Example of an autonomous cube storage system (Source: Gebhardt Group, VersaStore)

The retrieval of bins by the robots constitutes a complex decision problem. On the one hand, a customer order typically comprises multiple items, each of which may be available in multiple storage bins located at different stacks and depth positions within the system. On the other hand, picking in many real-world applications is subject to order consolidation constraints, meaning that every item belonging to a given customer order must be routed to one and the

same picking station. Against this background, fulfilling a customer order gives rise to two central decision problems:

1. Selection of appropriate bins, i.e., retrieval positions: For each requested item, a decision must be made regarding which specific bin the item should be retrieved from. This decision significantly influences the number of required relocations, access times, and overall system efficiency. If an item is stored in only one bin, its retrieval position is predetermined and no selection decision is required.
2. Single-station order assignment under order-consolidation constraints: Additionally, it must be determined to which picking station the respective order as a whole is assigned. This decision affects, among other factors, the robots' travel distances, station utilization, and order processing times.

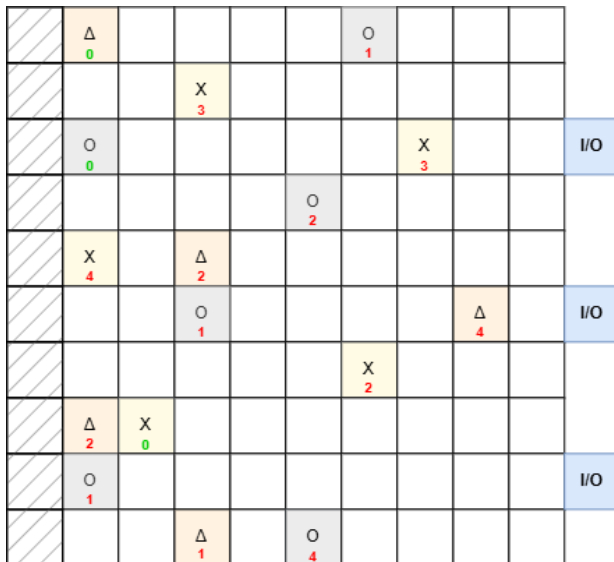


Figure 2: Decision problem in an autonomous cube storage system ($10 \times 10 \times 5$, top view) with three Input/Output stations for an order of three items (X, O, Δ). Multiple spatially distributed retrieval locations are available for each item. Numbers denote retrieval depth, defined as the number of blocking bins above the target bin.

These two decisions are mutually dependent: on the one hand, the cost of a given retrieval position for a particular item depends, among other factors, on its distance to the assigned station; on the other hand, the determination of the optimal station for the order is influenced by the spatial distribution of the required bins. This gives rise to an integrated optimization problem, in which a picking station and a specific retrieval position for each item must be determined jointly for a given customer order. The objective is to minimize the operational effort associated with fulfilling the order, with the aim of improving system throughput.

1.2 LITERATURE REVIEW

Autonomous cube storage systems are classified as robotic compact storage and retrieval systems (RCS/RS) [2]. Existing work in this area has primarily addressed reducing digging effort through inventory organization and retrieval scheduling [3], optimizing system-level scheduling [4] and storage [5] policies, and evaluating the system's performance [6-9]. Comprehensive surveys position cube-based systems within the broader evolution of robotic goods-to-person warehousing [10], while discrete-event simulation is used to quantify the effect of robot fleet size, port count, and grid dimensions on system throughput [11,12] and to evaluate storage policies [13].

The dependence of retrieval effort on stack depth and the number of blocking items is established in the block relocation problem (BRP), also referred to as the container relocation problem and defined as Bins Relocation Problem for AutoStore (BRP-AS) in [14], which minimizes the number of relocations required to retrieve a set of stacked items in a specified order and is proven NP-hard [15], [16]. Exact branch-and-bound and integer-programming methods are developed for small and medium instances [17], and heuristic and learning-based methods are proposed for larger instances [18], [19]; a survey consolidates solution methods for the BRP and the related pre-marshalling problem [20].

Order-to-station assignment is established as a planning problem distinct from storage assignment, batching, and routing, and is classified as such in the state-of-the-art review of order-picking planning problems by Van Gils et al. [21]. Workload imbalance across pickers and time slots is shown to materially affect picker idle time and throughput, motivating mixed-integer and local-search formulations of the order-picking workload-balancing problem [22], and comparative evaluations of balancing measures such as range-based and Rawlsian objectives [23]. In robotic mobile fulfillment systems (RMFS), order-and-rack assignment to multiple stations is formulated with explicit workload-balancing terms in the objective function [24]; this is extended to joint storage assignment and order and rack sequencing [25]. Jiao et al. jointly optimize order-to-station assignment and pod selection [26], while Valle and Beasley formulate a two-stage optimization model that simultaneously allocates customer orders and mobile racks to picking stations [27]. Both studies demonstrate that integrated assignment strategies outperform sequential approaches. Assignment of orders and pods to stations in multi-level systems is treated as a multi-criteria problem balancing picker workload, order-consolidation effort, and pod-movement effort simultaneously [28]. Zhen et al. formulate an integrated scheduling framework for RMFS that simultaneously considers order batching, workstation assignment, pod allocation, robot scheduling, and replenish-

ment decisions [29]. Their computational results demonstrate that coordinated planning consistently outperforms sequential optimization approaches.

The literature establishes station workload as a necessary cost component of order assignment, and several RMFS studies go further by making pod-to-station assignment itself a joint decision variable alongside order assignment [24–29], including cases where pod selection and rack allocation are optimized simultaneously with order-to-station assignment [26, 27]. However, these approaches mainly consider rack-based RMFS environments, where mobile racks or pods are transported as complete units, and do not address the unique characteristics of autonomous cube storage systems, particularly vertically stacked inventory and retrieval-dependent relocation effort. Because items are picked from whichever pod or rack already holds them, an item's retrieval position is always inherited from a prior storage/pod-membership decision rather than chosen independently, item by item, jointly with the station decision. Even the most integrated formulations therefore optimize *which pod goes to which station*, not *where within the system a given item should be retrieved from*. To the best of the authors' knowledge, the integrated selection of item retrieval positions and picking-station assignment has not been explicitly addressed for autonomous cube storage systems.

We address this gap in three steps. First, we formulate a tractable order-level surrogate objective that minimizes the estimated operational effort of fulfilling an order. We show that, under additive cost assumptions, the resulting model for joint retrieval-position selection and station assignment can be solved exactly with linear rather than exponential computational effort (Sections 2 and 3). Second, we use discrete-event simulation to evaluate whether this local decision rule translates into system-level throughput gains when traffic, queuing, and robot interaction effects partially violate the underlying assumptions. We find that its performance advantage over sequential strategies persists and even widens with system scale (Section 5). Finally, we show that this myopic, per-order cost minimization is not without cost: while it improves throughput, it also alters the spatial structure of the stored inventory, revealing a trade-off between immediate operational efficiency and long-term storage configuration (Section 6).

2 MATHEMATICAL FORMALIZATION OF THE PROBLEM

The decision problem under consideration exhibits structural similarities to classical facility location problems, in which demand elements are assigned to a selected facility at minimal cost and which are known to be NP-hard [30]. In the present context, the selection of a picking station corresponds to such an assignment decision.

In contrast to classical formulations, however, the demand elements (items) cannot be considered independently of one another, because all items of an order must be assigned to the same picking station. This creates a coupling of the assignment decisions within an order, whereas in the classical facility location problem, each customer can be assigned to a facility independently.

Furthermore, each demand element is not characterized by a fixed position but rather possesses multiple candidate retrieval positions. Thus, in addition to the station assignment, a specific retrieval position must be selected for each item.

The general structure of the joint decision problem for an order o can be formulated as:

$$\min \sum_{i \in I_o} \sum_{p \in P_i} \sum_{s \in S} c_{ps} \times x_{ips} \quad (1)$$

where:

I_o : the set of all items i contained in Order o

P_i : the set of all currently feasible retrieval positions p at which item i is available. Positions containing the requested item but already reserved for other orders are excluded. If $(|P_i| = 1)$, the retrieval position is predetermined.

S : the set of picking stations s

c_{ps} : a generic cost associated with retrieving an item from position p and assigning it to picking station s .

$$x_{ips} = \begin{cases} 1, & \text{if item } i \text{ is retrieved from position } p \text{ and routed to station } s \\ 0, & \text{otherwise} \end{cases}$$

Subject to:

$$\sum_{p \in P_i} \sum_{s \in S} x_{ips} = 1, \forall i \in I_o \quad (2)$$

$$\sum_{p \in P_i} x_{ips} = \sum_{p' \in P_j} x_{jp's}, \forall s \in S, \forall i, j \in I_o \quad (3)$$

The first constraint (2) ensures that every item is retrieved from exactly one position and routed to exactly one station. The second constraint (3) enforces that all items of order o are routed to the same station.

If the items were considered in isolation, the size of the solution space L_o for an order o would be given by:

$$L_o = |S| \cdot \prod_{i \in I_o} |P_i| \quad (4)$$

While the solution space L_o is exponential, the problem is jointly coupled only through the picking station decision. Under the additive cost assumptions defined below, the retrieval-position decisions for individual items become independent and separable once a candidate station is fixed. This structural separability avoids enumerating the exponential Cartesian product and results in a computational effort of:

$$\mathcal{O}(|S| \cdot |I_o| \cdot |P_{max}|), \text{ with } |P_{max}| = \max_{i \in I_o} |P_i| \quad (5)$$

which scales linearly in each of the three problem dimensions when the others are held fixed. To legitimately justify this, the following scientific assumptions are made:

- (1) At each retrieval position $p \in P_i$, the requested item i is assumed to be available in sufficient quantity.
- (2) Each physical bin (or retrieval position $p \in P_i$) contains exactly one stock keeping unit (SKU) or item type i . Mixed bins are excluded.
- (3) Dynamic traffic effects and interactions between robots, such as path contention, queuing at shared cells, or yielding, are not explicitly captured in the transport cost. The calculated transport costs thus represent a lower bound on the travel time actually realized in a system with multiple concurrently operating robots.
- (4) Interactions between items within the same order are not explicitly considered, although they may occur in real systems. In particular, it may happen that multiple requested items are located in the same stack, and the access effort for one item changes due to the retrieval of another.

Under these assumptions, the objective function is additively separable. Consequently, the joint optimization reduces to finding the minimum cost position for each item relative to each station, followed by selecting the station that yields the lowest aggregate cost.

3 SOLUTION APPROACH

3.1 INTEGRATED DECISION ALGORITHM

Consider a single customer order o consisting of several items, with I_o denoting the order's item set. Each item $i \in I_o$ may have one or more currently feasible retrieval positions, denoted by P_i . A set of available picking stations S is likewise given. The objective is to find a combination of:

- a picking station $s \in S$ for the entire order, and
- a retrieval position $p_i \in P_i$ for each item $i \in I_o$.

The objective function consists in minimizing the expected retrieval effort associated with fulfilling an order, comprising transport, congestion, and digging costs. By minimizing these individual effort components at the decision instant, the approach aims to increase the overall rate of order completions.

Under the adopted model assumption of additive item-related costs, the total cost for fulfilling order o at station s is defined:

$$C(o, s) = c^{cong}(s) + \sum_{i \in I_o} \min_{p \in P_i} [c^{move}(p, s) + c^{dig}(p)] \quad (6)$$

where:

$c^{cong}(s)$: the station congestion cost, equivalent of the open workload currently queued at station (Section 3.2.1)

$c^{move}(p, s)$: the approximated transport cost between retrieval position and picking station (Section 3.2.2)

$c^{dig}(p)$: the vertical access effort at position p due to overlying bins (Section 3.2.3)

The optimal station decision is then:

$$s^* = \arg \min_{s \in S} C(o, s) \quad (7)$$

and the corresponding position-related decision for each $i \in I_o$ is:

$$p_i^*(s^*) = \arg \min_{p \in P_i} [c^{move}(p, s^*) + c^{dig}(p)] \quad (8)$$

The overall approach thus first determines the cost-minimal retrieval positions for each admissible station and subsequently selects the station-position combination with minimal total cost.

For each order, the algorithm proceeds as follows:

1. For each admissible picking station, all feasible retrieval positions of every requested item are evaluated using the cost function.
2. For each item, the cost-minimizing retrieval position to the considered station is selected, and the corresponding cost is stored.
3. The minimum costs of all requested items are aggregated to obtain the total cost of assigning the order to the considered station.
4. The station with the lowest total cost is selected, together with the corresponding retrieval positions.

Algorithm 1: Integrated Station Assignment and Retrieval Position Selection

```

1:  for each  $o \in O$  do
2:    for each  $s \in S$  do
3:       $C(o, s) = c^{cong}(s)$ 
4:      for each  $i \in I_o$  do
5:         $p_i^*(s) = \arg \min_{p \in P_i} (c^{move}(p, s) + c^{dig}(p))$ 
6:         $C(o, s) = C(o, s) + c^{move}(p_i^*(s), s) + c^{dig}(p_i^*(s))$ 
7:      end for
8:    end for
9:     $s^* = \arg \min_{s \in S} C(o, s)$  and  $\{p_i^*(s^*) | i \in I_o\}$ 
10:  end for

```

Under the adopted model assumptions introduced in Section 2, the approach yields an exact optimum of the simplified surrogate optimization model. With respect to the real system, however, the algorithm possesses a heuristic character, since the underlying cost function relies on simplifying assumptions and, in particular, does not explicitly capture dynamic traffic effects, robot-to-robot interactions, or possible dependencies between items. Its practical relevance, particularly with regard to more realistic system dynamics, is therefore to be investigated in Section 4.

3.2 DERIVATION OF THE COST FUNCTION COMPONENTS

3.2.1 APPROXIMATION OF STATION CONGESTION COST

For the approximation of station congestion, only bins that are assigned to station s at the instant of decision-making and are awaiting processing (excluding the bin currently being processed) are considered. The open workload of a station s is defined as:

$$L_s = \sum_{o \in O_s} |I_o^{remaining}| \quad (9)$$

where:

O_s : the set of all orders currently assigned to station s

$I_o^{remaining}$: the set of items/bins belonging to order o that have not yet entered processing.

The station congestion cost is then given by:

$$c^{cong}(s) = L_s \cdot E[T^{pick}] \quad (10)$$

where $E[T^{pick}]$ is the expected time that the station requires to fully process a single bin

Limitations. The approximated station congestion cost does not account for whether the bins in $I_o^{remaining}$ are actually physically present at the station yet. This simplification is motivated by the fact that, in real autonomous

cube storage systems, bin provisioning and station processing partly occur in parallel. A second simplification concerns the temporal validity of the backlog itself. The underlying open workload L_s is a dynamic quantity that continuously builds up as new orders are assigned and depletes as bins are processed; the value entering $c^{cong}(s)$ therefore reflects a snapshot of this backlog taken at the decision instant, rather than the congestion level the order will actually encounter upon arrival at the station. This is considered acceptable because $c^{cong}(s)$ is used solely as a relative penalty to differentiate more congested from less congested stations at the moment of assignment, and all candidate stations are scored against the same snapshot. This provides a consistent basis for comparing all candidate stations at the decision instant. A precise prediction of the congestion level actually encountered at execution time would instead require explicitly modeling this dynamic evolution together with additional information on robot availability, traffic interactions, retrieval processes, and synchronization effects, which would substantially extend the scope of the decision problem under consideration.

3.2.2 APPROXIMATION OF TRANSPORT COST

The modelling of the transport time between a retrieval position p and a picking station s is based on classical kinematic principles of time-optimal trajectory motion [31] and provides an exact travel-time expression within the idealizing assumptions detailed below.

The spatial structure of the warehouse is described by a Manhattan metric. For a position $p = (x_p, y_p)$ and a station $s = (x_s, y_s)$, the horizontal and vertical distances to be covered are:

$$\begin{aligned} d_x(p, s) &= |x_p - x_s| \\ d_y(p, s) &= |y_p - y_s| \end{aligned} \quad (11)$$

Because the robot travels along the grid's rail system, motion along the x-axis and motion along the y-axis are kinematically independent: the robot must complete travel along one axis, then reorient its wheel sets by 90° to the perpendicular axis before it can begin travel along the other. This reorientation requires a fixed duration t_r .

Let

$$\delta(p, s) = \begin{cases} 1, & \text{if } d_x(p, s) > 0 \text{ and } d_y(p, s) > 0 \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

denote whether the path between p and s requires travel on both axes. The total transport time is given by:

$$c^{move}(p, s) = t^{axis}(d_x) + \delta(p, s) \cdot t_r + t^{axis}(d_y) \quad (13)$$

Where $t^{axis}(\cdot)$ is a single-axis travel time.

For a single-axis distance d , it is assumed that the robot accelerates and decelerates with constant acceleration a and cannot exceed a maximum speed v_{max} . The critical distance at which the maximum speed can be reached is:

$$d_{crit} = \frac{v_{max}^2}{a} \quad (14)$$

For $d < d_{crit}$, a triangular velocity profile results, in which the maximum speed is not reached. The transport time in this case is:

$$t^{axis}(d) = 2 \cdot \sqrt{\frac{d}{a}} \quad (15)$$

For $d \geq d_{crit}$, a trapezoidal velocity profile with a phase of constant speed results:

$$t^{axis}(d) = \frac{d}{v_{max}} + \frac{v_{max}}{a} \quad (16)$$

Limitations. Several assumptions underlie equations (13)-(16). First, acceleration and deceleration are modeled with a single constant a in both phases; robots frequently exhibit asymmetric braking and acceleration behavior, and the effective value of a may itself vary with load state (empty vs. carrying a bin). Second, the bang-bang switching between acceleration phases neglects jerk limits present in real controllers, which smooth these transitions and would yield slightly longer travel times than (15)/(16) predict. Third, x_p, y_p, x_s, y_s are treated as continuous variables, whereas real robot motion on the grid occurs in discrete cell increments; the continuous approximation is standard and accurate at the scale of a full grid but introduces a small discretization error at the level of individual cells. Finally, consistent with assumption (3) introduced in Section 2, the travel time computed by (13)-(16) represents an ideal, conflict-free trajectory for a single robot. Interactions with other robots are not captured, so that the resulting travel time constitutes a lower bound on the value actually realized in a multi-robot system.

3.2.3 APPROXIMATION OF ACCESS EFFORT

In addition to the horizontal transport movement discussed in the previous section, autonomous cube storage systems incur additional retrieval-related effort due to the vertical stacking of bins. Since bins are stored in vertically stacked columns without individual access points, a requested bin that is not located at the topmost position of its stack can only be retrieved after all bins located above it have been relocated.

Consequently, the retrieval effort associated with a candidate retrieval position depends on two factors: (i) the vertical depth of the target bin within its stack and (ii) the number and cost of the relocation operations required to clear access to it.

Let p denote a candidate retrieval position containing the requested item, let $B(p)$ denote the set of positions of blocking bins located above the target bin, and let $z(p)$ denote the retrieval depth, i.e. the number of stack levels between the target bin and the top of the stack. The retrieval effort associated with position p is approximated as:

$$c^{dig}(p) = c^{lift}(p) + \sum_{b \in B(p)} c^{reloc}(b) \quad (17)$$

Where $c^{lift}(p)$ denotes the vertical travel effort required to reach the target bin at depth $z(p)$, and $c^{reloc}(b)$ denotes the effort associated with relocating a single blocking bin from position b in $B(p)$.

The vertical access component of the target bin is approximated as:

$$c^{lift}(p) = 2 \times \frac{z(p) \cdot \Delta z}{v^{lift}} \quad (18)$$

Where v^{lift} denotes the vertical travel speed of the robot's lifting mechanism and Δz denotes the (uniform) height of a single storage cell, such that $z(p) \cdot \Delta z$ gives the total vertical distance to be traversed in one direction (up or down).

This approximation neglects acceleration, deceleration, and stack-entry effects (see Modeling Assumptions below) but preserves the essential property that deeper bins require proportionally longer access time.

The relocation effort of a blocking bin depends on the relocation strategy employed by the storage system. In this study, a nearest-neighbor relocation policy is assumed. Each blocking bin is temporarily moved to the nearest feasible storage position available at the decision time.

Under this assumption, the relocation effort of a blocking bin b is approximated by:

$$c^{reloc}(b) = c^{lift}(b) + 2 \cdot c^{move}(b, r(b)) + c^{lower}(r(b)) + 2 \cdot c_{handling} \quad (19)$$

Where $r(b)$ denotes the selected relocation position. The vertical efforts required to extract blocking bin from its originating position b and subsequently place it at $r(b)$, $c^{lift}(b)$ and $c^{lower}(r(b))$, are approximated analogously to (18), applied respectively to the bin's own depth within its originating stack $z(b)$ and to its new depth at the relocation position $z(r(b))$. The horizontal transport effort $c^{move}(b, r(b))$ is computed using the kinematic cost model introduced in (13-16), and is incurred twice, since after depositing a blocking bin at $r(b)$, the robot must return empty to the original stack position before it can continue, either to relocate the next blocking bin or to retrieve the target bin once the stack is cleared. $c_{handling}$ denotes the fixed handling overhead: gripper closing, lift-out, and

swing-back for extraction, and the mechanically symmetric sequence for placement, incurred twice for each blocking bin, once when it is extracted from its original position, and once when it is placed at its relocation position. The equivalent handling overhead associated with retrieving the target bin itself is deliberately excluded from (17): since exactly one such operation is required for every candidate position p , it constitutes a constant additive offset independent of p and therefore does not affect the relative ordering of candidate positions. The blocking bins' handling overhead, by contrast, cannot be omitted on the same grounds, since it scales with $|B(p)|$, which varies across candidate positions and is therefore decision-relevant.

Limitations. The retrieval-effort approximation introduced above rests on several simplifications. First, a single robot performs the entire retrieval operation, i.e. all relocations of blocking bins, followed by retrieval of the target bin, sequentially and without support from other robots. Second, the lifting mechanism is assumed to move at a constant velocity v_{lift} . Acceleration and deceleration phases of the vertical motion, as well as the reduced-speed phase typically observed when the gripper enters the stack, are neglected. As with the bang-bang assumption in the transport-time model, this yields a lower-bound approximation of the true vertical travel cost. Since the model is intended to support a *relative* comparison between retrieval positions rather than an absolute prediction of cycle time, this simplification is considered acceptable. Third, the current storage configuration is assumed to remain unchanged until execution of the retrieval operation and the relocation positions remain available. Finally, relocation operations are evaluated independently and do not alter the retrieval costs of other candidate positions.

4 EVALUATION FRAMEWORK

The evaluation serves as an out-of-model validation to assess the robustness of the proposed integrated decision logic. While the analytical model provides the decision rule under idealized conditions, a discrete-event simulation (DES) evaluates whether these decisions remain beneficial when core assumptions, such as the absence of robot interactions and traffic conflicts, are relaxed. It pursues two main objectives:

1. *quantifying the efficiency gain achieved by the integrated decision logic relative to sequential decomposition strategies under realistic operational dynamics,*
2. *analyzing the robustness of the approach to increasing system scales.*

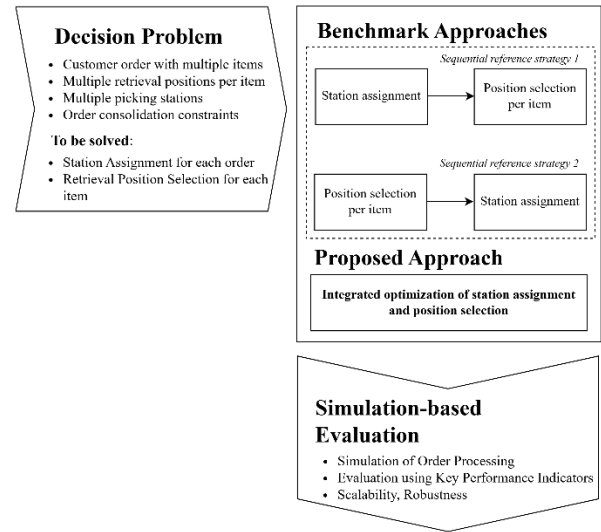


Figure 3: Overview of the experimental evaluation framework

4.1 REFERENCE STRATEGY 1: STATION-FIRST STRATEGY

This strategy corresponds to a hierarchical decomposition, in which the station decision is prioritized and the retrieval-position decision is made subsequently. First, the picking station for the entire order is selected based on the current station-related congestion. The station with minimal station congestion cost is assigned to the order:

$$s^* = \arg \min_{s \in S} c^{cong}(s) \quad (20)$$

Subsequently, for each item $i \in I_o$, the retrieval position that minimizes transport and access costs relative to the previously fixed station is chosen:

$$p_i^*(s^*) = \arg \min_{p \in P_i} [c^{move}(p, s^*) + c^{dig}(p)] \quad (21)$$

4.2 REFERENCE STRATEGY 2: POSITION-FIRST STRATEGY

The second strategy corresponds to a local greedy optimization on an item basis with subsequent global consolidation at the station level. In contrast to the first reference strategy, an independent optimization of the retrieval position selection is performed first for each item. For each item $i \in I_o$, the position p_i^* that incurs the lowest local access cost is selected:

$$p_i^* = \arg \min_{p \in P_i} c^{dig}(p) \quad (22)$$

Only afterwards is a picking station s^* determined that, considering the already fixed retrieval positions, minimizes the sum of aggregated transport costs and station congestion cost:

$$s^* = \arg \min_{s \in S} \left[c^{cong}(s) + \sum_{i \in I_o} c^{move}(p_i^*, s) \right] \quad (23)$$

4.3 EVALUATION METRICS

To evaluate and compare the integrated selection and assignment approach with the two sequential benchmark strategies, key performance indicators (KPIs) are defined. Primary KPIs capture performance improvement at the order level (processing time) and system level (throughput) (Table 1).

Table 1: Primary KPI

KPI	Definition
Mean order processing time ($\bar{T}_{proc}$)	Average elapsed time between the start of order processing and completion across all completed orders.
Throughput (TH)	Number of completed customer orders during the simulation horizon divided by the simulated operating time, expressed as completed orders per hour.

Secondary KPIs (Table 2) provide further insights into the differences between strategies in terms of horizontal transport effort, digging effort, and station congestion. Spatial indicators capture additional effects on inventory and workload distribution.

Table 2: Secondary KPIs

KPI	Definition
Mean number of relocations per order ($\bar{N}_{reloc}$)	Average count of relocation operations required to retrieve all items of an order.
Mean robot travel time per order ($\bar{T}_{travel}$)	Average cumulative horizontal travel time per order, including both retrieval and relocation movements.
Mean robot digging time per order ($\bar{T}_{dig}$)	Average cumulative vertical digging time accrued by robots in servicing an order, including both retrieval and relocation movements.
Station utilization ($U_{station}$)	Share of simulation time each picking station spends in the state busy.
Mean grid-height standard deviation ($\bar{\sigma}_{grid}$)	Average standard deviation of stack heights across all storage positions, computed from periodic samples.
Robot idle time (R_{idle})	Share of simulation time robots spend in non-productive states (yielding, waiting), aggregated across the robot fleet.

5 EXPERIMENTS AND RESULTS

5.1 SIMULATION MODEL

The experimental validation of the integrated decision approach is conducted using a discrete-event simulation model implemented in Tecnomatix Plant Simulation. The modelling aims to represent the operational dynamics of autonomous cube storage systems in order to evaluate the approach's performance under practically relevant conditions.

The simulated storage system is based on a three-dimensional grid structure with dimensions $X \times Y \times Z$, where X and Y denote the horizontal extent in grid units and Z the maximum stack height. A dedicated routing column is reserved at the edge of the grid to enable robot movements and is excluded from storage operations. The system's total capacity is therefore given by $(X - 1) \cdot Y \cdot Z = N_p$ bin storage locations. Bins are modeled with sufficient item capacity to satisfy any requested quantity in a single visit, ensuring that each SKU request is fulfilled by retrieving exactly one bin.

M picking stations are located at the perimeter of the storage system and are served by N robots. The manual picking stations are modeled as service systems with stochastic processing times. The processing time T^{pick} of a single bin is independent of the picked item quantity, representing bin handling and scanning overhead, and follows a continuous uniform distribution between 20 and 50 seconds.

$$T^{pick} \sim U(20, 50) \text{ s} \quad (24)$$

The stations process orders sequentially in the order of their arrival. Upon completion of the picking process, the bin is returned to the storage system by an available robot to the nearest feasible storage position, following the same nearest-neighbor relocation policy applied to blocking bins during retrieval (Section 3.2.3). To incorporate dual-command operations, each picking station is equipped with a buffer. Robots delivering a bin to a station can simultaneously perform a storage task by transporting a picked bin back to the storage system, provided that such a bin becomes available at the station within a predefined waiting time threshold. If no processed bin is available within this timeout period, the robot departs without a storage task.

The autonomous robots are modelled as discrete agents operating on the storage surface and performing both transport and lifting functions. Each robot possesses the capability to pick up individual bins, transport them within the x-y plane, and lower or raise them along the z-axis. Each robot occupies two adjacent grid cells and can translate along the x- and y-directions without changing its body orientation. The kinematic robot parameters are based on manufacturer data provided within the scope of the ENA-BLOCK project (No. KK5174007CI4), supported by

the Federal Ministry for Economic Affairs and Energy. For reasons of confidentiality, the underlying numerical values are not disclosed.

To ensure collision-free robot movements within the dynamic operating environment, path planning is based on the Safe Interval Path Planning (SIPP) algorithm combined with the A* search algorithm. SIPP enables the consideration of time-dependent obstacles by calculating safe time intervals for each grid cell during which occupation by other robots is excluded [32]. The resulting path planning guarantees collision-free trajectories considering the current and planned robot movements.

The storage system is initialized with a filling rate of 85%. The SKU density is set to 10% of the total grid capacity. SKU demand follows a Pareto distribution, where approximately 20% of SKUs generate around 80% of all order lines. One simulation run corresponds to a single production shift of 8 hours. Customer orders are generated in batches at the beginning of each shift. Each batch comprises n_b orders, representing the total workload assigned that shift. n_b is sized such that robots and picking stations remain continuously supplied with work throughout the 8-hour shift across all scenarios and strategies.

Each order o consists of n_o distinct SKUs, where n_o is drawn from a discrete uniform distribution over the interval $[n_{min}, n_{max}]$. A requested quantity is specified for each SKU, indicating the number of units to retrieve. Once generated, the batch is released into the simulation, and orders are processed continuously throughout the shift on a first-come, first-served basis. An integrated assignment decision is triggered as soon as an order becomes eligible for processing. At this instant, the algorithm observes the current set of feasible retrieval positions, the station backlog, and the system configuration. The selected station and retrieval positions remain fixed for the order during subsequent execution. The order is then decomposed into n_o retrieval tasks, one for each distinct SKU. Finally, these tasks are assigned sequentially to available robots.

Table 3 provides an exemplary snapshot of the task pool during a simulation run. Each row represents one retrieval task and contains the corresponding order ID, SKU ID, requested quantity, and current task status

Table 3: Exemplary snapshot of the task pool

Order ID	SKU ID	Quantity	Status
1	19	1	Picked
1	28	1	Picking
1	353	1	Picking
2	135	3	In progress
2	331	7	In progress
3	331	3	In progress
...	...	...	...

5.2 EXPERIMENTAL DESIGN

The experimental design is structured according to the evaluation objectives defined in Section 4. First, the performance of the proposed integrated decision approach is compared with the two sequential benchmark strategies. Second, the robustness and scalability of the approach is evaluated with respect to increasing system size and structural complexity. Each strategy is evaluated over 10 independent simulation replications.

To enable a valid comparison between the integrated approach and the two reference strategies, replications within each strategy used distinct random-number streams. Corresponding replications across the three strategies used the same sets of random-number streams to reduce the influence of stochastic input variation, thereby supporting a more reliable comparison of the assignment strategies. Pairwise differences were assessed using the two-sample t-tests reported by the Plant Simulation Experiment Manager, while one-way ANOVA was used to assess overall differences among the three strategies.

5.2.1 EXPERIMENT GROUP 1: BASELINE PERFORMANCE EVALUATION

The first experiment group evaluates the performance improvement achieved by the proposed integrated decision approach under a reference operating condition. The primary objective is to quantify whether the simultaneous consideration of picking station assignment and retrieval position selection leads to higher throughput compared to sequential decision-making strategies.

Table 4 summarizes the baseline parameter configuration. Each strategy is evaluated with 10 replications under this configuration, yielding 30 simulation runs for this study.

Table 4: Baseline scenario parameters

Parameter	Reference Value
Robot fleet size (N)	5
Number of picking stations (M)	5
Grid size ($X \times Y \times Z$)	$10 \times 10 \times 6$
Number of storage positions (N_p)	540
Order size [items/order]	$U\{1,2,3\}$
SKU density [%]	10
Station processing time [s]	$U(20,50)$
Fill rate [%]	85

5.2.2 EXPERIMENT GROUP 2: SCALABILITY EVALUATION

The second experiment investigates the structural scalability of the approach. To this end, the robot fleet size, the number of picking stations, and the warehouse dimensions, are scaled jointly, maintaining a constant robot-to-station ratio. Order size, SKU density, station processing time, and storage utilization (fill rate) remain fixed at their baseline values (Table 4).

Table 5 summarizes the structural scalability scenarios. Each scenario is evaluated for all three assignment strategies with 10 replications per strategy, resulting in 90 simulation runs.

Table 5: Structural scalability scenarios

Scenario	N	M	$X \times Y \times Z$	N_p
S1	6	4	$15 \times 15 \times 8$	1680
S2	9	6	$20 \times 15 \times 8$	2280
S3	12	8	$25 \times 20 \times 8$	3840

5.3 RESULTS

5.3.1 BASELINE COMPARISON

Table 6 reports the primary performance measures for the three assignment strategies. The abbreviations I, SF, and PF are used to refer to the Integrated, Station-First, and Position-First strategies, respectively.

Table 6: Primary KPI results, Experiment Group 1

KPI	I	SF	PF	Pairwise p
$\bar{T}_{proc}$ [s]	132.3	137.3	134.3	≤ 0.009
TH [order/h]	101.4	96.5	98.6	≤ 0.004

The Integrated strategy achieves the lowest mean order processing time and the highest throughput, followed by Position-First and Station-First, with pairwise Welch's two-sample t -tests indicating statistically significant differences between all three strategies (all $p < 0.01$).

Table 7 reports the secondary KPI results. The one-way ANOVA results indicate that neither the average number of relocation operations nor the average digging time differs significantly between the three assignment strategies ($p = 0.666$ and $p = 0.121$, respectively). Consequently, vertical retrieval effort contributes only marginally to the observed throughput differences under the baseline scenario.

Table 7: Secondary KPI results, Experiment Group 1

KPI	I	SF	PF	P (ANOVA)
$\bar{N}_{reloc}$	0.414	0.430	0.420	0.666
$\bar{T}_{travel}$ [s]	70.3	75.8	71.4	< 0.001
$\bar{T}_{dig}$ [s]	29.9	30.2	29.7	0.121
$U_{station}$ [%]	42.7	40.7	41.6	< 0.001
$\bar{\sigma}_{grid}$ [bins]	2.135	2.124	2.114	0.217
R_{idle} [%]	21.2	21.0	21.6	0.056

In contrast, substantial differences are observed for robot travel time. The Station-First strategy produces the highest average travel time per order (75.8 s), whereas the Integrated and Position-First strategies require only 70.3 s and 71.4 s, respectively. The one-way ANOVA confirms a statistically significant effect of the assignment strategy on travel time ($p < 0.001$), indicating that the assignment logic affects the realized horizontal movement of robots. Post-hoc pairwise comparisons further show that the difference between Integrated and Position-First is not statistically significant ($p=0.185$), whereas both outperform Station-First ($p<0.001$).

The resource-utilization measures are consistent with these findings. The Integrated strategy achieves the highest average station utilization (42.7%), followed by Position-First (41.6%) and Station-First (40.7%). The ANOVA confirms significant differences in station utilization between the strategies ($p < 0.001$). Together with the travel-time results, this indicates differences in how the assignment strategies distribute and utilize system resources during execution.

In contrast, no statistically significant strategy effects are observed for grid-height variance ($p = 0.217$), or robot idle time ($p = 0.056$).

5.3.2 SCALABILITY EVALUATION

The ordering observed at baseline (Experiment Group 1) holds unchanged across all three structural scalability scenarios, both for mean processing time (Figure 4) and throughput (Figure 5): Integrated achieves the lowest mean processing time and the highest throughput, followed by Position-First, with Station-First consistently last. Given that S1-S3 vary system size while preserving the robot-to-station ratio, this consistency indicates that the performance advantage of the integrated persists as system size increases.

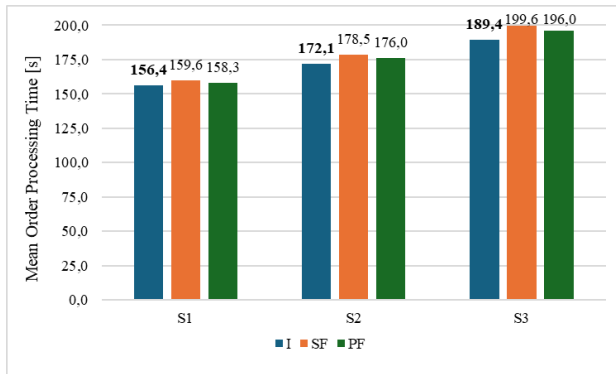


Figure 4: Mean order processing time by scenario and strategy

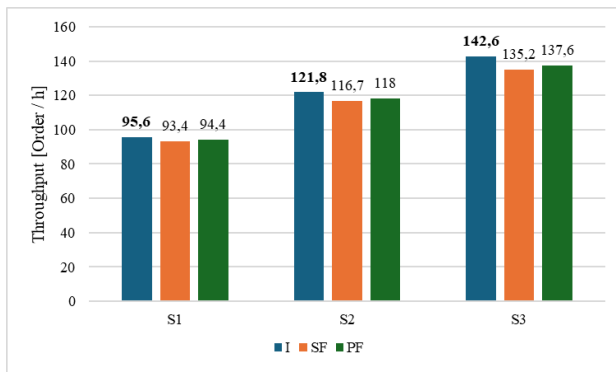


Figure 5: Throughput by scenario and strategy

The relative advantage of the integrated approach over Station-First widens as system size increases, from 2.0% (S1) to 3.6% (S2) to 5.1% (S3) in mean processing time (Figure 6), with a comparable trend in throughput (2.3%, 4.2%, 5.2%) (Figure 7). This pattern is consistent with the mechanism underlying Station-First's deficit: because the station is selected without regard to retrieval position, unfavorable spatial combinations can lead to larger transport detours as the grid expands.

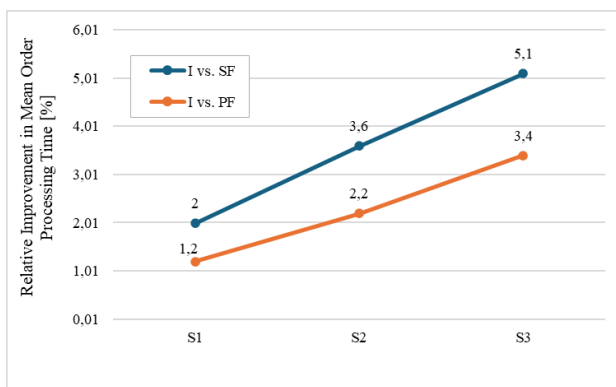


Figure 6: Relative improvement of the Integrated strategy over Station-First and Position-First in mean order processing time, by scenario

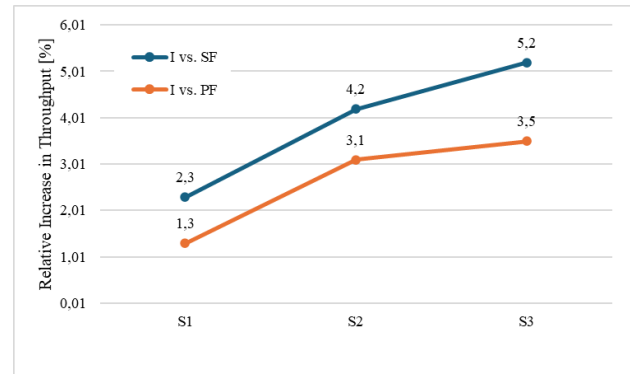


Figure 7: Relative improvement of the Integrated strategy over Station-First and Position-First in throughput, by scenario

The simulated travel time results support this interpretation: Station-First exhibits the largest increase in mean travel time from S1 to S3 (+55.9%), compared with +49.6% for Integrated and +54.1% for Position-First, and its mean robot travel time per order remains highest at every scale (Figure 8).

Mean digging time shows small but statistically significant differences between Integrated and Position-First at every scale ($p = 0.004-0.03$), with Position-First consistently lowest (Figure 9), consistent with its decision logic, which selects retrieval positions purely by digging cost (Eq. (22)). Yet, the magnitude of this difference (0.7-1.0 s, or 2–3% of mean digging time) is an order of magnitude smaller than the travel-time gap between strategies, and does not scale with system size in the same way. Thus, the observed differences in digging time are unlikely to explain the widening processing-time gap, consistent with the baseline finding that vertical retrieval effort contributes only marginally to overall performance differences. Mean relocations per order do not differ significantly between any pair of strategies at any scenario (all $p \geq 0.10$).

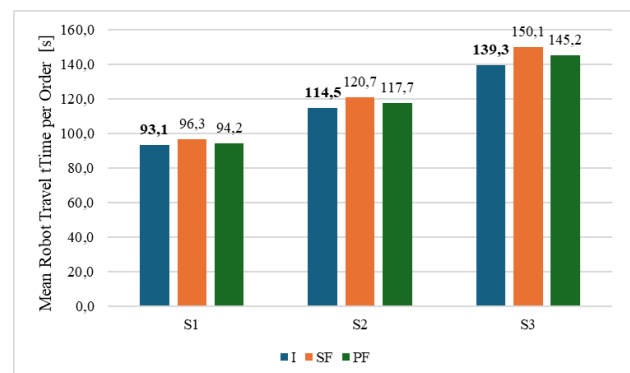


Figure 8: Mean robot travel time by scenario and strategy

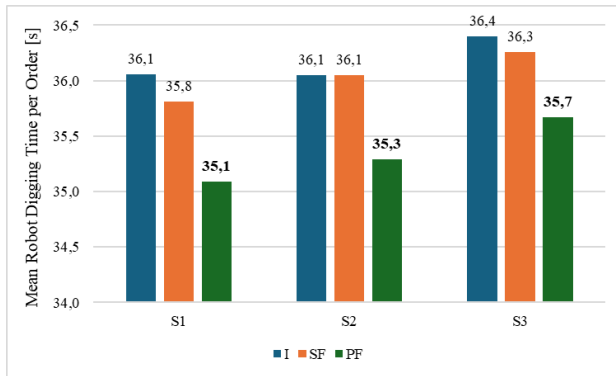


Figure 9: Mean digging time per order by scenario and strategy

Station utilization mirrors the baseline pattern: the Integrated strategy achieves the highest utilization at every scale (Figure 10), and the gap relative to both benchmark strategies is statistically significant throughout, widening further at S3 ($p < 0.001$).

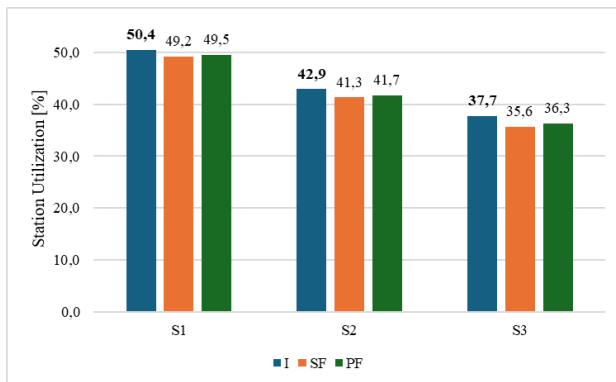


Figure 10: Station utilization by scenario and strategy

Robot idle time and grid-height standard deviation both show statistically significant, scale-growing differences between strategies. The Integrated strategy exhibits the lowest idle time (Figure 11) but the highest grid-height standard deviation (Figure 12) at every scale, suggesting that the efficiency gains of the Integrated strategy come at the cost of a more uneven vertical stack-height distribution.

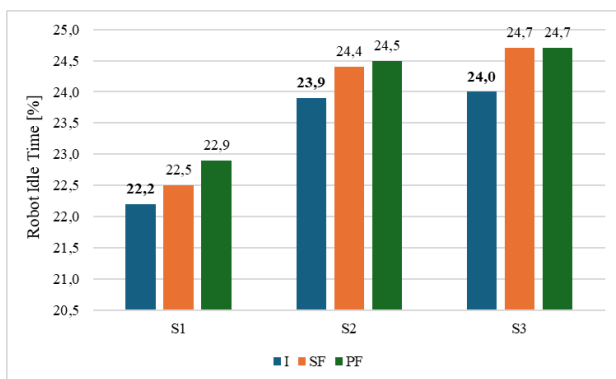


Figure 11: Robot idle time by scenario and strategy

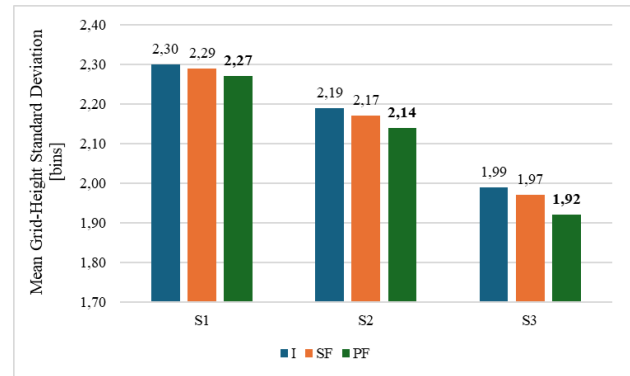


Figure 12: Mean grid-height standard deviation by scenario and strategy

6 DISCUSSION

The results support the premise that jointly deciding station assignment and retrieval position yields a consistent, statistically significant improvement in processing time and throughput over both sequential strategies, at baseline and across all scalability scenarios. The secondary KPIs provide additional evidence for interpreting these differences at the system level. In particular, the Integrated strategy generally exhibits lower realized robot travel time than Station-First, while differences in digging time and relocation counts remain comparatively small. These observations are consistent with the spatial coupling between station assignment and retrieval-position selection underlying the proposed decision logic.

The comparison between Integrated and Position-First provides further insight into the role of station workload in the decision process. Position-First and Integrated exhibit statistically indistinguishable travel times at baseline ($p = 0.185$), while Integrated achieves higher throughput and station utilization. This pattern is consistent with the fact that Position-First determines retrieval positions before considering the station assignment, whereas Integrated evaluates station assignment and retrieval positions jointly. The results therefore suggest that considering station workload together with transport and retrieval-access costs can provide an operational advantage beyond the spatial effects captured by travel time alone.

Both sequential strategies commit early to one decision without knowledge of the other, producing combinations that are locally reasonable but jointly suboptimal, while the integrated strategy can balance the trade-offs between cost components in Eq. (6) simultaneously. These findings confirm that while the cost function (Eq. (6)) serves as a surrogate for throughput, its joint application is the primary driver for the observed improvements in system efficiency and scalability.

The widening of the Integrated strategy's advantage with system size is consistent with a scale-dependent weakness specific to Station-First: because its station choice is spatially blind (Eq. (20)), the detour this imposes grows with the spatial extent of the grid, whereas the Integrated strategy can adapt its joint position-station decision to the increasing spatial extent. This performance gain is not free of trade-offs: the higher grid-height standard deviation observed for Integrated at every scale shows that its cost-minimizing logic, which ignores the resulting distribution of stack heights, alters the spatial structure of the inventory by increasing the dispersion of stack heights.

Within the simulated eight-hour horizon this does not translate into higher digging time, but whether it compounds over longer operating periods remains open.

Finally, the gains observed in the dynamic simulation persist despite the presence of traffic, robot interactions, and path-planning conflicts. The resulting performance differences are therefore obtained under system conditions that are not fully represented in the analytical cost model. This supports the practical relevance of the proposed decision logic, while also highlighting the gap between the simplified decision model and the full dynamic system.

7 CONCLUSION

This paper investigated the joint decision problem of picking station assignment and retrieval position selection in autonomous cube storage systems. In contrast to sequential decision strategies, the proposed integrated approach evaluates these decisions simultaneously by considering transport effort, station congestion, and vertical retrieval effort. A discrete-event simulation was employed as an out-of-model validation to evaluate the approach's robustness under both baseline and increasing system-scale conditions. By incorporating realistic robot interactions and dynamic path planning, the simulation specifically assessed whether the decision logic remains effective when its underlying simplifying assumptions are relaxed.

The results indicate that the decision rule derived from the analytical model is robust against the stochastic and dynamic system conditions represented in the simulation. The evaluation confirms that the integrated approach provides a significant advantage even when the simplifying assumptions required for the mathematical optimization are no longer strictly met. Under the baseline configuration, it achieved the lowest mean order processing time (132.3 s) and the highest throughput (101.4 orders/h), outperforming both the station-first and position-first strategies. The advantage persisted across all scalability scenarios and increased with system size, reaching a 5.1% reduction in mean processing time and a 5.2% increase in throughput compared with the station-first strategy in the largest scenario. The results indicate that the performance gains are accompanied by lower realized robot travel time and higher

picking-station utilization, while differences in digging effort and relocation frequency are comparatively small.

The scalability results further suggest that the integrated approach is particularly beneficial as system complexity increases. By jointly considering the spatial relationship between retrieval positions and picking stations, the approach reduces unfavorable transport patterns observed under the sequential decision rules and is associated with more effective utilization of system resources. At the same time, the results reveal a relevant trade-off: the integrated strategy's myopic retrieval-position selection produces a more uneven distribution of stack heights across the storage grid. This suggests that short-term operational optimization can inadvertently degrade the long-term spatial homogeneity of the inventory. Whether this effect accumulates over longer operating horizons and eventually increases digging effort or reduces system performance remains open.

Overall, the findings indicate that jointly optimizing retrieval position selection and picking station assignment is a promising approach for improving the operational efficiency and scalability of autonomous cube storage systems. The proposed decision logic provides an efficient alternative to sequential assignment strategies while remaining computationally tractable under the assumptions considered.

Future research should extend the approach by incorporating an explicit conflict- or traffic-density term into the transport cost and extend the congestion approximation to account for its temporal evolution. In addition, the observed effect on inventory distribution motivates the investigation of multi-objective decision models that explicitly balance immediate retrieval efficiency with long-term storage configuration. A further direction concerns the integration of learning-based decision-making. Recent work has demonstrated the potential of deep reinforcement learning for jointly optimizing retrieval and relocation decisions in multi-deep warehouse systems, including strong generalization across previously unseen warehouse configurations [33]. Building on this line of research, future work could investigate learning-based extensions of the proposed integrated station-assignment and retrieval-position selection framework. Such approaches could incorporate dynamic traffic conditions, robot availability, evolving station workloads, and changing storage configurations directly into the decision process. They could further relax the assumption in Eq. (6) that retrieval and relocation costs of items are independent and additive, by explicitly capturing cost interactions between items, both within the same order and across concurrently processed orders, thereby overcoming several of the simplifying assumptions of the present model.

LITERATURE

- [1] W. Ha, "The Characteristics of Cube Storage: AutoStore System", in *International Conference on Information and Communication Technology Convergence (ICTC)*, 2024, pp. 391–394, doi: 10.1109/ICTC62082.2024.10826700.
- [2] M. Tutam, J. Liu, and J. A. White, "Consideration of skewness in designing robotic compact storage and retrieval systems", *Expert Systems with Applications*, Vol. 248, Art. no. 123361, 2024, doi: 10.1016/j.eswa.2024.123361.
- [3] A. Yue and S. L. Smith, "Minimizing Robot Digging Times to Retrieve Bins in Robotic-Based Compact Storage and Retrieval Systems", 2023, doi: 10.48550/arXiv.2312.05338.
- [4] W. Y. Ha, "Optimizing Scheduling for Enhanced Efficiency in Cube Storage Warehouses: A Genetic Algorithm Approach Across Different Schemes", *IEEE Access*, Vol. 13, pp. 84303–84312, 2025, doi: 10.1109/ACCESS.2025.3566989.
- [5] B. Zou, M. B. M. de Koster, and X. Xu, "Evaluating Dedicated and Shared Storage Policies in Robot-Based Compact Storage and Retrieval Systems", *ERIM Report Series Reference*, 2016. Available at: <https://ssrn.com/abstract=2888649>.
- [6] P. Trost and M. Eder, "An analytical performance approach for RCS/RS with one robot serving multiple stack heights under a one-path relocation strategy", *Scientific Reports*, Vol. 14, Art. no. 3593, 2024, doi: 10.1038/s41598-024-53884-6.
- [7] P. Trost and M. Eder, "A performance calculation approach for a robotic compact storage and retrieval system (RCS/RS) serving one picking station", *Production & Manufacturing Research*, Vol. 12, No. 1, 2024, doi: 10.1080/21693277.2024.2336056.
- [8] P. Trost and M. Eder, "An analytical approach for the performance calculation of an RCS/RS with several picking stations", *International Journal of Advanced Manufacturing Technology*, Vol. 134, pp. 627–644, 2024, doi: 10.1007/s00170-024-14124-3.
- [9] P. Trost and M. Eder, "An analytical performance investigation of RCS/RS under a class-based access structure over the stack height", *Production & Manufacturing Research*, Vol. 13, No. 1, 2025, doi: 10.1080/21693277.2025.2530519.
- [10] Í. Barros and T. Nascimento, "Robotic Mobile Fulfillment Systems: A Survey on Recent Developments and Research Opportunities", *Robotics and Autonomous Systems*, Vol. 137, Art. no. 103729, 2021, doi: 10.1016/j.robot.2021.103729.
- [11] P. Trost, G. Kartnig, and M. Eder, "Simulation study of Autostore-systems", in N. Zrnic, G. Kartnig, and S. Bošnjak (Eds.), *Proceedings of the XXIV International Conference MHCL'22*, 2022, pp. 81–86, doi: 10.34726/3044.
- [12] P. Sereda, G. Kartnig, and M. Eder, "Simulation of the maximum throughput of Autostore storage systems", *Logistics Journal: Proceedings*, Vol. 2022, 2022, doi: 10.2195/lj_proc_trost_de_202211_01.
- [13] M. Beckschäfer, S. Malberg, K. Tierney, and C. Weskamp, "Simulating Storage Policies for an Automated Grid-Based Warehouse System", in T. Bektaş, S. Coniglio, A. Martinez-Sykora, and S. Voß (Eds.), *Computational Logistics. ICCL 2017. Lecture Notes in Computer Science*, Vol. 10572, Cham: Springer, 2017, doi: 10.1007/978-3-319-68496-3_31.
- [14] Y. Liu and X. Dong, "Bins relocation problem in AutoStore: an overview", in *International Conference on Algorithms, High Performance Computing, and Artificial Intelligence (AHPCAI 2024)*, Vol. 13403, Art. no. 134033R, 2024, doi: 10.1117/12.3051657.
- [15] M. Caserta, S. Schwarze, and S. Voß, "A mathematical formulation and complexity considerations for the blocks relocation problem", *European Journal of Operational Research*, Vol. 219, No. 1, pp. 96–104, 2012, doi: 10.1016/j.ejor.2011.12.039.
- [16] D.-Y. Lin, Y.-J. Lee, and Y. Lee, "The container retrieval problem with respect to relocation", *Transportation Research Part C: Emerging Technologies*, Vol. 52, pp. 132–143, 2015, doi: 10.1016/j.trc.2015.01.024.
- [17] B. Jin and S. Tanaka, "An exact algorithm for the unrestricted container relocation problem with new lower bounds and dominance rules", *European Journal of Operational Research*, Vol. 304, No. 2, pp. 494–514, 2023, doi: 10.1016/j.ejor.2022.04.006.
- [18] D. Feillet, S. N. Parragh, and F. Tricoire, "A local-search based heuristic for the unrestricted block relocation problem", *Computers & Operations Research*, Vol. 108, pp. 44–56, 2019, doi: 10.1016/j.cor.2019.04.006.
- [19] L. Liu, Y. Feng, Q. Zeng et al., "A Q-learning-based algorithm for the block relocation problem", *Journal of Heuristics*, Vol. 31, Art. no. 14, 2025, doi: 10.1007/s10732-024-09545-y.

- [20] C. Lersteau and W. Shen, "A survey of optimization methods for Block Relocation and PreMarshalling Problems", *Computers & Industrial Engineering*, Vol. 172, Art. no. 108529, 2022, doi: 10.1016/j.cie.2022.108529.
- [21] T. van Gils, K. Ramaekers, A. Caris, and R. B. M. de Koster, "Designing efficient order picking systems by combining planning problems: State-of-the-art classification and review", *European Journal of Operational Research*, Vol. 267, No. 1, pp. 1–15, 2018, doi: 10.1016/j.ejor.2017.09.002.
- [22] S. Vanheusden, T. van Gils, A. Caris, K. Ramaekers, and K. Braekers, "Operational workload balancing in manual order picking", *Computers & Industrial Engineering*, Vol. 141, Art. no. 106269, 2020, doi: 10.1016/j.cie.2020.106269.
- [23] S. Vanheusden, T. van Gils, K. Braekers, K. Ramaekers, and A. Caris, "Analysing the effectiveness of workload balancing measures in order picking operations", *International Journal of Production Research*, Vol. 60, No. 7, pp. 2126–2150, 2022, doi: 10.1080/00207543.2021.1884307.
- [24] Y. Zhuang, Y. Zhou, Y. Yuan, X. Hu, and E. Hassini, "Order picking optimization with rack-moving mobile robots and multiple workstations", *European Journal of Operational Research*, Vol. 300, No. 2, pp. 527–544, 2022, doi: 10.1016/j.ejor.2021.08.003.
- [25] Y. Zhuang, Y. Zhou, E. Hassini, Y. Yuan, and X. Hu, "Improving order picking efficiency through storage assignment optimization in robotic mobile fulfillment systems", *European Journal of Operational Research*, Vol. 316, No. 2, pp. 718–732, 2024, doi: 10.1016/j.ejor.2024.02.025.
- [26] G. Jiao, H. Li, and M. Huang, "Online joint optimization of pick order assignment and pick pod selection in robotic mobile fulfillment systems", *Computers & Industrial Engineering*, Vol. 175, Art. no. 108856, 2023, doi: 10.1016/j.cie.2022.108856.
- [27] C. A. Valle and J. E. Beasley, "A two stage approach for order and rack allocation with order backlog in a mobile rack environment", *Annals of Operations Research*, Vol. 351, pp. 167–201, 2025, doi: 10.1007/s10479-024-06058-6.
- [28] G. Tadumadze, J. Wenzel, S. Emde et al., "Assigning orders and pods to picking stations in a multi-level robotic mobile fulfillment system", *Flexible Services and Manufacturing Journal*, Vol. 35, pp. 1038–1075, 2023, doi: 10.1007/s10696-023-09491-0.
- [29] L. Zhen, Z. Tan, R. de Koster, and S. Wang, "How to Deploy Robotic Mobile Fulfillment Systems", *Transportation Science*, Vol. 57, No. 6, pp. 1671–1695, 2023, doi: 10.1287/trsc.2022.0265.
- [30] H. Xiao, J. Zhang, Z. Zhang, and W. Li, "A Survey of Approximation Algorithms for the Universal Facility Location Problem", *Mathematics*, Vol. 13, No. 7, p. 1023, 2025, doi: 10.3390/math13071023.
- [31] Bozer, Y. A., and White, J. A., "Travel-Time Models for Automated Storage/Retrieval Systems", *IIE Transactions*, 1984, doi: 10.1080/07408178408975252.
- [32] M. Phillips and M. Likhachev, "SIPP: Safe Interval Path Planning for Dynamic Environments", in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011, pp. 5628–5635, doi: 10.1109/ICRA.2011.5980306.
- [33] F. Li, J. Zhou, Y. Tian, R. Noortwyck, A. Ounissi, B. Hong, K. Wen, Q. Kang, and R. Schulz, "Deep reinforcement learning with instance-invariant baseline regularization for joint retrieval and relocation scheduling in multi-deep warehouses", *Expert Systems with Applications*, Vol. 328, 2026, Art. no. 132893, doi: 10.1016/j.eswa.2026.132893.

Aya Ounissi, M.Sc., Head of the Logistics Processes Department at the Institute of Mechanical Handling and Logistics, University of Stuttgart.

Phone: +49 711 685 83935

E-Mail: aya.ounissi@ift.uni-stuttgart.de

Sruthi Mangala Suresh, B.Sc., Student Assistant at the Institute of Mechanical Handling and Logistics, University of Stuttgart.

E-Mail: sruthi.mangala-suresh@ift.uni-stuttgart.de

Univ.-Prof. Dr.-Ing. Robert Schulz, Head of the Institute of Mechanical Handling and Logistics, University of Stuttgart.

Phone: +49 711 685 83771

E-Mail: robert.schulz@ift.uni-stuttgart.de

Funing Li, M.Sc., Research Associate at the Institute of Mechanical Handling and Logistics, University of Stuttgart.

Phone: +49 711 685 83698

E-Mail: funing.li@ift.uni-stuttgart.de

Address: Institute of Mechanical Handling and Logistics, University of Stuttgart, Holzgartenstraße 15B, 70174 Stuttgart, Germany.