Robotics and Computer-Integrated Manufacturing 44 (2017) 117–128

Contents lists available at ScienceDirect

Robotics and Computer-Integrated Manufacturing

journal homepage: www.elsevier.com/locate/rcim

Design and simulation based validation of the control architecture of a stacker crane based on an innovative wire-driven robot

Bashir Salah [a,*], Omar Janeh [b], Bernd Noche [b], Tobias Bruckmann [c], Saber Darmoul [a]

[a] Industrial Engineering Department, College of Engineering, King Saud University, P.O. Box 800, 11421, Riyadh, Saudi Arabia
[b] Institute of Product Engineering, College of Engineering, University of Duisburg-Essen, Germany
[c] Chair of Mechatronics, University of Duisburg-Essen, Germany

ARTICLE INFO

Article history:
Received 18 December 2015
Received in revised form
10 August 2016
Accepted 10 August 2016

Keywords:
Automated Storage/Retrieval System
Wire-driven robot
Control architecture

ABSTRACT

Automated Storage/Retrieval Systems (AS/RS) have an important role in the improvement of the performance of automated manufacturing systems, warehouses and distribution centers. Existing AS/R systems are usually based on Cartesian Storage/Retrieval Manipulators (SRM). Such systems have reached their maximum performance due to the limitations of their underlying mechanical design and associated control architecture. Going beyond the limits of existing systems requires structural innovation and breakthrough solutions to enhance their design and performance. In this study, we introduce the design and simulation based evaluation of a stacker crane based on an innovative wire-driven SRM. We describe the basic components and provide an overview of the mechanical design of the system. We design the high-level control architecture that allows handling mini-load operations. We develop the equations that determine the single and dual command cycle times for the wire-driven SRM in case of random and class-based storage policies. We validate the suggested control architecture using a simulation software specifically developed for this purpose. We benchmark the wire-driven SRM against an equivalent Cartesian SRM. Results show that the new wire-driven SRM design and control architecture are more competitive than Cartesian SRM in terms of travel cycle times, and more suitable for buildings growing in height.

1. Introduction

Automated Storage and Retrieval Systems (AS/RS) are computer-controlled material handling systems that are used to automatically place loads into and withdraw them from defined storage locations [1]. An AS/RS implements a defined degree of automation to ensure speed and precision in performing storage and retrieval operations [2]. AS/RS were first introduced in the 1950s to eliminate (or at least reduce) drawbacks of manual material handling [1,3]. Nowadays, AS/RSs are used in many manufacturing industries, distribution centers and warehouses. They contribute to save time and cost by limiting both damage to products and non-value added labor related manipulations [4]. They have many advantages over traditional (usually manual) storage technologies, such as improving system efficiency and space utilization, increasing reliability, reducing error rates and costs (e.g. due to labor, damage, or loss by theft) [1]. AS/RS contribute to facing continuous change in a company's internal and external environments, due to several factors, including the challenges of meeting the fluctuating production volumes, reducing energy consumption, improving inventory management, tracking and traceability, etc.

1.1. Crane based mini-load AS/RS

Since the 1950's, there have been many advancements in AS/RS technology. Many types of AS/R systems exist, which differ according to various options classified in [5]. Crane based mini-load AS/RS are of particular interest to us in this article. Such systems are designed to handle unit loads that generally correspond to bins, and typically consist of the following main components illustrated in Fig. 1:

- **Racks** are typically metal structures with locations that can accommodate loads (e.g., pallets, tote boxes);
- **Cranes** are the fully automated storage and retrieval machines/

* Corresponding author.
E-mail addresses: bsalah@ksu.edu.sa (B. Salah), omar.janeh@stud.uni-due.de (O. Janeh), bernd.noche@uni-due.de (B. Noche), tobias.bruckmann@uni-due.de (T. Bruckmann), sdarmoul@ksu.edu.sa, darmoulsaber@gmail.com (S. Darmoul).

http://dx.doi.org/10.1016/j.rcim.2016.08.010
0736-5845/

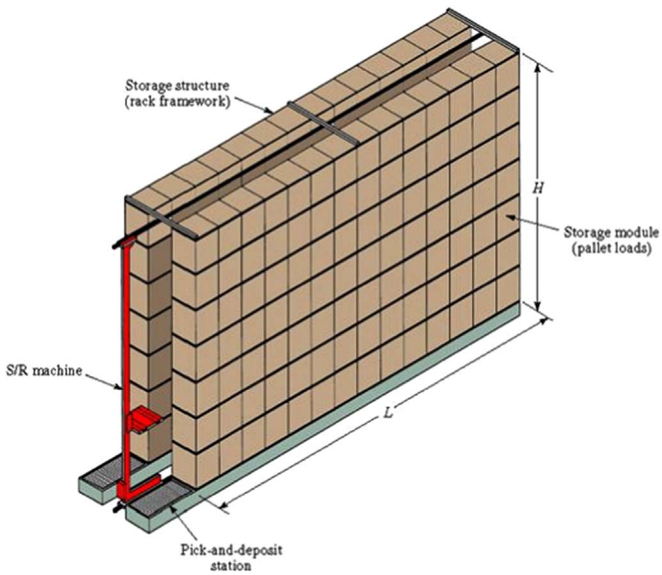

Fig. 1. Structure of crane based AS/RS [1].

manipulators (SRM) that can autonomously move, pick up and drop off loads;

- **Aisles** are formed by the empty spaces between the racks, where the cranes can move;
- **An input/output point** or station (I/O-point) is a location where retrieved loads are dropped off, and where incoming loads are picked up for storage.
- **Pick positions** (if any) are places where people are working to remove individual items from a retrieved load before the load is sent back into the system.
- **Management and control system**, which is a computer software used to manage space, track inventory and monitor SRM movements.

Much research has been devoted to designing and controlling crane based AS/R systems in order to improve precision and speed in performing storage and retrieval operations [2,5,6]. Many references discussed issues related to the design and operation of AS/R systems in order to optimize their performance [5,7–9]. These issues include concerns about sizing the AS/RS, i.e. how to determine settings and trade-offs related to the number of racks, their length and width, the number and dimensions of storage compartments, their capacity (single or double deep), the number of loads transported by the crane (single or dual shuttle), and the number of cranes. These issues also include concerns about management strategies, such as storage assignment policies, batching, dwell point location (i.e. location of the crane when it is idle), and sequencing of storage and retrieval requests. However, although questioning the crane design and motorization would allow significant improvement of performance, very few works considered the limitations due to the mechatronic design and actuation of the crane itself.

1.2. *Limitations of Cartesian S/R machines*

Existing mini-load AS/RS cranes are usually based on a Cartesian robot that has three separate mechanical drives:

- A vertical drive, which raises and lowers the mini load;
- A horizontal drive, which moves the mini load back-and-forth along the aisle;
- A shuttle drive, which transfers the mini load between the S/R machine carriage and both sides of the aisle.

These drives are often operated separately to transfer a load between any two points (for example from an I/O station to a storage location, or between storage and retrieval locations). The mast of the S/R machine moves horizontally along the aisle, while a shuttle moves vertically up and down the mast. Such separate and simultaneous drive operation introduces limitations on speed and overall performance [10]. Although cranes travel vertically and horizontally simultaneously, the actual travel time equals the maximum of the horizontal and vertical travel time (Chebyshev distance metric) [5].

In Cartesian AS/RS technologies, the maximum possible throughput is achieved when the shape of the system is considered to be L-square (the time to reach the furthest location on x-axis is equal to the time to reach the furthest location on y-axis) [11,12]. Although a good balance between rack height and length can help in reducing travel times, this solution is still insufficient to improve travel times significantly. Another limitation is due to the insufficient mass ratio between the transport device and the transported loads. The reason behind this poor mass ratio is mainly related to the serial structure of the storage and retrieval device [18]. It is crucial to design an AS/RS in such a way that it can efficiently handle current and future demand requirements while avoiding bottlenecks and overcapacity. Due to the inflexibility of the physical layout and the equipment, it is essential to design it right at once [5].

1.3. *Wire-driven robots: a promising technology*

In the last two decades, a considerable amount of research has focused on developing wire- (also called cable-) driven manipulators, which are parallel manipulators that adopt flexible cables instead of rigid limbs [14]. Wire-driven parallel manipulators possess some intrinsic advantages over rigid parallel manipulators, such as simple structure, large workspace, high load-weight ratio, and good dynamic performance. Accordingly, wire-driven parallel manipulators have been increasingly and widely used in applications, such as astronomical observation, structure building device, rescue, service or rehabilitation, and multiple aerial robots [14]. In the case of AS/RS, some authors [15,16] recently investigated breakthrough solutions that take advantage of the special properties of wire-driven manipulators to realize revolutionary storage and retrieval machines for high racks. Our article builds on the work by [15]. More particularly, we provide a high-level description of the basic components and the mechatronic design of an innovative stacker crane based on a wire-driven storage and retrieval machine. We design the control architecture that allows handling mini-load operations based on a parallel, wire-driven manipulator. We validate the suggested control architecture using a simulation software specifically developed for this purpose.

1.4. *Simulation and performance evaluation of AS/RS*

Simulation comprises an indispensable set of technological tools and methods for the successful implementation of digital manufacturing, since it allows for the experimentation and validation of product, process and system design and configuration [17]. In literature, most authors focus on the evaluation of the performance of single aisle AS/RSs with one I/O-point. Travel time estimates for both single and dual command cycles in different types of AS/RS configurations are an appropriate analytical tool for comparing control rules and storage assignment policies. Simulation enables performing more extensive experiments under various stochastic conditions. To evaluate the design and control rules of an AS/RS, several performance measures are reviewed in [5]. However, as we suggest an innovative wire-driven S/R machine, existing formulas are not suitable to evaluate the performance of our assignment policies, since the considered AS/RS are generally based on Cartesian manipulators. Therefore, one of the contributions of this article is to suggest formulas to evaluate the performance of the wire-driven S/R machine based on single and dual command cycle times and two different storage assignment policies. These formulas will be evaluated using simulation.

1.5. Problem statement

Cartesian Storage/Retrieval Manipulators (SRM) have reached their maximum performance due to limitations inherent to their underlying mechanical design and associated control architecture. Limitations are due to maximum speed restrictions and to insufficient mass ratio between the transport device and the transported loads. The reason behind this poor mass ratio is mainly related to the serial structure of the storage and retrieval device [18]. Going beyond the limits of existing systems requires structural innovation and breakthrough solutions to enhance their design and performance. In this article, we discuss a new AS/RS design and control architecture based on a wire-driven SRM inspired by the kinematic chain of a Stewart-Gough platform (SGP). A wire-driven SRM improves the mass ratio, thus contributing to reduce the energy consumption and allowing a faster movement, which are valuable advantages for most logistic processes. More particularly, we provide a high-level description of the basic components and the mechatronic design of the AS/RS (Section 2). We design the control architecture that allows handling mini-load operations, and we develop the equations that determine the single and dual command cycle times for the wire-driven SRM in case of random and class-based storage policies (Section 3). We validate the suggested control architecture using a simulation software specifically developed for this purpose, and we benchmark the wire-driven SRM against an equivalent Cartesian SRM (Section 4). We show that the new wire-driven SRM design and control architecture are more competitive than Cartesian SRM in terms of travel cycle times, and more suitable for buildings growing in height. We provide a discussion (Section 5) to analyze advantages and limitations of wire-driven SRM. This discussion enables us to draw several future research directions.

2. An overview of the mechanical design of the system

Gough and Whitehall [19] and Stewart [20] have introduced the Stewart-Gough platform (SGP, cf. Fig. 2) since the 1960s and used

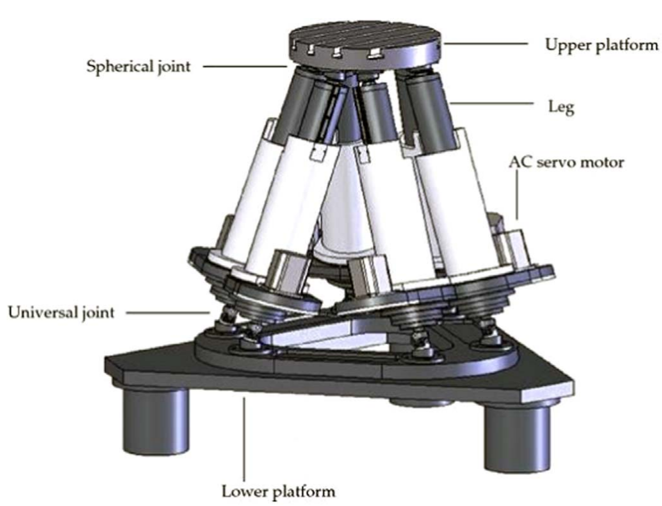

Fig. 2. A typical Stewart–Gough platform [21].

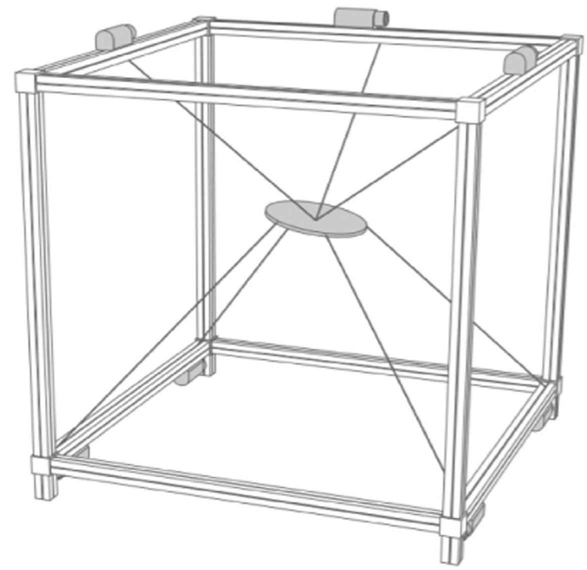

Fig. 3. A wire-driven manipulator inspired from the kinematic chain of an SGP [22].

it as a universal six degree of freedom (6-DOF) mechanism in applications, such as a tire test machine and to actuate a flight simulator.

The kinematic chain of the actuators of an SGP provides inspiration to develop a new design of the (S/R) mini-load machine. In Fig. 2, we replace the rigid piston-actuated legs by electric motor actuated wires, and we replace the upper platform by the S/R machine to obtain a wire-driven manipulator illustrated in Fig. 3. In this new design, the S/R machine is tied to cables, which are driven by electric motors at the edges of the platform. When actuated, the motors pull on the cables and make the mini-load move to the desired locations. In the general case, Fig. 3 illustrates a wire-driven platform (that corresponds to an SRM in the case of an AS/RS) that moves within a cubic storage room, where storage racks correspond to the walls (sides) of the room. Such a wire-driven SRM can fulfill up to 6-DOF [23,24]. In this article, we consider a special case of Fig. 3, restricted to only one storage rack, where the wire-driven SRM realizes 2-DOF. The proposed wire-driven AS/RS design is shown in Fig. 4.

The suggested wire-driven SRM moves in two dimensions along the storage rack. This system platform uses eight wires connected to eight electric motors, and relies on the main components and specifications provided in Table 1 and explained in more details in [15]. Although the specifications and assumptions in Table 1 are closely related to our simulations, they do not hinder the genericity of our modeling. Assumptions on wires are particularly discussed in Section 5. More details on the mechanical

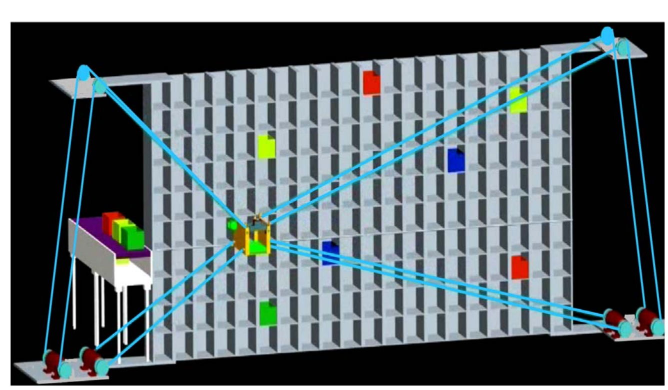

Fig. 4. Basic design of a wire-driven SRM [25].

Table 1
System specifications [15].

Property	Specifications
Drive and power	Maximum jerk 25 m s^{-3} Maximum acceleration 5 m s^{-2} Maximum velocity 6 m s^{-1} Maximum speed (revolutions per min) 1050 rpm Mass of platform including load 100 kg Maximum torque 86 N m Maximum power per drive 10 kW
Wires	Our model assumes: Linear elasticity and damping Massless wires E-module 65 GPa Diameter 5 mm Minimum wire force 100 N

design, specifications and control of the system can be found in [15,26].

The wire-driven SRM can move with higher speed and acceleration than Cartesian S/R machines, which enhances the system performance (cycle times and throughput). In addition, the proposed solution is expected to decrease the investment costs required to build the system due to the reduction of dependencies between various mechanical and electrical components. The objective of this article is to develop a control architecture and to suggest management strategies to sequence storage and retrieval requests in the suggested wire-driven AS/RS. We also suggest new performance evaluation formulas that are adapted to wire-driven manipulators. The performance of the suggested control architecture and management policies will be assessed using simulation.

3. High level control architecture

This section deals with the design and development of an architecture to control the wire-driven SRM to handle small loads that are contained in bins within the storage system. We consider single deep storage compartments and a single shuttle SRM that is able to handle only one load at a time. Fig. 5 shows an overview of the suggested control architecture.

The wire-driven SRM performs single and dual command cycles. In single command (SC) cycles, the SRM accomplishes either a storage or a retrieval operation between the I/O station and a storage location. In the case of a storage operation (respectively a retrieval operation), the SRM picks up a load from the I/O station (respectively from a storage location), travels to an empty location (respectively to I/O station) to deliver the load, drops the load off, and returns empty to the I/O station (respectively to a loaded location). In a dual command (DC) cycle, the SRM picks up a load at I/O station, travels loaded to an empty location, drops the load off in the empty location, travels empty to a loaded location in the rack, picks up the load, travels loaded to the I/O station, and drops the load off at the I/O station.

3.1. Storage and retrieval request management module

Enterprise production management systems, such as Enterprise Resource Planning (ERP) or Manufacturing Execution System (MES), provide data about confirmed and expected storage and retrieval orders. The storage and retrieval request management module relies on three modules (Select 1, 2 and 3, cf. Fig. 5), which main task is to sort the incoming orders to obtain four lists:

- **Dual command list (DCL):** is the list of orders that contain both storage and retrieval requests. The Select 1 module analyzes the list of incoming orders, and determines a sequence of orders that can be executed in dual command cycles. The problem of finding this sequence is modeled as a traveling salesman problem with multiple trips, where the objective is to minimize the total distance traveled by the SRM, and solved using a genetic algorithm [27];
- **Retrieval command list (RCL):** is the list of orders that contain only retrieval requests;
- **Storage command list (SCL):** is the list of orders that contain only storage requests;
- **Expected (Future) command list (ECL):** is the list of orders that are expected to be withdrawn in a near future, based on a user-defined time frame.

The storage and retrieval request management module gives priority to the DCL module, so that the maximum number of requests are handled in dual command cycles. As there may be some requests that cannot be served within a DCC scheme (since the SRM is single shuttle and in case for example all requests are retrieval requests or all requests are storage requests), they are handled separately within RCL or SCL lists.

3.2. Assignment policy module

The decision made at this stage is the allocation of storage locations to items. Different strategies exist to store items in, or retrieve them from AS/RS, based on turnover frequency and number of items to be stored or retrieved [7,9]. The performance of such assignment policies is generally related to Cartesian AS/RS, and is generally evaluated based on estimations of single and dual cycle times for Cartesian AS/RS [10]. However, as we suggest an innovative wire-driven S/R machine, existing formulas are not suitable to evaluate the performance of our assignment policies. Therefore, in the following, we describe the assignment strategies that we selected, and we suggest formulas to evaluate their performance based on single and dual cycle times [25]. Noteworthy, the velocity and acceleration capabilities of a real wire-driven S/R machine depend on the current platform pose, which is neglected within this study and subject to future research. Let us use the nomenclature of Table 2.

3.2.1. Random assignment

In this method, any bin is equally likely to be stored in any vacant storage location. A vacant location is randomly picked and assigned to store the bin, regardless of its turnover. In this assignment policy, single and dual command cycles are calculated as follows:

3.2.1.1. Single Command Cycle (SCC). As the wire-driven S/R machine can join any two points directly (cf. Fig. 6), the Euclidean distance can be used to calculate the expected single command cycle time (T_{SCC_R}) using Eq. (1):

$$E\left(T_{SCC_R}\right) = 2*\frac{1}{(L*H)}\int_0^H \int_0^L \frac{v}{a} + \frac{\sqrt{x^2+y^2}}{v}dxdy \tag{1}$$

3.2.1.2. Dual Command Cycle (DCC). A dual command cycle includes two movements (cf. Fig. 7): (1) one single command cycle (T_{SCC_R}, from I/O station to storage location, and then from retrieval location to I/O station); and (2) a time (T_{TB_R}) to travel from storage location to retrieval location. The expected travel time T_{TB_R} is calculated according to Eq. (2):

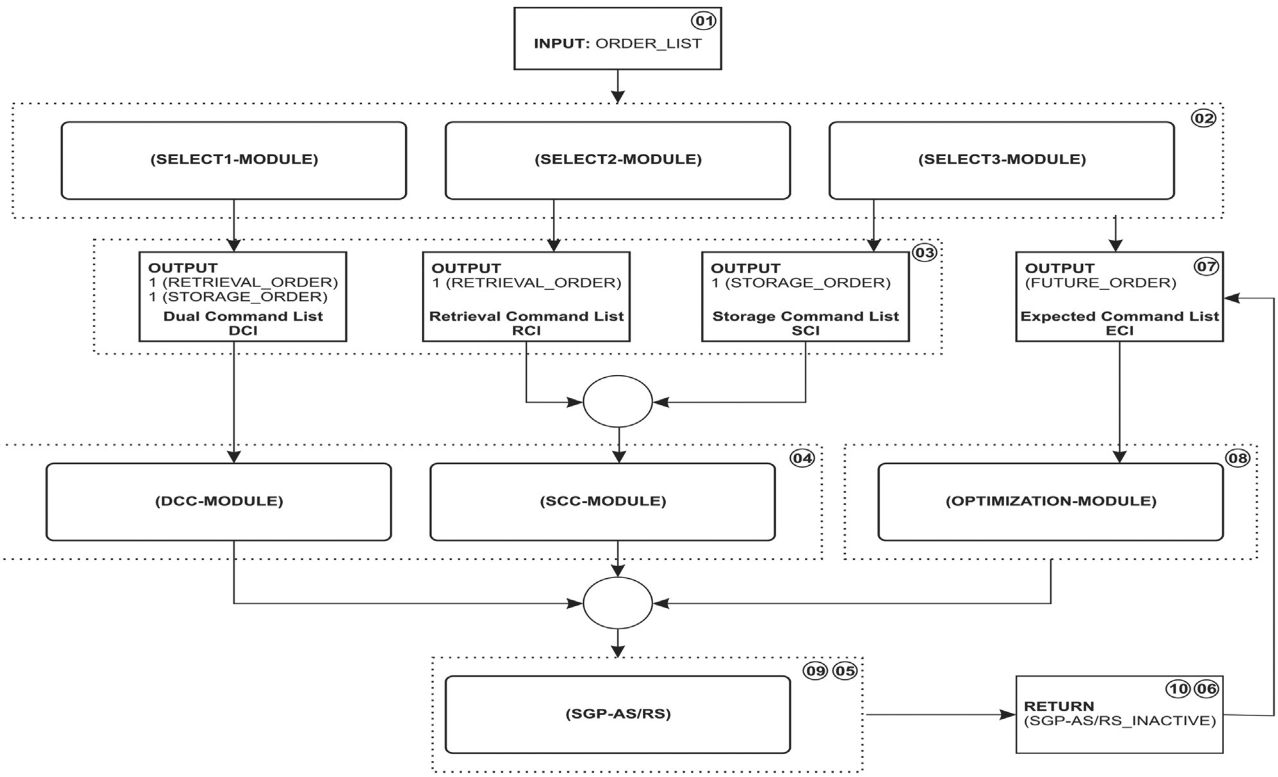

Fig. 5. Control architecture.

Table 2
List of variables.

Variable	Definition
L	Rack length (in m)
H	Rack height (in m)
x, y, x', y'	The I/O station has Cartesian coordinates (0, 0). The storage location is noted P, and has Cartesian coordinates (x, y). The retrieval location is noted P', and has Cartesian coordinates (x', y').
v	Speed of the S/R machine
a	Acceleration of the S/R machine
L_X, H_X	Cartesian border coordinates of each storage class X, along x-axis (L_X), and y-axis (H_X), where X is either storage class A, B, or C
P ($I \in$ class X)	Probability of an item I to belong to a class X, where X is either A, or B or C. this probability is determined from item turnover.

$$E(T_{TB\_R}) = \left[\frac{1}{(LH)^2} \int_0^H \int_0^L \int_0^H \int_0^L \frac{v}{a} + \frac{\sqrt{(x-x')^2 + (y-y')^2}}{v} dx.\,dy.\,dx'.\,dy' \right] \tag{2}$$

The expected travel time of a dual command cycle in case of random assignment is calculated according to Eq. (3):

$$E(T_{DCC\_R}) = E(T_{SCC\_R}) + E(T_{TB\_R}) \tag{3}$$

3.2.2. Class-based assignment

We partition the storage rack into a number of zones based on turnover rates using the ABC storage analysis (cf. Fig. 8). ABC analysis ranks all products in the inventory system based on their contribution to the total demand, with class A items representing the highest turnover products, class B representing the medium turnover products, and class C representing the lowest turnover products [1]. Class A zone is the closest to the I/O station, and is assigned to items with the highest turnover rate. Class C zone is the farthest to the I/O station, and is assigned to items with the lowest turnover rate.

In this assignment policy, single and dual command cycles are calculated as follows:

3.2.2.1. Single Command Cycle (SCC). Let T_{SCC_C} be the expected single command cycle time in case of a class–based policy, P the probability of an item to belong to a class, and A_{sc} the area of the storage class.

Then, T_{SCC_C} is directly proportional to ? ? and inversely proportional to A_{sc}. As there are three storage classes, the single command cycle time (T_{SCC_C}) is updated according to Eq. (4):

$$E(T_{SCC\_C}) = 2 * \left[\begin{array}{l} \frac{P(I \in A)}{Area\ A} * \left[\int_0^{H_A} \int_0^{L_A} \frac{v}{a} + \frac{\sqrt{x^2+y^2}}{v} dxdy \right] \\ + \frac{P(I \in B)}{Area\ B} * \left[\int_{H_A}^{H_B} \int_{L_A}^{L_B} \frac{v}{a} + \frac{\sqrt{x^2+y^2}}{v} dxdy \right] \\ + \frac{P(I \in C)}{Area\ C} * \left[\int_{H_B}^{H_C} \int_{L_B}^{L_C} \frac{v}{a} + \frac{\sqrt{x^2+y^2}}{v} dxdy \right] \end{array} \right] \tag{4}$$

3.2.2.2. Dual Command Cycle (DCC). The expected travel time T_{TB_C} between two storage locations (cf. Section 3.2.1.2) has to be updated to take into account storage partitioning. It is the sum of combinations of travel times between any two locations (there are nine possibilities: $T_{A \to A}$; $T_{A \to B}$; $T_{A \to C}$; $T_{B \to B}$; $T_{B \to A}$; $T_{B \to C}$; $T_{C \to C}$; $T_{C \to A}$; $T_{C \to B}$). T_{TB_C} is updated according to Eq. (5) to fit a class-based storage policy:

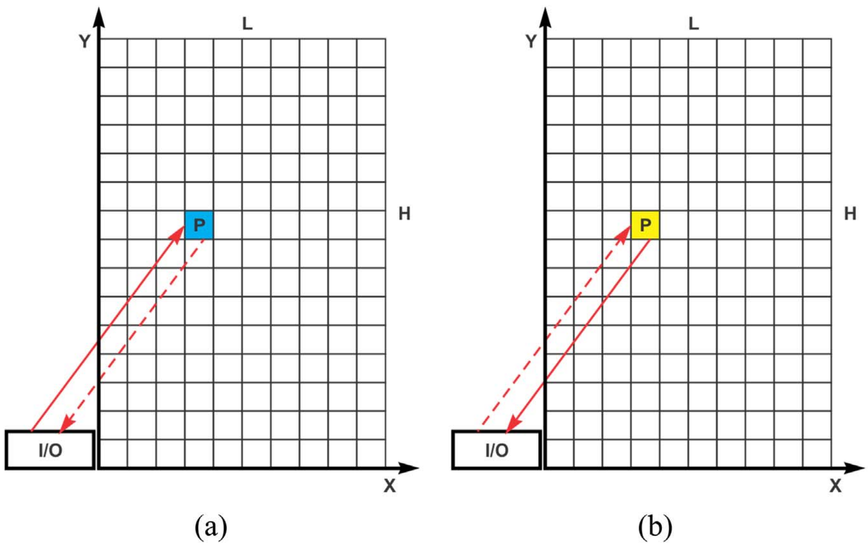

Fig. 6. Single command cycle in case of (a) storage and (b) retrieval.

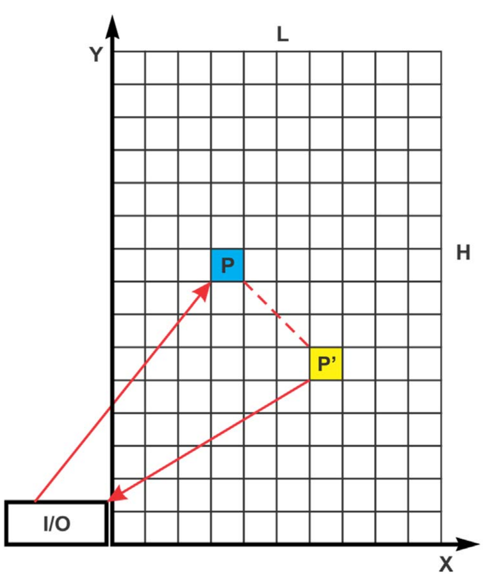

Fig. 7. Dual command cycle in random assignment.

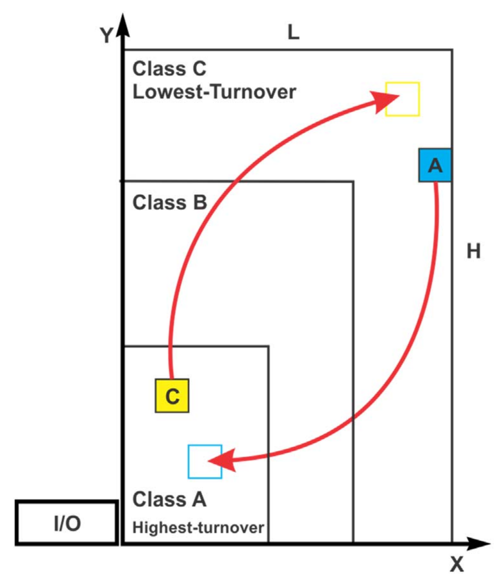

Fig. 8. ABC class-based storage partitioning.

$$E\left(T_{TB\_C}\right) = T_{A\to A} + T_{A\to B} + \ldots + T_{C\to B}$$

$$= \left[\begin{array}{l} \dfrac{P(I\in A)P(J\in A)}{(area\ A)^2} \\ *\left[\displaystyle\int_0^{H_A}\int_0^{L_A}\int_0^{H_A}\int_0^{L_A}\frac{v}{a} + \frac{\sqrt{(x-x')^2+(y-y')^2}}{v}dx.\,dy.\,dx'.\,dy'\right] \\ +\dfrac{P(I\in A)P(J\in B)}{(areaA)*(areaB)} \\ *\left[\displaystyle\int_{H_A}^{H_B}\int_{L_A}^{L_B}\int_0^{H_A}\int_0^{L_A}\frac{v}{a} + \frac{\sqrt{(x-x')^2+(y-y')^2}}{v}dx.\,dy.\,dx'.\,dy'\right] \\ +\cdots+\dfrac{P(I\in C)P(J\in B)}{(area\ C)*(area\ B)}* \\ \left[\left[\displaystyle\int_{H_A}^{H_B}\int_{L_A}^{L_B}\int_0^{H_A}\int_0^{L_A}\frac{v}{a} + \frac{\sqrt{(x-x')^2+(y-y')^2}}{v}ddx.\,dy.\,dx'.\,dy'\right]\right] \end{array}\right] \tag{5}$$

The expected dual command cycle time is calculated using Eq. (6):

$$E\left(T_{DCC\_C}\right) = E\left(T_{SCC\_C}\right) + E\left(T_{TB\_C}\right) \tag{6}$$

3.3. Optimization module

In case there are no storage or retrieval orders, the SRM is idle. We take advantage of this idle time to reorganize the storage distribution, so that we can mainly reduce delivery time and secondarily reduce dispersion. The optimization module is based on a heuristic that performs two tasks (cf. Fig. 9):

1. It rearranges loads that are far from the I/O station and that are expected to be shortly withdrawn (within a user-defined time frame) by placing them as close as possible to the I/O station in order to minimize delivery times;
2. It rearranges loads that are close to the I/O station and that are expected to be withdrawn at a date that exceeds the user-

defined time frame, by placing them in an empty location so that storage dispersion is minimized;

Orders that are expected to be withdrawn are either deterministic, or stochastic. Deterministic orders are confirmed orders that need to be delivered at their defined due date, which is known in advance. Stochastic orders are orders that are not yet confirmed, but that are expected to be served (with a probability) in a near future.

To explain the heuristic, let us define the list of variables of Table 3:

The optimization is based on a heuristic [7] that works according to the flowchart of Fig. 10.

Step 1: Collection of input data. The AS/RS software is fed up with information from the warehouse management system. This information includes the number of occupied places and their locations in the shelf, the expected order release dates, and the expected dates of incoming orders.

Step 2: If there are no orders that are expected to be released within the user-defined time frame, then no optimization is performed, and the SRM remains at rest.

Step 3: If there are orders that are expected to be released within the user-defined time frame, then calculation of utilization ratio α and determination of reorganization module.

– If $\alpha < X$ then occupation ratio is low, which means that the system has many free places, and that expected orders will most probably be storage orders. In this case, a storage-oriented reorganization is performed in order to relocate occupied places that will be released beyond T_f far from the I/O station based on the last out first move (LOFM) rule.
– If $\alpha > Y$ then occupation ratio is high, which means that the system has many occupied places, and that expected orders will most probably be retrieval orders. In this case, a retrieval-oriented reorganization is performed in order to relocate occupied places that will be released within T_f near to the I/O station based on the first out first move (FOFM) rule.
– If $X < \alpha < Y$, then occupation ratio is within a tolerable range, which means that the expected orders will most probably be a mix of storage and retrieval orders. In this case, a mix of storage-oriented and retrieval-oriented reorganization is performed. A movement sequence is generated such that dual command cycles are used (cf. Section 4.2).

4. System testing and results

In this section, we introduce the design of a customized simulation software that enables the testing, evaluation, validation and assessment of the suggested control architecture. We also analyze the performance of the wire-driven SRM by benchmarking it against an equivalent Cartesian SRM.

4.1. Design of a simulation software

As discussed in Section 1.3, available simulation packages focus on the simulation and performance analysis of Cartesian robots, and are not suitable for the testing, evaluation, validation and assessment of the suggested wire-driven SRM. Therefore, we design a simulation software that is customized for our needs. This software enables the user to test both the system components individually, and the overall control architecture. We developed a Graphical User Interface that allows the user to:

1. Interface enterprise information systems, such as Enterprise Resource Planning (ERP) and Manufacturing Execution System

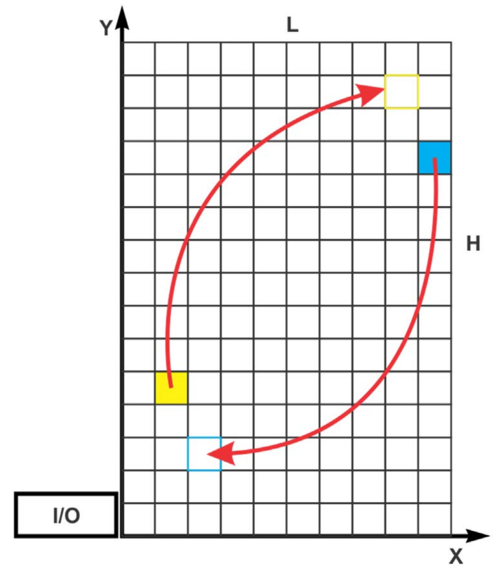

Fig. 9. Optimization in random assignment.

(MES), to collect storage and retrieval requests;
2. Size the system by introducing system parameters (dimensions, speed, etc.);
3. Select and perform an assignment policy;
4. Select and perform single command tasks;
5. Select and perform dual command tasks;
6. Perform optimization tasks;
7. Track system status;
8. Evaluate performance;

Fig. 11 shows the developed graphical user interface and explains its functions.

- *Storage/retrieval requests*: Two lists show the storage and retrieval requests coming from the enterprise information system and that have to be served by the wire-driven SRM. Using *insert* and *clear* buttons, the user can manually add or cancel requests, independently from enterprise information systems. The user can also assign a priority to each request, according for example to rush orders, important customers, etc.
- *System parameters*: This panel shows general information about system parameters and sizing, such as number of storage locations, number of free and occupied locations, SRM speed, acceleration, etc.
- *Assignment policies*: This panel allows the user to select either a random, or a class-based assignment policy.
- *Simulation results*: This panel shows detailed information about performance analysis, in terms of estimated cycle travel times (single and dual commands), system status and occupancy, etc.

Table 3
List of variables.

Variable	Definition
T_f	User defined time frame
$n_{Occupied}$	Number of occupied places in the rack
n_{Total}	Total number of places in the rack
$\alpha = \frac{n_{Occupied}}{n_{Total}}$	Measured occupation ratio of the rack
X, Y	Thresholds on occupation ratio to decide about retrieval oriented reorganization, storage oriented reorganization, or balanced reorganization

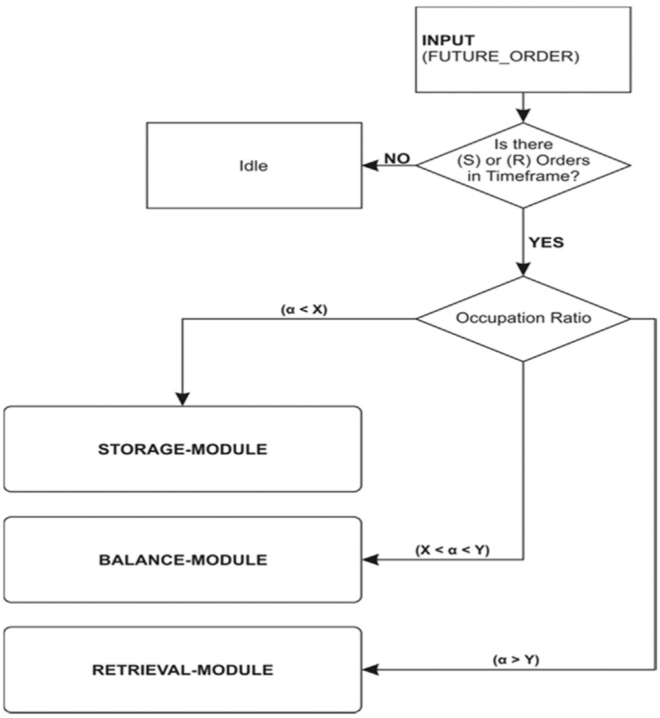

Fig. 10. Flowchart of the optimization algorithm.

4.2. Optimization of idle space

To illustrate the optimization module when the SRM is idle (cf. Section 3.3), let us consider the AS/RS of Fig. 12, where available input data are those of Table 4. An example of storage and retrieval lists, containing order numbers, release dates, and user-defined priorities, are provided in Tables 5 and 6 respectively. In Fig. 12, locations in grey have release dates that exceed the optimization time horizon (current time + time frame). Therefore, optimization of idle space does not include locations in grey. However, the release dates of both locations in blue (for storage orders) and in yellow (retrieval orders) fit within the optimization time horizon and are concerned by the optimization procedure.

As from Table 4 it can be noticed that $X = 20\% < \alpha = 26.66\% < Y = 60\%$, a mix of storage-oriented and retrieval-oriented reorganization should be performed. In this case, the optimization module refers to expected release dates and uses the last out first move (LOFM) heuristic to obtain a relocation sequence of storage orders, and the first out first move (FOFM) heuristic to obtain a relocation sequence of retrieval orders. The orders that have higher priority are sequenced first within each list. Since the system throughput depends on the travel time per transaction, dual command cycle is used to increase the overall system performance. Therefore, the optimization module determines a combined sequence of relocation. By assuming that the dwell point is located at the I/O point of the AS/RS, the optimization module builds a combined list by placing one after another a storage order, then a retrieval order, then a storage order, etc. according to the sequence of appearance of orders in their respective storage and retrieval lists. The resulting combined relocation sequence is provided in Table 7. Assignment of new locations is class-based. For storage orders, last out first moved orders will be assigned to random locations within class C, then class B, then class A if no locations are free, respectively. For retrieval orders, first out first moved orders will be assigned to random locations within class A, then class B, then class C if no locations are free, respectively. Fig. 13 shows the resulting storage rack after optimization.

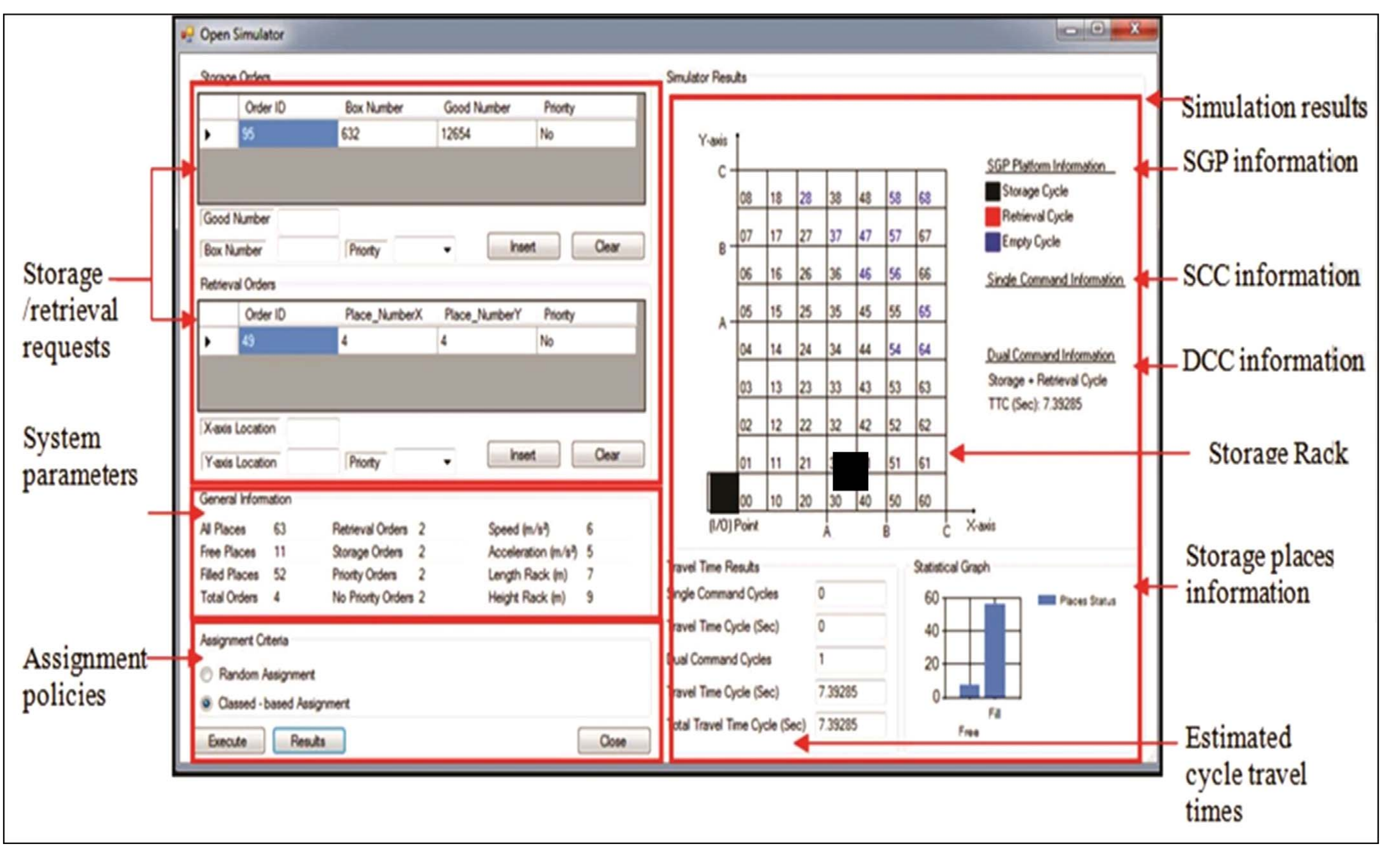

Fig. 11. Graphical user interface.

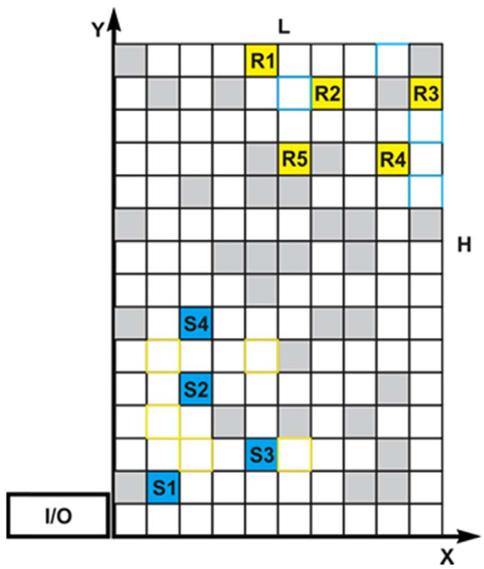

Fig. 12. AS/RS before optimization. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Table 4
Input data for optimization module.

Variable	Value
Current time	15:00:00
T_f	10 min
$n_{Occupied}$	40
n_{Total}	150
$\alpha = \frac{n_{Occupied}}{n_{Total}}$	26.66%
X	20%
Y	60%

Table 5
An example of a storage list.

Order number	Release time	Priority	Relocation sequence based on LOFM
R67	15:01:44	NO	S4
D12	15:03:03	NO	S3
G54	15:06:20	NO	S2
U34	15:07:55	NO	S1

4.3. Benchmarking against an equivalent Cartesian SRM

In this section, we benchmark the wire-driven SRM against an equivalent Cartesian SRM. First, we select a Cartesian SRM which characteristics are equivalent to the wire-driven SRM. Then, we provide equations for a Cartesian SRM in case of a random and class-based storage assignment policies. Finally, we compare the performance of the wire-driven SRM to the performance of the equivalent Cartesian SRM. For both types of robots, we assume that the input/output point is located in the lower-left corner of the storage rack.

4.3.1. Selection of an equivalent Cartesian SRM

Contrarily to Cartesian SRM, which have different speeds and accelerations along x-axis (v_x, $_{ax}$) and y-axis (v_y, a_y) (cf. Section 1.2), the wire-driven SRM has only one speed, denoted linear speed, and only one acceleration values. In the market, for safety and technical reasons, equipment providers usually differentiate between longitudinal and vertical speeds and accelerations in Cartesian SRM. Consequently, it is not possible to find Cartesian SRM with same values of speed along x-axis and y-axis, and same values of acceleration along x-axis and y-axis. To the best of our knowledge and search efforts, the closest SRM that is equivalent to the considered wire-driven SRM has the parameters shown in Table 8.

Table 6
An example of a retrieval list.

Order number	Release time	Priority	Relocation sequence based on FOFM
R13	15:09:24	NO	R5
L24	15:08:20	YES	R1
L53	15:08:53	NO	R4
R04	15:04:44	NO	R3
L33	15:02:20	NO	R2

Table 7
Combined relocation sequence.

S1	R1	S2	R2	S3	R3	S4	R4	R5

4.3.2. Equations for the equivalent Cartesian SRM

Let us define the shape factor w as:

$$w=\frac{V_X}{V_y}\times\frac{H}{L}$$

Then there are three cases for the rack depending on the value of w [10]:

$$\begin{cases} w > 1, & if\ \frac{H}{L} > \frac{V_y}{V_x} \\ w = 1, & if\ \frac{H}{L} = \frac{V_y}{V_x} \\ w < 1, & if\ \frac{H}{L} < \frac{V_y}{V_x} \end{cases}$$

Eqs. (7)–(9) below provided by [10] evaluate the single and dual command cycle times for a Cartesian SRM in the case of a random storage assignment policy, and depending on the value of the shape factor.

$$If\ w\le 1,\ \ E(T_{SCC\_R})_{Cartesian} = 2*\left[\left[1-\frac{w}{2}\right]\frac{V_x}{a_x}+\frac{w}{2}\frac{V_y}{a_y}+\frac{L}{V_x}\left[\frac{1}{2}+\frac{1}{6}w^2\right]\right] \quad (7)$$

$$If\ w>1,\ \ E(T_{SCC\_R})_{Cartesian} = 2*\left[\frac{1}{2w}\bullet\frac{V_x}{a_x}+\left[1-\frac{1}{2w}\right]\frac{V_y}{a_y}+\frac{L}{V_x}\left[\frac{w}{2}+\frac{1}{6w}\right]\right] \quad (8)$$

$$E\left(T_{DCC_R}\right)_{Cartesian}=3\times\frac{1}{2}\left(\frac{V_x}{a_x}+\frac{V_z}{a_z}\right)+2\times\frac{2}{3}\frac{L}{V_x}+\frac{14}{30}\frac{L}{V_x} \quad (9)$$

Eqs. (10)–(12) below provided by [10] evaluate the single and dual command cycle times for a Cartesian SRM in the case of a class-based storage assignment policy.

$If\ w \le 1,\quad then$

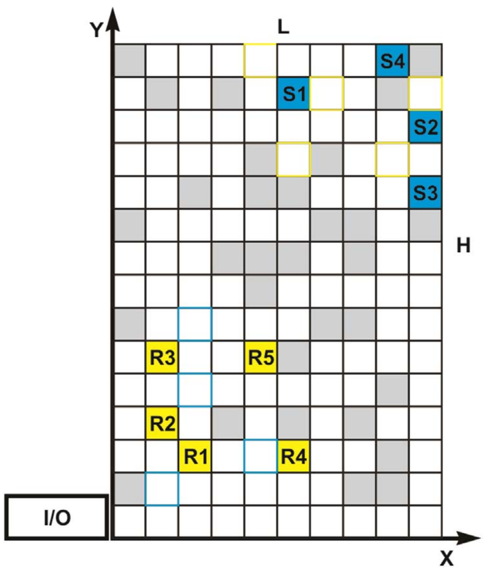

Fig. 13. AS/RS after optimization.

$$E\left(T_{SCC\_Cartesian}\right) = 2*\left[\frac{P(I \in A)}{Area\ A}*\left[2*\left[\left[1-\frac{w}{2}\right]\frac{V_x}{a_x}+\frac{w}{2}\frac{V_y}{a_y}+\frac{L_A}{V_x}\left[\frac{1}{2}+\frac{1}{6}w^2\right]\right]\right] + \frac{P(I \in B)}{Area\ B}*\left[2*\left[\left[1-\frac{w}{2}\right]\frac{V_x}{a_x}+\frac{w}{2}\frac{V_y}{a_y}+\frac{L_B}{V_x}\left[\frac{1}{2}+\frac{1}{6}w^2\right]\right]\right] + \frac{P(I \in C)}{Area\ C}*\left[2*\left[\left[1-\frac{w}{2}\right]\frac{V_x}{a_x}+\frac{w}{2}\frac{V_y}{a_y}+\frac{L_C}{V_x}\left[\frac{1}{2}+\frac{1}{6}w^2\right]\right]\right]\right] \quad (10)$$

If $w > 1$, *then*

$$E\left(T_{SCC\_Cartesian}\right) = 2*\left[\frac{P(I \in A)}{Area\ A}*\left[2*\left[\frac{1}{2w}\cdot\frac{V_x}{a_x}+\left[1-\frac{1}{2w}\right]\frac{V_y}{a_y} + \frac{L_A}{V_x}\left[\frac{w}{2}+\frac{1}{6w}\right]\right]\right] + \frac{P(I \in B)}{Area\ B}*\left[2*\left[\frac{1}{2w}\cdot\frac{V_x}{a_x} + \left[1-\frac{1}{2w}\right]\frac{V_y}{a_y}+\frac{L_A}{V_x}\left[\frac{w}{2}+\frac{1}{6w}\right]\right]\right] + \frac{P(I \in C)}{Area\ C}*\left[2*\left[\frac{1}{2w}\cdot\frac{V_x}{a_x}+\left[1-\frac{1}{2w}\right]\frac{V_y}{a_y} + \frac{L_A}{V_x}\left[\frac{w}{2}+\frac{1}{6w}\right]\right]\right]\right] \quad (11)$$

$$E(TB) = E\left(T_{ij}\right) = \sum \frac{v}{a} + \frac{L_{max}.(P-P')}{2v}*P(I=i,\ J=j) \quad (12)$$

4.3.3. Benchmarking results

Let us consider a storage rack defined according to the parameters shown in Table 9.

Table 10 shows the results of the comparison between the performance of the wire-driven SRM and the equivalent Cartesian SRM. It can be noticed that the wire-driven SRM is able to achieve competitive results and a substantial improvement of travel cycle times in both cases of random and class-based storage policies. Such improvements are not possible without a structural change in the mechanical design of the SRM.

Table 8
Wire-driven and equivalent Cartesian SRM parameters.

Wire-driven SRM		**Cartesian SRM**	
Linear speed:	$v = 6.00$ m/s	Longitudinal speed	v_x=5.00 m/s
		Vertical speed	v_y=3.32 m/s
Acceleration/ deceleration:	$a = 5.00$ m/s^2	Longitudinal acceleration:	a_x=4.00 m/s^2
		Vertical acceleration	a_y=3.00 m/s^2

Table 9
Parameters of the storage rack.

System parameter	**Value**
Rack length L	20 m
Rack height H	40 m

4.4. Performance analysis with respect to variation of system parameters

In this section, we study the impact of changing rack dimensions on the performance of the wire-driven SRM. We also compare this performance to the performance of the equivalent Cartesian SRM. A random storage assignment policy is considered. Table 11 shows the evolution of single command cycle time in seconds function of rack configurations with varying lengths and heights. Fig. 14 plots the evolution of single command cycle time in seconds for the wire-driven SRM (column 4 of Table 11) and for the equivalent Cartesian SRM (column 5 of Table 11) function of rack configurations with varying lengths and heights (column 1 of Table 11).

It can be noticed that the wire-driven SRM in red color (circles line) has approximately constant cycle time when system length and height change, while the Cartesian SRM in blue color (squares line) shows that the cycle time increases rapidly after shape factor (w) becomes more than one. More particularly, Fig. 14 shows that the wire-driven SRM performance is better suited for vertical buildings growing in height, which is a quality much appreciated by investors in cities where space is not available.

5. Discussion

The wire-driven parallel manipulator concept applied to AS/RS offers a number of advantages:

- A wire-driven SRM can be easily adapted to different load ranges and goods to be carried and moved;
- The serial structure of conventional SRM has high inertia and mass, which limit picking time and dynamic performance. Suspending only the lightweight transport platform with a gripper by eight pre-stressed tendons in a parallel configuration makes a wire-driven SRM intrinsically lightweight, which reduces the energy consumption, allows faster movements, and improves dynamic performance;
- A wire-driven SRM takes advantage of the vertical height of the building and is suitable for buildings growing in height;
- The mechanical structure of the wire robot system is simpler and uses fewer equipment than Cartesian AS/RS, which reduces the investment, operation, and maintenance costs.

Table 10
Benchmarking results.

Assignment Policy	Cycle	Wire-driven SRM	Cartesian SRM	Percentage of improvement
Random assignment	Single command	$E(T_{SCC\_R})$ = 11. 24sec.	$E(T_{SCC\_R})_{Cartesian}$=13.7*sec*	21.8%
	Dual command	$E(T_{DCC\_R})$ = 16. 17sec.	$E(T_{SCC\_R})_{Cartesian}$ = 19. 9sec	23.06%
Class based assignment	Single command	$E(T_{SCC\_C})$ = 7. 636sec.	$E(T_{SCC\_C})_{Cartesian}$=10.2*sec*	33.5%
	Dual command	$E(T_{DCC\_C})$ = 10. 19sec.	$E(T_{SCC\_C})_{Cartesian}$=13.8*sec*	35.4%

Table 11
Impact of changing rack dimensions on single command time for wire-driven and Cartesian SRM in case of random storage policy.

Rack configuration number	Rack dimensions with constant area = 800 m^2	Shape factor $w=\frac{V_X}{V_y}\times\frac{H}{L}$	Wire-driven SRM single command cycle time (s)	Cartesian SRM single command cycle time (s)	Performance improvement (%)
1	L=60 m H=13.33 m	0.35	12.90	14.90	13.4
2	L=55 m H=14.55 m	0.4	12.18	14.00	12.9
3	L=50 m H=16 m	0.5	11.50	13.21	12.9
4	L=45 m H=17.78 m	0.6	10.86	12.47	12.8
5	L=40 m H=20 m	0.8	11.25	11.91	5.9
6	L=35 m H=22.86 m	1	9.88	11.63	15.5
7	L=30 m H=26.67 m	1.4	9.64	11.64	17.2
8	L=25 m H=32 m	2	9.70	12.60	23
9	L=20 m H=40 m	3	11.25	14.60	22.6
10	L=15 m H=53.33 m	5.5	11.94	18.32	34.9
11	L=13.33 m H=60 m	7	12.90	20.80	38
12	L=10 m H=80 m	12.5	11.80	26.27	55.2

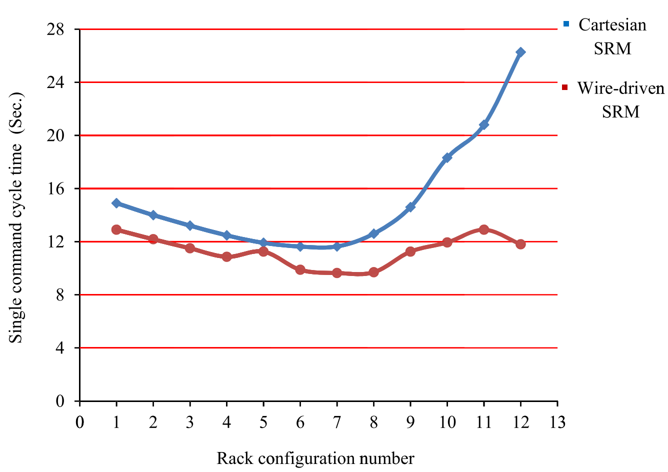

Fig. 14. The impact of changing rack dimensions on cycle travel time for both Cartesian and wire-driven SRM. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

On the other hand, wire-driven SRM require a more complex control system and redundant control techniques, including computerized winches, to maintain the appropriate tension on the cables, to insure safety and stability, and to achieve the advantages described throughout the article. More particularly, our formulas to estimate cycle times do not consider pick-up and drop-off times, which are closely related to the design and realization of the wire-driven storage/retrieval platform, for example to make it able to perform rotations. In addition, our formulas do not consider cable elasticity and mass, and stiffness requirements, which influence modeling and dynamic performance. It is worth recalling that the velocity and acceleration capabilities of a real wire-driven S/R machine depend on the current platform pose, which is neglected within this study. Such considerations are the focus of our future works.

6. Conclusion

In this article, we provided a high-level description of the control architecture of a stacker crane based on an innovative wire-driven storage and retrieval machine (SRM). We described the basic components, the mechanical design of the system, and the control architecture that allows handling mini-load operations. We developed the equations that determine the single and dual command cycle times for the wire-driven SRM in case of random and class-based storage policies. We validated the suggested control architecture using a simulation software specifically developed for this purpose. We benchmarked the wire-driven SRM against an equivalent Cartesian SRM. Our results show that the new wire-driven SRM design and control architecture are more competitive than Cartesian SRM in terms of travel cycle times, and more suitable for buildings growing in height. The competitiveness of our results are due to the facts that the suggested wire-driven SRM (1) has lower total transported mass than its equivalent Cartesian SRM; (2) is able to perform direct motion with high velocities and accelerations compared to its equivalent Cartesian SRM. For future work, it is recommended to study the position of the dwell point of the platform and decide the best position of the platform after execution of each command. Also, refining the cycle time formulas by considering cable mass, elasticity and pick-up and drop-off times are the focus of future research.

References

[1] M.P. Groover, Automation, Production Systems, and Computer-Integrated Manufacturing, Prentice Hall, 2008.
[2] Smita U. Chakole, Development of robotic automated storage and retrieval system (AS/RS), Int. J. Comput. Eng. Res. 3 (3) (2013).
[3] C.K. Soyaslan, M.A. Fenercioglu, An approach of control system for automated storage and retrieval system (AS/RS), in: Proceedings of the World Academy of Science, Engineering and Technology No. 69, 2012. (and).
[4] Stephen C. Graves, Warren H. Hausman, Leroy B. Schwarz, Storage-retrieval interleaving in automatic warehousing systems, Manag. Sci. 23 (1977) 935–945.

[5] K.J. Roodbergen, I.F.A. Vis, A survey of literature on automated storage and retrieval systems, Eur. J. Oper. Res. 194 (2) (2009) 343–362.
[6] I.F.A. Vis, A comparative analysis of storage and retrieval equipment at a container terminal, Int. J. Prod. Econ. 103 (2) (2006) 680–693.
[7] J. Gu, M. Goetschalckx, L.F. McGinnis, Research on warehouse design and performance evaluation: a comprehensive review, Eur. J. Oper. Res. 203 (3) (2010) 539–549.
[8] J. Gu, M. Goetschalckx, L.F. McGinnis, Research on warehouse operation: a comprehensive review, Eur. J. Oper. Res. 177 (1) (2007) 1–21.
[9] R. de Koster, T. Le-Duc, K.J. Roodbergen, Design and control of warehouse order picking: a literature review, Eur. J. Oper. Res. 182 (2) (2007) 481–501.
[10] D. Arnold, K. Furmans, Materialfluss in Logistiksystemen, 6th ed., Springer Berlin Heidelberg,, Wiesbaden, 2009.
[11] J.A. Bozer, A. Yavuz, White, Travel-time models for automated storage/retrieval systems, IIE Trans. 16 (1984) 329–338.
[12] M. Vasili, T.S.H.T.S. Hong, N. Ismail, S. Sulaiman, W.S. Voon, Bead-Sort Algorithm for load shuffling in miniload AS/RS with an open-rack structure, in: Proceedings of the 2009 International Conference on Computers and Industrial Engineering, 2009, pp. 1799–1804.
[14] X. Tang, D. Sun, Y. Qiu, Advances in cable-driven parallel manipulators, Adv. Mech. Eng. (2014).
[15] T. Bruckmann, W. Lalo, C. Sturm, Application examples of wire robots, in: H. Gattringer, J. Gerstmayr (Eds), Multibody System Dynamics, Robotics and Control, Springer Vienna, Vienna, 2013, pp. 291–310.
[16] M. Hassan, A. Khajepour, Analysis of a large-workspace cable-actuated manipulator for warehousing applications, in: Proceedings of the 33rd Mechanisms and Robotics Conference, Parts A and B, vol. 7, 2009, pp. 45–53.
[17] D. Mourtzis, M. Doukas, D. Bernidaki, Simulation in manufacturing: review and challenges, Procedia CIRP 25 (2014) 213–229.
[18] T. Bruckmann, W. Lalo, C. Sturm, Multibody System Dynamics, Springer, Vienna, 2013.
[19] V.E. Gough, S.G. Whitehall, Universal tyre test machine, in: Proceedings of the FISITA 9th Int. Tech. Congr. 1962 117 137.
[20] D. Stewart, A platform with six degrees of freedom, Arch. Proc. Inst. Mech. Eng. 1847-1982 (vols. 1–196), vol. 180(1965), 2006, pp. 371–386.
[21] D. Karimi, M.J. Nategh, A statistical approach to the forward kinematics nonlinearity analysis of gough-stewart mechanism, J. Appl. Math. 2011 (2011) 1–17.
[22] T. Bruckmann, Auslegung und Betrieb redundanter paralleler Seilroboter, Universität Duisburg-Essen, Fakultät für Ingenieurwissenschaften, Dissertation, 2010.
[23] T. Bruckmann, A. Pott, M. Hiller, Calculating force distributions for redundantly actuated tendon-based Stewart platforms, in: J. Lennarčič, B. Roth (Eds.), Advances in Robot Kinematics, Dordrecht: Springer, Netherlands, 2006, pp. 403–412.
[24] R. Verhoeven, M. Miller, Tension distribution in tendon-based Stewart platform" in: ARK, Caldes de Malavalla, June 29 to July 2, 2002, pp. 117–124.
[25] B. Salah, M. Ramadan, B. Noche, Travel time analysis of Stewart-Gough platform in automated storage and retrieval system, in: Proceedings of the IEEE International Conference on Automation and Logistics, (ICAL), 2011, pp. 143–148.
[26] T. Bruckmann, W. Lalo, K. Nguyen, B. Salah, Development of a storage retrieval machine for high racks using a wire robot, in: Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, 2011, pp. 771–780.
[27] J.-Y. Potvin, Genetic algorithms for the traveling salesman problem, Ann. Oper. Res. 63 (3) (1996) 337–370.