Performance-Based Comparison of Control Policies for Automated Storage and Retrieval Systems Modelled by Coloured Petri Nets

Mariagrazia Dotoli  
Dipartimento di Elettrotecnica ed Elettronica  
Politecnico di Bari  
Via Re David 200, 70125 Bari  
ITALY

Maria Pia Fanti  
Dipartimento di Elettrotecnica ed Elettronica  
Politecnico di Bari  
Via Re David 200, 70125 Bari  
ITALY

Abstract — The industrial manufacturing environment is nowadays characterized by fierce global competition, rapid market changes and short product life cycles. Such a complex scenario originated a vast demand for sophisticated techniques guaranteeing adequate planning and control of warehouses. A widely used solution is to adopt Automated Storage and Retrieval Systems (AS/RSs). A typical AS/RS comprises a number of parallel aisles with storage racks, serviced by automated stacker cranes and rail guided vehicles. This paper compares several management strategies addressing the system operational control, i.e., dealing with the AS/RS real time behaviour. A common coloured timed Petri net models the system and the controlled AS/RS operation is highlighted by way of several discrete event simulations carried out in the Matlab-Stateflow software environment. The proposed control policies are compared and discussed on the basis of appropriate performance indices.

I. INTRODUCTION

A widely used solution to nowadays fierce global competition and rapid market changes in modern manufacturing systems is to adopt Automated Storage and Retrieval Systems (AS/RSs) for storing and retrieving finished products and parts [11, 12]. An AS/RS is a combination of automatic handling, storing/retrieval equipment and control systems, characterized by high accuracy and speed. Typically, an AS/RS consists of several aisles with storage racks on either side, each serviced by a stacker crane, operating storage and retrieval of the parts. The cranes move in three directions: along the aisle to perform transfers, sideways between the aisle and the racks at the sides, and vertically to reach the storage/retrieval location. Each aisle is equipped with a storage and a retrieval conveyor. These are serviced by Rail Guided Vehicles (RGVs), unloading the parts to be stored or loading them after retrieval. Finally, a main input and an output buffer station, where the RGVs load or unload pallets, are present [6].

In the control architecture of an AS/RS three hierarchical levels may be identified [8]: the strategic level, containing control policies for long term system performance, the tactical level, collecting control rules for short term performance, and the operational level, dealing with real time behaviour. This paper focuses on the operational control level, that is organized in two subsequent phases. The first one is the scheduler, taking decisions on selecting a proper storage policy, a suitable position of the cranes when idle and a proper retrieval and RGVs routing policy. The second phase is the real time controller, that is in charge of taking decisions on resource allocations in order to avoid conflicts and deadlocks. Numerous studies in the field appeared in recent literature; in particular, investigations deal with typical operational problems, such as defining proper storage and retrieval sequencing policies, in order to maximize the system throughput [1, 7]. However, authors do not examine the RGVs and cranes management, an issue that is crucial for the efficient operation of heterogeneous systems such as AS/RSs.

In this paper we carry out a performance-based comparison of several management strategies of the scheduler. The system is modelled in a Coloured Timed Petri Net (CTPN) framework presented by the authors in [2]. The CTPN model describes in a modular and yet unified framework the heterogeneous system, comprising both the RGVs and cranes subsystems. In particular, the CTPN model is resource oriented: places are resources, tokens are jobs, vehicles and cranes, while colours represent the assigned picking and storing tasks. Moreover, transitions model controllable events concerning resource acquiring and release. The modelling approach is particularly suitable for simulation verification [4, 10]: the controlled system operation is highlighted by way of several discrete event simulations carried out in the Matlab-Stateflow software environment. The control policies are compared and discussed on the basis of appropriate performance indices, including throughput, completion time as well as RGVs and cranes utilization.

II. THE AS/RS DESCRIPTION

In the following we describe a generic heterogeneous AS/RS including unidirectional storage and retrieval conveyors, narrow-aisle stacker cranes, a Rail-Guided Vehicle System (RGVS), an input storage station and an output retrieval station [6] (see Fig. 1). Let \( J = \{ j_k; k=1,...,N_J \} \) be the set of jobs to store or retrieve and \( V = \{ v_h; h=1,...,N_V \} \) and \( G = \{ g_i; i=1,...,N_G \} \) the sets of RGVs and cranes in the system, respectively. In the sequel, subscripts are omitted when unnecessary.

The RGVS rail comprises \( N_{RGVS} \) zones, each representing a location adjacent to one of the \( N_A \) material handling resources, i.e., the conveyors and the input and output stations (see Fig. 1). In the model, each RGVS zone is viewed as a resource that trucks can acquire. Moreover, each crane position, including the home positions, is a resource that the corresponding crane can acquire. Now, let \( N_{i} \) be the...
total number of locations in the racks, i.e., of resources that parts can acquire, and \( r_0 \) a fictitious resource modelling the system output. Thus, we call resources both the physical zones of the system and the actual AS/RS material handling resources. Hence, \( R = \{ r_i, i = 0, \ldots, N_p + N_R + N_A + N_L \} \) is the system resource set. Each resource \( r_i \in R \) has unit capacity, while \( r_0 \) is always available and exhibits infinite capacity.

In the sequel \( r(j) \), \( r(v) \) and \( r(g) \) (\( s(j) \), \( s(v) \) and \( s(g) \)) denote the retrieval (storage) operation respectively assigned to job \( j \), vehicle \( v \) and crane \( g \). Besides, \( r(j) \), \( r(v) \) and \( r(g) \) (\( r(s(j)) \), \( r(s(v)) \) and \( r(s(g)) \)) denote the corresponding residual sequences that have to be performed to complete the preset retrieval (storage) operation starting from a system configuration.

Example 1. Fig. 1 shows the layout of a simple multi-product AS/RS served by several RGVs. The GVS consists of six zones \((r_1, \ldots, r_6)\), each with unit capacity, and comprises \( N_v \) vehicles. In the sequel, \( N_v = 1 \) and \( N_v = 2 \) are considered. The AS/RS includes an input (\( r_7 \)) and an output (\( r_9 \)) station, two unidirectional storage conveyors (\( r_{10} \) and \( r_{11} \)), two unidirectional retrieval conveyors (\( r_{12} \) and \( r_{13} \)) and two narrow aisles (\( N_a = 2 \)). Each aisle includes a rail with four positions, along which the corresponding crane moves, in addition to a home position (\( r_{13} \) and \( r_{26} \) respectively), where the idle S/R machine waits for the next task. Moreover, each crane can move sideways to serve the aisle racks, for a total of thirteen positions associated to each corridor. Every aisle comprises two racks with four locations (see Fig. 1). Hence, with reference to the AS/RS in Fig. 1, it holds: \( N_p = 6 \), \( N_R = 6 \), \( N_A = 26 \), \( N_L = 16 \).

III. THE CTPN MODEL OF THE SYSTEM

This section reviews the CTPN model proposed by the authors in [2] for a heterogeneous AS/RS served by a RGVs. The modelling framework is modular, i.e., the RGVs, storage/retrieval stations, conveyors and aisles are modelled separately in the common CTPN formalism.

A. Overview of Coloured Timed Petri Nets

A Coloured Timed Petri Net is an 8-tuple CTPN\((P, T, \text{Co}, \text{Inh}, \mathbf{C}^+, \mathbf{C}^-, \Omega, M_0)\) where \( P \) is a set of places, \( T \) is a set of transitions, Co is a colour function defined from \( P \times T \) to a set of finite and not empty set of colours \( \text{Co}(p) \times \text{Co}(t) \) [5]. Inh is a weight function for an inhibitor arc connecting a place \( p \in P \) and a transition \( t \in T \) (i.e., \( \text{Inh}(p,t) = 1 \)), so that transition \( t \) can be enabled if \( p \) does not contain any token. \( \mathbf{C}^+ \) and \( \mathbf{C}^- \) are the post-incidence and the pre-incidence \(|P| \times |T|\) matrices respectively, so that \( \mathbf{C}^+(p,t) \) associates to each set of colours of \( \text{Co}(t) \) a set of colours of \( \text{Co}(p) \). \( \mathbf{C}^-(p,t) \) is represented by means of an arc from \( t \) to \( p \) (from \( p \) to \( t \)) labelled with the function \( \mathbf{C}^-(p,t) \) \((\mathbf{C}^-(p,t))\). The symbol \( |\mathcal{A}| \) denotes the cardinality of a generic set \( \mathcal{A} \). The set \( \Omega \) is defined as follows: \( \Omega = \bigcup_{p \in P} \mathbf{J}^+(T) \times \mathbf{C}(X) \). A marking \( M \) is a mapping defined over \( P \) so that \( M(p) \) is a set of elements of \( \text{Co}(p) \), also with repeated elements (i.e., a multi-set) corresponding to the token colours in the place \( p \). \( M_0 \) is the initial marking of the net. Just like in ordinary Petri nets, the flow incidence matrix is \( C = \mathbf{C}^+ \times \mathbf{C}^- \). Now, to investigate the performance of the system, we extend the coloured Petri net with time by way of a global clock [5]. The clock values \( \tau \in \mathbb{R} \) represent the model continuous time. Moreover, we define on the place set \( P \) the function \( \delta : P \rightarrow \mathbb{R} \), where \( \delta(p) \) describes the model earliest time instant at which a token may be removed from place \( p \) by the firing of the enabled transition. In addition to colours, we attach to each token a time stamp, which is reset to zero as soon as the token arrives in the place. When the stamp is equal to or is larger than \( \delta(p) \), the transition enabled by the token is ready for execution.

B. The CTPN Model of the AS/RS

In our model the CTPN\((P, T, \text{Co}, \text{Inh}, \mathbf{C}^+, \mathbf{C}^-, \Omega, M_0)\) describes the complete AS/RS. In particular, a place \( r_i \in P \) denotes a resource \( r_i \in R \) and there is a one to one relationship between resources and places. A transition \( t \in T \) models the flow of tokens (i.e., jobs, vehicles and cranes) in the system. In particular, the transition set \( T \) is partitioned in two subsets: \( T_I \) collects the source transitions \( t_0 \) modelling a job entering the system through a storage station \( r_i \in P \); \( T_F \) is the set of transitions \( t_m \) modelling the flow of parts, vehicles and cranes between two consecutive resources \( r_i \) and \( r_m \). To admit just one token in each resource of \( R \) - \( \{ r_0 \} \), there is an inhibitor arc between each place \( r_m \in P \) with \( r_m \neq r_0 \) and transition \( t_m \in T_F \), i.e., \( \text{Inh}(r_m, t_m) = 1 \).

A coloured token in a place can represent a piece, an RGV or a crane, respectively idle or carrying. The colour of each token may be one of the following couples:
\[ i < r(j), r(v), r(g) > \], \( i < s(j), s(v), s(g) > \), where \( r(j) \) (\( r(v) \)) is the residual sequence of resources in a retrieval (storage) operation that the job \( j \) has to accomplish;
\(ii\) \(<rr(v),j>\) \(<rs(v),j>\), where \(rr(v)\) \((rs(v))\) is the residual sequence of zones that the vehicle \(v\in V\) is booked or occupied by \(j\) has to visit for a retrieval (storage) operation;

\(iii\) \(<rr(g),j>\) \(<rs(g),j>\), where \(rr(g)\) \((rs(g))\) is the residual sequence of zones that the crane \(g\in G\) is booked or occupied by \(j\) has to visit for a retrieval (storage) operation;

\(iv\) \(<tm,P>\), where \(tm\in P\) is the resource occupied by an idle crane or truck.

The colour domain of a place \(r_i\in P\) is: \(Co(r_j)\)\(<rr,j>\) or \(<rs,j>\), where \(rr\) \((rs)\) is a sequence of resources and \(r_i\) is the first resource of \(rr\) \((rs)\) and \(j\in J\cup\varnothing\). Moreover, Co associates with each transition \(t_{im}\in T_F\) a set of possible colours: \(Co(t_{im})\)\(<rr,j>\) or \(<rs,j>\) such that \(rr\) or \(rs\) is a sequence of type \((r_i,r_m,...)\) or \((r_m,r_i,...)\) and \(j\in J\). Hence, the state of the RGVS is represented by the CTPN marking and the following mutually exclusive situations hold for each \(r_i\in P\) with \(i=1,...,N_Z\) (or \(r_i\in P\) with \(i=N_Z+1,...,N_Z+N_R+1,...,N_Z+N_R+N_A\)):

\[M(r_i)=\begin{cases} <rr(v),j> & \text{if } r_i \text{ is occupied by a crane } g \text{ that is booked by or carrying } j \text{ for a retrieval operation; } \\
<rs(v),j> & \text{if } r_i \text{ is occupied by a vehicle } v \text{ that is booked by or carrying } j \text{ for a storage operation; } \\
<rr(j),j> & \text{if } r_i \text{ is occupied by an idle } v \text{ or a crane } g \text{ that is booked by or carrying } j \text{ for a retrieval operation; } \\
<rs(j),j> & \text{if } r_i \text{ is occupied by an idle } v \text{ or a crane } g \text{ that is booked by or carrying } j \text{ for a storage operation; } \\
<\varnothing> & \text{if } r_i \text{ is an idle crane or truck.}
\end{cases}\]

In addition, if \(tm\) is a storage/retrieval station or a unidirectional conveyor, three conditions are possible:

\[M(t_{im})=\begin{cases} <rr(j),j> & \text{if } r_i \text{ is for a retrieval operation; } \\
<rs(j),j> & \text{if } r_i \text{ is for a storage operation; } \\
<\varnothing> & \text{if } r_i \text{ is an idle crane or truck.}
\end{cases}\]

The pre- and the post-incidence matrices \(C^+\) and \(C^-\) are respectively defined as follows:

\(D1\) for each arc \((t_{im},r_i)\), \(C^+(t_{im},r_i)=1\), where I stands for “the function makes no transformation in the elements”, otherwise \(C^-(t_{im},r_i)=0\). Hence, each token leaving a resource \(r_i\in P\) is not modified;

\(D2\) for each arc \((t_{im},r_i)\), \(C^+(t_{im},r_i)=U\), where U is “the function that updates the colour \(<rr(j),j>\) \(<rs(j),j>\)” , otherwise \(C^-(t_{im},r_i)=0\). Here, \(rr\) \((rs)\) is the residual sequence of resources obtained from \(rr\) \((rs)\) omitting the first element \(r_i\). When a token leaves \(r_i\) and reaches \(r_m\), its colour, i.e., its residual path, is updated;

\(D3\) for each arc \((t_{im},r_i)\), \(C^+(t_{im},r_i)=r_i\). When a token leaves \(r_i\) and reaches \(r_i\) again, its colour becomes \(<r_i,\varnothing>\).

The initial marking \(M_0\) is defined as follows:

\[M_0(r_i)=\begin{cases} <rr(j),j> & \text{if } r_i \text{ has to visit for a retrieval (storage) operation; } \\
<rs(j),j> & \text{if } r_i \text{ has to visit for a retrieval (storage) operation; } \\
<\varnothing> & \text{if } r_i \text{ is an idle crane or truck.}
\end{cases}\]

A transition \(t_{im}\in T\) is enabled at marking \(M\) with respect to a colour \(<r_i,t_{im},...,j>\) if the following two conditions are verified for \(r_i\in P\) and \(t_{im}\in T_F\):

\(C1\) \(M(r_i)=<\varnothing>\) if \(Inh(t_{im},r_i)=1\);

\(C2\) for each \(r_i\) input place of \(t_{im}\in T_F\), \(M(r_i)\geq C^+(t_{im},r_i)\) \(<r_i,t_{im},...,j>\), or \(M(r_i)\geq C^-(t_{im},r_i)\) \(<r_i,t_{im},...,j>\).

Condition \(C1\) follows from the inhibitor arc related to the transition and condition \(C2\) represents the enabling condition of the CTPN at marking \(M\). When the stamp \(s(<r_i,t_{im},...,j>)=\delta(t_i)\), the transition enabled by the token is ready for execution. Now, if \(t_{im}\in T\) fires, then the new marking \(M'\) is the following:

\[M'(t_i)=M(t_i)+C'(t_{im},r_0)(<r_0,t_{im},...,j>)-C'(t_{im},r_0)(<r_0,t_{im},...,j>);
\]

\[M'(t_i)=M(t_i)+C'(t_{im},r_0)(<r_0,t_{im},...,j>)-C'(t_{im},r_0)(<r_0,t_{im},...,j>);\]

Finally, the time stamp is set to zero.

Now, the three types of events that can change the marking of the CTPN are the following [3]:

- type 1 event: a job \(j\) enters the system for a storage or retrieval operation.
- type 2 event: a job \(j\) books a crane \(g\) or a vehicle \(v\) for a storage or a retrieval task.
- type 3 event: an idle/booked vehicle \(v\) or crane \(g\) or a job \(j\) moves from a resource to another.
- type 4 event: a job \(j\) leaves the system or is stored in a rack.

\[\text{Fig.2 The CTPN at marking } M_0 \text{ for Example 1 (NV=2)}\]

Example 2. The CTPN describing the RGVS behaviour for Example 1 is composed of \(N_Z+N_R+N_G+1=15\) places, corresponding to the system resources, including the \(N_G=2\) crane home positions. Place \(r_0\) denotes the system output. Moreover, a place \(r_i\in P\) for \(i=1,...,N_Z+N_R+N_G\) denotes the corresponding resource \(r_i\in R\{r_0\}\). A token in \(r_i\in R\) represents a vehicle \(v\in V\) or a crane \(g\in G\) idle, booked or carrying in \(r_i\), or else a job \(j\in J\), e.g. entering the system from the storage station or leaving it from the output station. Fig. 2 shows the CTPN describing the RGVS of Fig. 1, with \(NV=2\) vehicles idle in \(r_2\) and \(r_6\). Places \(r_{13}\) and \(r_{26}\) in Fig. 2 are associated with the home positions of the cranes and these are modelled by tokens with colours respectively \(<r_{13}>\) and \(<r_{26}>\).
indicating that the cranes are idle in the home positions. On the contrary, when a crane is booked, its token colour is the path to reach the rack location where the piece is to be stored or retrieved. If a crane carries a job, the colour of its token is the residual path assigned to the part. Transitions \( t_{1,26} \) and \( t_{13,13} \) (\( t_{26,12} \) and \( t_{13,10} \)) represent the storage (retrieval) operations (see Fig. 2). Finally, a CTPN model for each crane subsystem may easily be implemented: the corresponding places are the home position, the aisle positions and the rack locations, while tokens may represent the idle crane, the busy crane or a stored piece. The resulting model is omitted for sake of brevity: the reader is referred to [3].

IV. THE HIERARCHICAL CONTROL STRUCTURE

The operational controller comprises two levels [3]. The first and higher one (scheduler) selects the storage location assignment policy, defines the requests sequencing policies, decides the cranes operation mode and selects the time instant where to assign a new operation. The second control level is the real time controller, validating the proposed operation and managing resource allocations.

A. The Scheduler

The main scheduler control activities are described in the following [8, 11, 12].

i) Storage Location Assignment: a control policy imposes constraints on the selection of open locations for incoming parts. Well-known policies in the related literature are the following: the random assignment policy, that allows a product to be stored anywhere in the rack, the dedicated storage policy, that assigns specific locations in the racks to each product, the class-based storage policy, that partitions products among a number of classes and reserves a region within the racks to each class.

ii) Queue Selection: a control policy determines which queue (storage or retrieval) to serve next. Widespread policies are as follows: the first come first serve (FCFS) policy, that assigns priority to the queue with the oldest request, the shortest process time (SPT) rule, selecting the queue with the request requiring the shortest completion time, the interleave rule, alternating the queues to be served, the storage priority rule, serving storage operations first, the retrieval priority policy, completing retrieval requests first.

iii) Storage/Retrieval Sequence Selection: different rules can be used to select the next storage or retrieval request upon completion of the current task. The most common rules are: the first in first out (FIFO) rule, that assigns priority to the oldest request, the last in first out (LIFO) rule, allocating precedence to the latest job, the SPT rule, picking first the item with shortest completion time, the FIFO batch SPT rule, selecting the job with shortest process time in the first \( n \) items which have been in the system for the longest time.

iv) Stacker Crane Mode (Dwell Point Selection): a control strategy determines the generic crane command cycle. AS/RSs are typically unit-load retrieval systems, i.e., cranes have unit capacity. Accordingly, cranes either perform one stop (storage or retrieval) or two stops (storage followed by a retrieval) in a single trip along the corresponding aisle. Hence, cranes trips are usually referred to as a single command cycle and a dual command cycle, respectively [11]. Therefore, a single command cycle starts with the crane idle in the home position, performing a storage or retrieval operation and ending up in the home position, while a dual command cycle begins with the crane idle in the home position, serving two subsequent storage and retrieval requests and returning to the home position.

B. The Real Time Controller

The real time controller performs the following activities.

i) Resource validation: when an RGV has to move to the next position, or a job has to acquire a resource, the controller validates movements to prevent collisions.

ii) Storage/retrieval validation: when the scheduler proposes a new storage/retrieval task, the controller validates the operation so that no deadlock situation occurs.

Now, the inhibitor arcs connecting each resource \( r_i \in \{r_0\} \) in the CTPN to transitions \( t_{in}, t_{ni} \) intrinsically manage collisions in the model. Moreover, the authors have shown in [3] that in an AS/RS serviced by RGVs deadlock may occur either at bi-directional conveyors or at unidirectional conveyors. In the latter case, three vehicles must be performing an operation on the same aisle. In this paper we deal with AS/RSs including unidirectional conveyors only, serviced by one or two trucks (\( N_v=1 \) or \( N_v=2 \)). Hence, the generic AS/RS considered is deadlock free and the real time controller manages only collisions by way of the CTPN inhibitor arcs.

C. The Storage Operation

Upon validation of a storage task by the real time controller, the scheduler assigns the job the operation, i.e., a route starting from the storage station and going through a certain number of RGVS zones to the corresponding storage conveyor. The task includes crane selection and indication of the destination in a rack. Moreover, the controller books an idle RGV to load the part. When the truck carrying the job arrives to the storage conveyor, the part is loaded onto the conveyor and is picked up by the crane for transportation to the assigned location. At storage completion the S/R machine returns to its home position (single command operation), or performs a retrieval (dual command cycle). Therefore, with reference to Example 1 and Fig. 1, a possible storage for a part \( j_1 \) under a single command cycle is: \( s(j_1)=\{(f_1,f_2,f_3,f_4,f_5,f_6,f_15,f_16,f_39)\} \). The corresponding paths for the booked RGV (say \( v_1 \) idle in \( r_0 \)) and crane \( g_2 \) are:

\[
\begin{align*}
\text{rs}(v_1) &= \{(f_6,f_1,f_1)\}, \\
\text{rs}(g_2) &= \{(f_1,f_2,f_3,f_4,f_5,f_6,f_15,f_16,f_39)\}.
\end{align*}
\]

D. The Retrieval Operation

Upon retrieval validation, the appropriate crane is activated to travel to the rack location and an RGV is booked for transportation from the retrieval conveyor to the
This section presents a simulation study performed for the AS/RS in Example 1. The aim of the investigation is to compare the effectiveness of three AS/RS scheduling policies by means of appropriate performance measures.

A. Description of the simulation experiments

Several simulation experiments are conducted for the AS/RS in Example 1 under the following assumptions. A set containing all the 32 possible retrieval and storage operations for Example 1 is created and five sets containing respectively 50, 100, 200, 300 and 500 tasks are generated by randomly picking an operation in the first set. The resulting series of tasks are executed with $N_V=1$ and $N_V=2$ by the controlled AS/RS. In particular, the scheduler control activity is designed as follows. As regards the storage location assignment, the simplest and most common rule is considered, i.e., the random assignment policy. Moreover, the FCFS queue selection policy with FIFO storage/retrieval sequence selection and single command mode is considered. Results are compared to the adoption of the random assignment policy and the interleave rule with FIFO storage/retrieval sequence selection and dual command cycle. In the latter case, the interleave rule is investigated both for one dual cycle and four dual cycles executed in succession involving the same aisle and crane. Hence, six experiments are carried out under the following assumptions: FCFS policy ($N_V=1$ and $N_V=2$), interleave rule with one dual cycle ($N_V=1$ and $N_V=2$) and interleave rule with four dual cycles ($N_V=1$ and $N_V=2$). We remark that under the interleave rule a storage always precedes a retrieval task.

B. Performance measures

In order to compare the effectiveness of the selected scheduling strategies in assigning the operations to the AS/RS, several system performance measures are considered.

The first index of effectiveness of the management policies taken into account is the system throughput, i.e., the number of completed storage and retrieval operations in a specified time period. A supplementary measure of performance for each simulation experiment is the completion time, i.e., the run time required to complete all the missions in each set of operations defined in the previous section (50, 100, 200, 300, 500 tasks). Both the above indices assess the selected management policy efficiency in having the system accomplish paths.

Additional performance measures are the RGVs and cranes utilization, evaluated by determining the average percentage time that such means of transport are busy.

In particular, trucks activity may be classified as follows.

(i) Booked travel – the booked RGV is either empty and travelling toward the loading area (i.e., the input station for a storage or a retrieval conveyor for a retrieval) or waiting in order to avoid collisions.

(ii) Loaded travel – the vehicle is carrying a part to accomplish a mission. This state comprises two sub-states: the truck is either transporting a piece toward the assigned unloading area (i.e., the selected storage conveyor for a storage or the output station for a retrieval), or blocked in a zone by the control system in order to avoid collisions.

(iii) Idle – the vehicle, after unloading a part at the destination zone, is idle waiting for an assignment.

Similarly, cranes activity can be classified as follows.

(i) Loaded and booked travel – the crane is either carrying a piece to the assigned unloading area (i.e., the selected rack location for a storage or the retrieval conveyor for a retrieval) or empty and travelling (i.e., toward the home position after performing a storage under the single command cycle, or toward a rack location to load a piece for a retrieval under the single or dual command cycles).

(ii) Idle – the crane is idle in its home position waiting for a task to accomplish.

Accordingly, the RGVs utilization is a performance index measuring the mean percentage of the completion time that the vehicles are busy. This performance index is the sum of the RGVs average percentage time of booked travel and of the RGVs average percentage time of loaded travel.

Likewise, for each simulation experiment the cranes utilization is evaluated, i.e., the mean percentage of the completion time that the stacker machines are busy in the corresponding aisle.

C. Implementation of the CTPN model in Matlab-Stateflow

The CTPN representing the system is implemented in the Matlab-Stateflow environment [9], where it is possible to integrate modelling and simulation of Stateflow event-driven systems (e.g., the RGVS and cranes dynamics) with the execution of Matlab computation routines (e.g., developing the resource and operation validation algorithms), while keeping track of time by way of a software clock. The rationale for choosing Matlab-Stateflow as simulation environment, rather than more evolved and specific software, lies in its simplicity and immediacy. In fact, rather than mimicking the operation of an actual AS/RS in its complexity (number of rack locations, RGVs and cranes in the system, etc.), the purpose of the present simulation study
is testing the efficiency of well-known scheduling policies with reference to several performance measures.

A detailed discussion of the Matlab-Stateflow AS/RS model is beyond the scope of this paper. Nevertheless, we briefly describe the model rationale, since the execution of the CTPN dynamics in Matlab-Stateflow is carried out by a novel implementation technique. More precisely, here the CTPN is represented by a finite states automaton, where a single state represents the generic CTPN place and super-states represent the main AS/RS subsystems. In addition, the CTPN transitions are represented by the state machine transition functions or conditions. Fig. 3 shows the Matlab-Stateflow model implementing the AS/RS in Example 1. The chart comprises eight main subsystems: “In Block”, “RGVS Block”, “Update Marking Block”, “Save Marking Block”, “Storage1 Block”, “Storage2 Block”, “Retrieval1 Block” and “Retrieval2 Block”. More precisely, the “In Block” picks the operation to execute, according to the chosen queue selection and storage/retrieval sequencing rules, and defines the stop criterion of the simulation (run time expiration, pre-set tasks completion, etc.). The “RGVS Block” handles the vehicle bookings and travels in the system and prevents collisions when \( N_V \geq 2 \). Moreover, the “Update Marking Block” updates the current marking of the CTPN, while the “Save Marking Block” stores the actual state of the Petri net taking into account its timing. When a new route is assigned to a part, the appropriate “Storage1 Block”, “Storage2 Block”, “Retrieval1 Block” or “Retrieval2 Block” is invoked, depending on which operation is assigned and which crane is involved.

Hence, implementing the AS/RS in Example 1 in the Matlab-Stateflow environment results in a compact and modular model in which it is easy to modify the system implementation. Indeed, the system model is always similar to the one represented in Fig. 3. E.g. an additional “Storage Block” and a “Retrieval Block” are to be included in the chart if a supplementary aisle is to be added to the system, and some additional places, i.e., finite states machine sub-states, are to be inserted in the “RGVS Block”. If the number of RGVs is to be varied and the system is deadlock free (e.g., \( N_V=1 \) to \( N_V=2 \)), hence only the “RGVS block” must be adjusted. On the other hand, if the number of trucks is increased and deadlock may occur (see section IV), then an operation validation algorithm, i.e., a Matlab deadlock prevention routine, must be included in the “RGVS block” [3]. Preventing deadlock in such warehouse systems will be the subject of future research.

D. Simulation results

Results of the simulation experiments for Example 1 are summarized in Figs. 4-8, as well as in Tab. I and in Tab. II. In Figs. 4-6 results of a preliminary test on the case study are reported. In particular, three simulations with a fixed run time \( T=10^6 \) time instants and one RGV (\( N_V=1 \)) are carried out and the scheduling policies are compared with reference to the throughput. It is apparent from Fig.4 that the interleave policy is more effective than the FCFS rule, especially when only one storage and one retrieval are subsequently performed involving the same crane. The corresponding cranes and RGV performance indices are reported in Fig. 5 and Fig. 6, respectively. It should be noted that such indices do not vary significantly under the different scheduling rules, and such steadiness may be explained with the fact that only one RGV is used in the experiment.

The following experiments record results obtained with a varying number of trucks in the case study. In particular, Fig. 7 reports the completion times of the simulations when the previously defined random sets of operations are executed under the different scheduling policies for Example 1. Obviously, such execution times increase when the tasks set dimension raises. Moreover, Fig. 7 shows, as expected, that when the number of trucks increases from \( N_V=1 \) to \( N_V=2 \) the completion time is considerably reduced under any scheduling policy, since operations may be carried out in parallel, although in general execution times are not halved, because trucks are occasionally blocked by the real time controller in order to avoid collisions.

The RGVs average percentage time of loaded and booked travel under the scheduling rules are respectively reported in Fig. 8a and in Fig. 8b. We remark that both the indices are almost unaffected by a change in the scheduling rule and by an increase in the number of tasks to perform. Moreover, both indices tend to diminish when the RGVs fleet sizes increases from \( N_V=1 \) to \( N_V=2 \).

Tab. I shows the RGVs utilization index, which is the sum of the indices reported in Fig. 8a and in Fig. 8b, under the different scheduling policies for Example 1. The table shows that such an index is nearly unmodified by a variation in the number of operations or even in the scheduling rules. In particular, Tab. I shows that when \( N_V=1 \), i.e., only one truck is available, about 83-85% of the completion time the vehicle is busy either carrying jobs or traveling empty to a loading zone, while in the rest of the run time it is idle. Moreover, when \( N_V=2 \), i.e., two vehicles are in the system, about 74-78% of the completion time the trucks are busy. We remark that under all the considered scheduling policies
the RGVs utilization decreases when the RGVs fleet size increases, as expected.

Tab. II shows the cranes utilization under the different scheduling policies for Example 1. Similarly to the RGVs utilization index, the table shows that the cranes utilization index is practically unchanged when a variation in the number of tasks or even in the scheduler control policies occurs. In particular, during about 50% of the completion time cranes are busy either carrying parts or traveling empty to a loading zone, while in the rest of the execution time they are idle in the corresponding home position.

As a summary of the performance for the different scheduling policies considered, we remark that the AS/RS is more efficient in terms of throughput and execution time under the interleave rule with one storage and one retrieval task alternated on the same aisle. The same scheduling policy with four alternations follows, as regards effectiveness, while the less efficient management strategy is the FCFS rule. Moreover, the RGVs and cranes utilization indices are practically independent of the scheduling policy considered.

VI. CONCLUSIONS

This paper presents a performance based comparison of several management policies for operational control of automated storage and retrieval systems. A deadlock-free case study is considered. The system structure and dynamics are described by a resource oriented Coloured Timed Petri Net (CTPN) previously proposed by the authors, allowing the investigation of the system performance. In order to compare the policies on the basis of appropriate performance indices, we propose a Matlab-Stateflow environment that reproduces the CTPN modularity and implements the discrete-event system via a finite states automaton. Several simulation experiments performed for the case study show the effectiveness of one of the considered control strategies. Future research will deal with simulation verification of control policies managing deadlock conditions together with collisions.

VII. REFERENCES

Fig. 6 RGV performance indices under different scheduling policies for a fixed run time $T=10^6$ for Example 1 ($N_v=1$)

Fig. 7. Completion times for several random sets of operations under different scheduling policies for Example 1

(a) RGVs mean percentage loaded travel time (b) RGVs mean percentage booked travel time for several random sets of operations under different scheduling policies for Example 1

Tab. I. RGVs utilization for several random sets of operations under different scheduling policies for Example 1

<table>
<thead>
<tr>
<th>RGVs utilization</th>
<th>Number of tasks</th>
<th>50</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCFS 1 RGV</td>
<td></td>
<td>85,44</td>
<td>84,95</td>
<td>84,94</td>
<td>85</td>
<td>84,78</td>
</tr>
<tr>
<td>Interleave 1 alternation 1 RGV</td>
<td></td>
<td>83,25</td>
<td>83,45</td>
<td>83,55</td>
<td>83,58</td>
<td>83,6</td>
</tr>
<tr>
<td>Interleave 4 alternations 1 RGV</td>
<td></td>
<td>84,65</td>
<td>84,78</td>
<td>84,83</td>
<td>84,85</td>
<td>84,87</td>
</tr>
<tr>
<td>FCFS 2 RGVs</td>
<td></td>
<td>76,63</td>
<td>75,96</td>
<td>76,84</td>
<td>75,53</td>
<td>75,89</td>
</tr>
<tr>
<td>Interleave 1 alternation 2 RGVs</td>
<td></td>
<td>73,98</td>
<td>75,19</td>
<td>74,65</td>
<td>74,39</td>
<td>74,44</td>
</tr>
<tr>
<td>Interleave 4 alternations 2 RGVs</td>
<td></td>
<td>78,81</td>
<td>78,53</td>
<td>78,31</td>
<td>78,23</td>
<td>78,24</td>
</tr>
</tbody>
</table>

Tab. II. Cranes utilization for several random sets of operations under different scheduling policies for Example 1

<table>
<thead>
<tr>
<th>Cranes utilization</th>
<th>Number of tasks</th>
<th>50</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCFS 1 RGV</td>
<td></td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50,73</td>
<td>50</td>
</tr>
<tr>
<td>Interleave 1 alternation 1 RGV</td>
<td></td>
<td>51,725</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>Interleave 4 alternations 1 RGV</td>
<td></td>
<td>49,985</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>FCFS 2 RGVs</td>
<td></td>
<td>50</td>
<td>49,995</td>
<td>49,635</td>
<td>49,995</td>
<td>50</td>
</tr>
<tr>
<td>Interleave 1 alternation 2 RGVs</td>
<td></td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>Interleave 4 alternations 2 RGVs</td>
<td></td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>49,99</td>
<td>50</td>
</tr>
</tbody>
</table>