RESEARCH ARTICLE

Modular supervisory control for multi-floor manufacturing processes

Fotis N. Koumboulis[1] · Dimitrios G. Fragkoulis[2] · Aristides A. Michos[2]

Abstract

Due to space availability limitations and high land costs, there is an increasing development of multi-floor manufacturing (MFM) systems in urban and industrial areas. The problem of coordination in a multi-floor manufacturing process, in the Ramadge Wonham framework, is introduced. The manufacturing chain of each floor and the elevator system are modeled in the form of finite deterministic automata. The models of the multi-floor manufacturing process are parametric with respect to the number of floors and the number of manufacturing machines on each floor. The coordination desired performance is formulated in the form of desired regular languages in analytic forms. The languages are realized by appropriate supervisors in the form of finite deterministic automata. The models of the supervisors are also parametric with respect to the number of floors and the number of manufacturing machines on each floor. The total control of the coordination of the multi-floor manufacturing process is accomplished via a modular supervisory control architecture. The complexity of the supervisors as well as the complexity of the total modular supervisory architecture are determined in analytic forms with respect to the number of floors and the number of manufacturing machines on each floor. The special case of a two floor manufacturing process is presented as an illustrative example.

Keywords Manufacturing processes · Discrete event systems · Supervisory control · Industry 4.0

1 Introduction

Space availability limitations and high land costs, being typical characteristics of urban and industrial areas in developed countries, motivate the increasing development of multi-floor manufacturing (MFM) systems (see [1,2] and the references therein). Improvement of the effectiveness and flexibility of manufacturing processes is a trend in modern manufacturing systems ([3–5]). This trend, being a direction of Industry 4.0,

✉ Fotis N. Koumboulis
fkoumboulis@dind.uoa.gr

Dimitrios G. Fragkoulis
dfragkoulis@uoa.gr

Aristides A. Michos
armichos@core.uoa.gr

[1] Department of Digital Industry Technologies, School of Science, National and Kapodistrian University of Athens, Euripus Campus, 3400 Psahna Euboea, Greece

[2] Core Department, National and Kapodistrian University of Athens, Euripus Campus, 3400 Psahna Euboea, Greece

requires ongoing automation of manufacturing practices. In the framework of this trend, the requirement is more intensive in MFM processes, where the material handling network is in general more complex, as compared to single-level processes.

The basic components of a MFM process are the manufacturing machines (workstations), the horizontal material handling systems and the vertical material handling system. All components compose the production line. The horizontal material handling systems organize the production chain of each floor through robots (see [6,7]), automated guided vehicles (AGVs) (see [6,7,9]) and intelligent reconfigurable trolleys (IRTs) (see [1,2]). The vertical material handling systems are mainly elevators (see [6–8]).

In [1,2], the supply chain network design problem is considered for MFM clusters, through a decision-making approach. In [6], integer linear programming is used. In [7], integer linear programming combined with genetic algorithms, is proposed. Both works deal with intra-cell, inter-cell and inter-floor material handling, through cost minimization. In [8,9], the multi-floor facility layout problem is considered.

The coordination of the overall (horizontal and vertical) handling system is of critical importance for the improvement

of the effectiveness and flexibility of MFM processes through automation. Discrete event system (DES) approaches are offered for this goal. DES approaches have already been used to model and control single level production machine chains (indicatively, see [10–12]) and elevator systems (see [13,14]). To the best of our knowledge, the coordination of the handling material system for MFM processes has not as yet been studied through DES point of view.

The coordination problem of an MFM system will be studied here, in the Ramadge Wonham (RW) framework, indicatively see [11,15,16]. On each floor there is a chain manufacturing process, while the transfer between floors is accomplished through an elevator system. The multi-floor system is modeled using finite deterministic automata. The automata that model the product transfer on every floor and the automaton that models the elevator system are in accordance with the respective models presented in [12] for a single floor manufacturing process and [13] for a mining process elevator system. The coordination of the MFM process is analyzed into a set of desired languages and is accomplished through the development of a modular supervisory design scheme.

In comparison to rule-based control methods, the use of DES models and DES controllers in the RW framework has two distinct advantages. It provides the capability to verify, in a systematic way, the realizability of the controller with regard to the activation/deactivation of the transitions of the process triggered by uncontrollable events, and the property of nonblocking for the controlled system. Finally, another advantage of the RW framework is that is offered for realization of the designed controllers to industrial control devices, such as PLCs and PACs, indicatively see [17,18].

Here, the multi-floor process is modeled in parametric form. The parametric nature of the model is expanded in two directions. The first, is the description of the horizontal material handling system. In particular, the mobile robot systems are described by general operation DES models, being independent from special technical characteristics of each robot. The model of the vertical material handling system belongs also to this direction. The second direction is the parametric number of the manufacturing machines and the parametric number of the floors. The parametric nature of the model is expanded to the proposed supervisors. The parametrical structure of the models and the supervisors provides generality and flexibility in the analysis of the controlled process.

In [12], a two layer supervisor has been designed only for the horizontal handling system. In particular, a modular supervisor architecture with two supervisor automata has been proposed. Here, a supervisory control scheme with one supervisor per floor, for the horizontal material handling, is proposed. It is mentioned that the present results facilitate the analysis of the control scheme, namely the check of the controllability of the desired languages and the nonblocking property of the controlled automaton.

The paper is organized as follows: in Sect. 2, the multi-floor process is described. In Sect. 3, a fully parametric discrete event model of the process is developed. In Sect. 4, the design specifications are imposed and translated to appropriate desired languages in analytic forms. The properties of these languages are proven. Finally, in Sect. 5, the realization of the supervisory control scheme is accomplished in a modular way.

2 The multi-floor manufacturing process

The plant is an interconnected MFM system. The raw product is on the ground (lowest) floor. Also, the manufactured product is on the ground floor. On the other floors there are installed manufacturing machines. The machines of the same floor are connected through a local rail track system. The local rail track system is a closed path, in the form of a ring track system. One mobile robot platform is installed on the ring track system of the respective floor. The task of the robot platform is the transfer of the products among manufacturing machines of the same floor. The floor platform is considered to move clockwise. On every platform two synchronized coordinating robotic manipulators are mounted. On every floor there are several input and output stations where the robotic platform delivers and takes the products. The products are transferred from the ground level at the appropriate floor and vice versa through an elevator system (see [1]).

One mobile robot platform serves all manufacturing machines of the floor. The lowest flower is indexed by 0. There are no manufacturing machines on the ground floor. Upstairs the ground floor, there are m floors with manufacturing machines. These floors are indexed by $i \in \{1, \ldots, m\}$ where the highest floor is indexed by m.

In Fig. 1, the schematic of the i-th floor of the MFM system is presented. The number of the manufacturing machines on the i-th floor is n_i. The machines on the i-th floor are indexed by $j \in \{1, \ldots, n_i\}$. Each machine has an input station and an output station. The input station of the j-th machine is indexed by $2j$ and the output station of the j-th machine is indexed by $2j + 1$. On every floor there is a station indexed by 1 and a station indexed by $2n_i + 2$. Station 1 is the station where the elevator unloads products and Station $2n_i + 2$ is the station where the manufactured products of the i-th floor are accumulated before transferred to the car of the elevator. On each floor, the products are manufactured serially. In other words, the product manufactured by the j-th machine is transferred to be manufactured by the $j + 1$-th machine. When the manufacturing of the product of the n_i-th machine
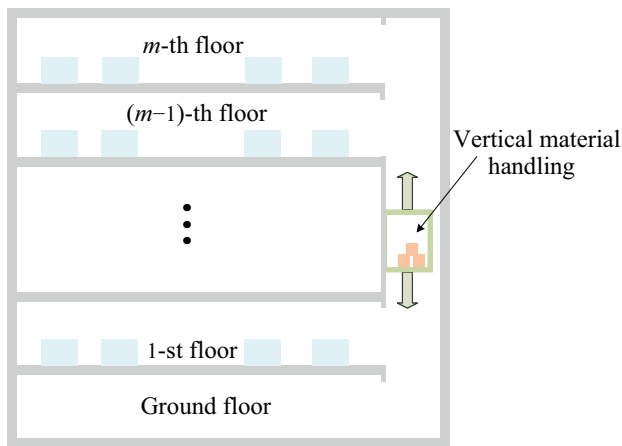
Fig. 1 Top view of the i-th floor of the manufacturing process (horizontal material handling)
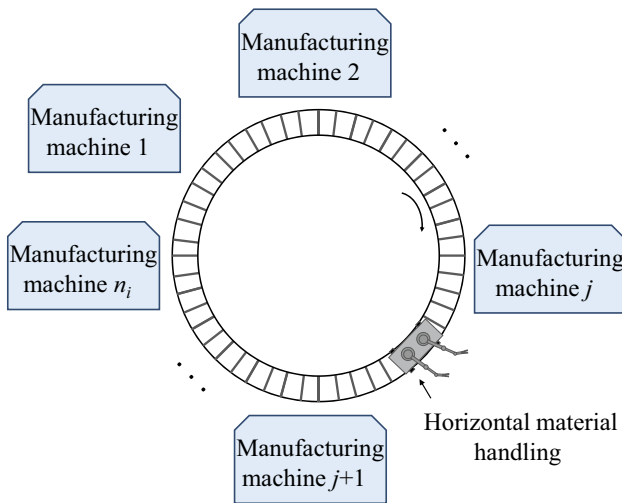

Fig. 2 Side view of the i-th floor of the manufacturing process (vertical material handling)

of the i-th floor is completed, then the manufactured product is placed at Station $2n_i + 2$ of the i-th floor.

In Fig. 2, the schematic of the side view of the MFM process is presented. The elevator system has one car that moves in a vertical shaft between floors. The elevator has two different operations: The first is the transfer of products from the input/output (ground) floor to any other floor and vice versa. The second is the transfer of products from one floor to the next floor, i.e., to transfer products from the i-th floor to the $i + 1$-th floor. The idle position of the elevator is at the level of the input/output floor. In order to handle calls from different floors, a prioritization system will be designed.

3 Modeling of the process

3.1 Horizontal material handling

The finite deterministic automata describing the product transfer between the machines installed on the same floor, namely the horizontal material handling system, will be presented. The automata are in accordance with the model of the single floor manufacturing process presented in [12]. On the ground floor being the input/output floor, there are no manufacturing machines and stations. Thus, the automaton of each floor is indexed by $i \in \{1, \ldots, m\}$. Before presented the automaton of the i-th floor, it is noted that each station has a terminal switch being activated when the platform reaches the station. When a manufactured product is available at an output station then an appropriate event is generated. Thus, there are $n_i + 1$ product availability events, including the output station of the elevator. The robotic platform delivers the products from the output station of a machine to the input station of the next machine, the delivery process is concluded when the terminal switch is activated. The automaton of the i-th floor is of the form of the 6-tuple (for 6-tuple automata, see [12,13,15,19,20]):

$$^{i}G = (^{i}Q, {}^{i}E, {}^{i}H, {}^{i}f, {}^{i}x_0, {}^{i}Q_m), \quad i \in \{1, \ldots, m\}.$$

The states of the mobile robot system are four. The states of the platform are two. The first state of the platform is denoted by $^{i}q_{1,1}$ and is the state where the platform is still. The second state of the platform is denoted by $^{i}q_{1,2}$ and is the state where the platform is moving. The states regarding the manipulators are also two. The first state, describing the case where the grippers of the two manipulators are open and the manipulators are at the low position, is denoted by $^{i}q_{2,1}$. The second state, describing the case where the grippers of the manipulators are closed and the manipulators are at the high position, is denoted by $^{i}q_{2,2}$. Thus, the set of the states of the automata of the i-th floor are

$$^{i}Q = \{(^{i}q_{1,1}, {}^{i}q_{2,1}), (^{i}q_{1,1}, {}^{i}q_{2,2}), (^{i}q_{1,2}, {}^{i}q_{2,1}), \\ (^{i}q_{1,2}, {}^{i}q_{2,2})\}.$$

The alphabet of the automaton is

$$^{i}E = \{^{i}e_1, {}^{i}e_2, {}^{i}e_3, {}^{i}e_4\} \cup {}^{i}E_T \cup {}^{i}E_P.$$

The event $^{i}e_1$ is the command for the mobile robot system to start moving. This command is usually generated by the operator. The event $^{i}e_2$ is the command to stop the robotic mobile system and is usually generated by the operator or automatically in the case of the presence of an obstacle in front of the cart. The event $^{i}e_3$ is the command to move the manipulators to the low position and open the grippers. The

event $^i e_4$ is the command to close the grippers and move the manipulators to the high position. The events set $^i E_T = \bigcup_{k=1}^{2n_i+2} \{^i e_{T,k}\}$ includes all the event signals generated when the terminal switches of stations $1,2,3,\ldots,2n_i+2$ are turned on. The events set $^i E_P = \bigcup_{j=0}^{n_i} \{^i e_{P,2j+1}\}$ includes all the event signals denoting that there are available products at stations $1, 3, 5, \ldots, 2n_i + 1$.

The initial state of the automaton is $^i x_0 = (^i q_{1,1}, ^i q_{2,1})$. All states of the automaton describe desirable operations. Therefore, all states of the automaton are marked, i.e., $^i Q_m = {}^i Q$.

The active event sets of the states of the automaton are

$$^i H \left((^i q_{1,1}, ^i q_{2,1}) \right) = {}^i H \left((^i q_{1,1}, ^i q_{2,2}) \right) = {}^i E - \{^i e_2\},$$
$$^i H \left((^i q_{1,2}, ^i q_{2,1}) \right) = {}^i H \left((^i q_{1,2}, ^i q_{2,2}) \right) = {}^i E.$$

The transitions of the automaton are

$$^i f((^i q_{1,1}, ^i q_{2,1}), ^i e_1) = (^i q_{1,2}, ^i q_{2,1}),$$
$$^i f((^i q_{1,1}, ^i q_{2,1}), ^i e_3) = (^i q_{1,1}, ^i q_{2,2}),$$
$$^i f((^i q_{1,1}, ^i q_{2,1}), e) = (^i q_{1,1}, ^i q_{2,1}),$$
$$e \in {}^i E_P \cup {}^i E_T \cup \{^i e_4\},$$
$$^i f((^i q_{1,1}, ^i q_{2,2}), ^i e_1) = (^i q_{1,2}, ^i q_{2,2}),$$
$$^i f(^i q_{1,1}, ^i q_{2,2}), ^i e_4) = (^i q_{1,1}, ^i q_{2,1}),$$
$$^i f((^i q_{1,1}, ^i q_{2,2}), e) = (^i q_{1,1}, ^i q_{2,2}),$$
$$e \in {}^i E_P \cup {}^i E_T \cup \{^i e_3\},$$
$$^i f((^i q_{1,2}, ^i q_{2,1}), ^i e_3) = (^i q_{1,2}, ^i q_{2,2}),$$
$$^i f((^i q_{1,2}, ^i q_{2,1}), e) = (^i q_{1,2}, ^i q_{2,1}),$$
$$e \in {}^i E_P \cup \{^i e_1, ^i e_4\},$$
$$^i f((^i q_{1,2}, ^i q_{2,1}), e) = (^i q_{1,1}, ^i q_{2,1}),$$
$$e \in {}^i E_T \cup \{^i e_2\},$$
$$^i f((^i q_{1,2}, ^i q_{2,2}), ^i e_4) = (^i q_{1,2}, ^i q_{2,1}),$$
$$^i f((^i q_{1,2}, ^i q_{2,2}), e) = (^i q_{1,2}, ^i q_{2,2}),$$
$$e \in {}^i E_P \cup \{^i e_1, ^i e_3\},$$
$$^i f((^i q_{1,2}, ^i q_{2,2}), e) = (^i q_{1,1}, ^i q_{2,2}),$$
$$e \in {}^i E_T \cup \{^i e_2\}.$$

The closed behavior of an automaton [11] (language generated by an automaton [15]) is the set of all words triggering transmissions from the initial state of the automaton to any state of the automaton. The closed behavior of an automaton is denoted as $L(\cdot)$, where the argument quantity is the automaton. The marked behavior of an automaton [11] (language marked by an automaton [15]) is the set of all words triggering transitions from the initial state of the automaton to any marked state of the automaton. The marked behavior of an automaton is denoted as $L_m(\cdot)$, where the argument is the automaton. Clearly, the closed behavior of the present

automaton is equal to its marked behavior. The closed and the marked behaviors are $L(^i G) = L_m(^i G) = \overline{^i L}$, where

$$
\begin{aligned}
^i L = \Big(& (^i c_{q_{1,1}} + {}^i e_3(^i c_{q_{1,2}}^*)) (^i e_4 + {}^i e_1(^i c_{q_{2,2}} \\
& + {}^i e_2(^i c_{q_{1,2}}^*)(^i e_1))^*(^i e_2)(^i c_{q_{1,2}}^*)(^i e_4)) \\
& + (^i e_1 + {}^i e_3(^i c_{q_{1,2}}^*)(^i e_1)(^i c_{q_{2,2}} \\
& + {}^i e_2(^i c_{q_{1,2}}^*)(^i e_1))^*(^i e_4))(^i c_{q_{2,1}} + {}^i e_3(^i c_{q_{2,2}} \\
& + {}^i e_2(^i c_{q_{1,2}}^*)(^i e_1))^*(^i e_4))^*(^i e_2 + {}^i e_3(^i c_{q_{2,2}} \\
& + {}^i e_2(^i c_{q_{1,2}}^*{}^i e_1)^*(^i e_2)(^i c_{q_{1,2}}^*)(^i e_4)) \Big)^*,
\end{aligned}
$$

where $^i c_{q_{1,1}}$, $^i c_{q_{1,2}}$, $^i c_{q_{2,1}}$ and $^i c_{q_{2,2}}$ are the regular expressions of the languages $^i E - \{^i e_1, ^i e_2, ^i e_3\}$, $^i E - \{^i e_2, ^i e_3\}$, $^i E - \{^i e_1, ^i e_2, ^i e_4\}$ and $^i E - \{^i e_2, ^i e_4\}$, respectively. Details about regular expressions and regular languages can be found in [11,15]. The symbol $\overline{\cdot}$ denotes the prefix closure of the argument language. According to [11,15], the prefix closure of a languages is the set of the words being prefixes of the words of the argument language. Also, according to [11,15], an automaton is nonblocking if the prefix closure of its marked behavior is equal to its closed behavior. Clearly, the present automaton is a nonblocking automaton, i.e., $L(^i G) = \overline{L_m(^i G)}$. The controllable event set is $^i E_c = \{^i e_1, ^i e_3, ^i e_4\} \bigcup {}^i E_T \bigcup {}^i E_P$. The uncontrollable event set is $^i E_{uc} = \{^i e_2\}$.

3.2 Vertical material handling

The finite deterministic automaton describing product transfer system between floors, namely the vertical material handling system, will be presented. The product transfer is accomplished through an elevator system. The system consists of a car, a motor, one door on each floor, appropriate sensors and a robotic system. The car moves vertically along the elevator shaft. The motor generates car motion in two directions (up or down). The door of each floor opens and closes through appropriate commands. At each level, in front of the shaft, there is a door. The door is locked by a mechanism unless the car is stopped and aligned with the door. If the door is unlocked, it may open and close manually.

A sensor (switch contact circuit) detects whether the door is opened or closed. The sensors of the $m + 1$ doors are connected through the same cable. This way, a common "open" signal and a common "closed" signal is produced. At each floor there are two buttons. The first is the "call" button and the second is the "send" button. The call button commands the car to arrive at the floor and the send button commands the car to go the ground floor. At the ground floor, there is a button that commands the car to come from every floor.

At each floor, two more sensors are installed. The first is the floor sensor. It is placed in the shaft a little over the top of the door of the floor The second is the car alignment sensor

and is placed in the shaft at the height of the floor. The total number of the shaft's sensors is $2m + 2$. When a level sensor is activated, then it would be desired for the motor to switch to slow speed. When an alignment sensor is activated then it would be desired for the motor to stop. Finally, there is an emergency signal produced by sensors in the shaft e.g., fire detector, earthquake detector.

The robotic system is installed inside the car. The robotic system is mounted on one side of the car. The mission of the robotic system is to load and unload products. The products are unloaded to Station 1 of each floor and the manufactured products are loaded from Station $2n_i + 2$ of the i-th floor, where $i \in \{1, \ldots, m\}$. To cover the ground floor, where there are no manufacturing machines, it is considered that $n_0 = 0$. This way, when the car is at the ground level, the products are loaded from Station 2 of the ground floor. For brevity purposes the robotic system of the car will not be modeled here.

Finally, it is mentioned that the automaton of the elevator system will be developed in accordance with the automaton presented in [13] for a mining process.

If the car is aligned with the level of the i-th floor, then the state of the car is denoted by $q_{l,i}$, where $i \in \{0, \ldots, m\}$. The set of all states $q_{l,i}$ is denoted by Q_l. Clearly, $|Q_l| = m + 1$, where $|\cdot|$ denotes the cardinality of the argument set. The set of the states of the motor is $Q_r = \{q_{r,1}, q_{r,2}, q_{r,3}, q_{r,4}, q_{r,5}\}$. If the car is stopped, then the motor is at state $q_{r,1}$. If the car moves up with high speed, then the motor is at state $q_{r,2}$. If the car moves up with low speed, then the motor is at state $q_{r,3}$. If the car moves down with high speed, then the motor is at state $q_{r,4}$. If the motor moves down with low speed, then the motor is at state $q_{r,5}$. If the doors of all floors are closed, then the state of the doors is denoted by $q_{d,1}$. If the door of at least one floor is open, then the state of the doors is denoted by $q_{d,2}$. The set of the states of the doors is $Q_d = \{q_{d,1}, q_{d,2}\}$. According to the above, the set of the states of the elevator system is $Q_e = Q_l \times Q_r \times Q_d$. The members of Q_e are of the form $q_{i,\eta,k} = (q_{l,i}, q_{r,\eta}, q_{d,k})$, where $i \in \{0, 1, \ldots, m\}$, $\eta \in \{1, \ldots, 5\}$ and $k \in \{1, 2\}$.

The event $e_{l,i}$, $i \in \{0, 1, \ldots, m\}$, is generated when the downside of the car comes in contact with the level sensor of the i-th floor. The set $E_l = \{e_{l,0}, \ldots, e_{l,m}\}$ is the set of the events generated by the level sensors. The event $e_{a,i}$ takes place when the downside of the car is aligned to the respective alignment sensor of the i-th floor. The set $E_a = \{e_{a,0}, \ldots, e_{a,m}\}$ is the set of the events provoked by the alignment sensors. The event e_o is the event taking place when at least one door is open. The event e_{cl} is the event taking place when all doors are closed. The event e_g is the event taking place when the button of the ground floor is pressed. The set $E_{ca} = \{e_{ca,1}, \ldots, e_{ca,m}\}$ is the set of the events taking place when the respective call buttons are pressed. The set $E_s = \{e_{s,1}, \ldots, e_{s,m}\}$ is the set of the events taking place

when the respective send buttons is pressed. The events $e_{e,a}$ and $e_{e,d}$ are generated by the rising edge and falling edge of the emergency signal, respectively. The set of all events of the elevator system is

$$E_e = E_l \cup E_a \cup E_{ca} \cup E_s \cup \{e_o, e_{cl}, e_g, e_{e,a}, e_{e,d}\}.$$

Clearly, $|E_e| = 4m + 7$.

The transitions $\forall (i, \eta, k) \in \{0, 1, \ldots, m\} \times \{1, \ldots, 5\} \times \{1, 2\}$ are

$$f(q_{i,\eta,k}, e_{e,d}) = q_{i,\eta,k}, \quad f(q_{i,1,1}, e_{e,a}) = q_{i,1,1},$$
$$f(q_{i,1,2}, e_{e,a}) = q_{i,1,2}, \quad f(q_{i,2,1}, e_{e,a}) = q_{i,1,1},$$
$$f(q_{i,2,2}, e_{e,a}) = q_{i,1,2}, \quad f(q_{i,1,1}, e_o) = q_{i,1,2},$$
$$f(q_{i,1,2}, e_{cl}) = q_{i,1,1}, \quad f(q_{i,\eta,k}, e_g) = q_{i,4,k},$$
$$f(q_{i,\eta,k}, e_{ca,i}) = q_{i,2,k}, \quad f(q_{i,\eta,k}, e_{s,i}) = q_{i,4,k}.$$

The transitions $\forall (i, \eta, k) \in \{0, 1, \ldots, m\} \times \{1, 3, 5\} \times \{1, 2\}$ are

$$f(q_{i,2,k}, e_{l,i}) = q_{i,3,k}, \quad f(q_{i,4,k}, e_{l,i}) = q_{i,5,k},$$
$$f(q_{i,\eta,k}, e_{l,i}) = q_{i,\eta,k}.$$

The transitions $\forall (i, \rho, k) \in \{0, 1, \ldots, m\}^2 \times \{1, 2\}$ are

$$f(q_{i,2,k}, e_{a,\rho}) = q_{\rho,2,k}, \quad f(q_{i,3,k}, e_{a,\rho}) = q_{\rho,1,k},$$
$$f(q_{i,4,k}, e_{a,\rho}) = q_{\rho,4,k}, \quad f(q_{i,5,k}, e_{a,\rho}) = q_{\rho,1,k}.$$

The six-tuple

$$G_e = (Q_e, E_e, f_e, H_e, x_{e,0}, Q_{m,e})$$

is the finite deterministic automaton of the elevator system. The set $H(q_{i,j,k})$ is the set of active events at each state $q_{i,j,k}$ and is determined by the transition function. The initial state is $x_{e,0} = q_{1,1,1}$. All states of the automaton are marked, i.e., $Q_{e,m} = Q_e$. Clearly, $L(G_e) = L_m(G_e)$. For the shake of brevity, the analytic expression of the system's closed behavior is not presented. The set of the uncontrollable events is $E_{uc} = \{e_{e,a}, e_{e,d}\}$. The set of the controllable events is $E_c = E - \{e_o, e_{cl}, e_{e,a}, e_{e,d}\}$.

4 Desired behavior and desired languages

According to [11,15], the formulation of the design specifications in the form of regular languages is one of the two possible alternatives. The second is the formulations of the specifications in the form of desired automata. Clearly, the use of regular languages is the most direct formal translation of natural language rules. The use of regular languages introduces analytic difficulties since it requires the proof of required language properties. On the other hand, the use of regular languages facilitates the realization of low complexity supervisors with predetermined satisfactory performance. Here, the desired behavior will first be presented in the form of rules and then the rules will be translated to regular languages.

4.1 Desired behavior of the horizontal material handling system

The configuration of the manufacturing process of the i-th floor is serial. The manufacturing process of the i-th floor is completed only if the product passes through all machines and uploaded at the Station $2n_i + 2$. The task of the mobile robotic platform is to deliver products from one station to another. It is desired performance is translated as follows: If the robotic platform is at Station $2j$ and there is product availability at Station $2j + 2$, then the mobile robotic platform moves to this station, else the mobile robotic platform moves towards Station $2j + 4$. If the platform is at Station $2n_i + 2$ and there is product availability at Station 1 then the mobile robotic platform moves to Station 1 otherwise the platform moves towards Station 2. This sequence of actions continuous interminably. At the beginning, the mobile robotic platform is stationed at station $2n_i + 2$ (initial state). This aforementioned desired behavior is in accordance to the desired behavior for the single floor processes presented in [12]. To guarantee this desired behavior, the following language is expressed in expanded form

$$^i K_D = \overline{\left(^{i,0}K_D \, ^{i,1}K_D \ldots ^{i,n_i}K_D\right)^*}, \; i \in \{1, \ldots, m\},$$

where the regular expression of the language $^{i,j}K_D$, with $j \in \{0, 1, \ldots, n_i\}$, is proposed to be

$$\begin{aligned}
^{i,j}c_D = {}& ^i e_1 \left(^i e_2 (^i e_1)\right)^* (^i e_{T,2j+2}) + {}^i e_{P,2j+1} (^i e_1) \\
& \times \left(^i e_2 (^i e_1)\right)^* (^i e_{T,2j+1})(^i e_3)(^i e_4)(^i e_1) \quad (1) \\
& \times \left(^i e_2 (^i e_1)\right)^* (^i e_{T,2j+2})(^i e_3)(^i e_4).
\end{aligned}$$

4.2 Desired behavior of the vertical material handling system

The desired behavior of the vertical material handling system, i.e., the elevator system, is analyzed to the following specifications:

– If at least one door is open, then the motor must stay stopped. To satisfy this specification, if one door is open then all button events must be deactivated.
– If the call button of the ground floor or any send button at the rest floors is pressed, then the car must go to the ground floor without intermediate stations.
– If the call button from any floor is pressed, then the car must go directly to the requested floor without intermediate stations. To satisfy this specification, the call and the send buttons of all other floors must be deactivated.
– If the emergency signal is activated while the car moves, then the car must stop immediately and remain stopped until the falling edge of the emergency signal takes place. If the emergency signal is activated and the car is stopped at the level of a floor, then the door of the floor must stay unlocked and the car is not allowed to move until the falling edge of the emergency signal takes place.

The above specifications are in accordance with respective specifications proposed in [13], for a mining process. Here, specifications regarding the load of the car are not considered.

The specifications are analyzed to m languages, one for each floor. The ground floor is excepted. The desired language of the elevator for each floor is $^i K_{D,e} = \overline{^i K} \cap L(G_e)$, where $i \in \{1, \ldots, m\}$ and

$$\begin{aligned}
^i K = {}& \big((E_{ca} - \{e_{ca,i}\} + E_l + E_a)^* (e_o e_{cl} + e_{e,a} e_{e,d} \\
& + (E_s + e_g) e_{l,1} e_{a,1} + e_{ca,i} e_{l,i} e_{a,i})\big)^*.
\end{aligned}$$

Obviously, $^i K \not\subseteq L(G_e)$.

4.3 Properties of the desired languages

According to [11,15], the supervisory control scheme to be successful must comply with two basic guidelines. The first is that in order to guarantee the realizability of the supervisory control scheme with regard to the uncontrollable events of the system, namely a supervisor must not deactivate a transition triggered by an uncontrollable event. This means that the desired language must be controllable regarding the automaton of the system. The controllability of the desired languages will be proved in Lemma 1 and Lemma 2. The second guideline is that the controlled automaton must be nonblocking. This property can be satisfied by two other properties of the desired languages. First, the desired languages must be closed with regard to the marked behavior of the system. This property will be proved in Lemma 3 and Lemma 4. Secondly, the desired languages must be nonconflicting. This property will be proved in Lemma 5. This property is of essential importance for the superposition of the requirement of the desired behavior of the two languages. The latter requirement will be accomplished through modular supervisory control, see [11,15]. The latter aspect will be defined and clarified in Subsection 5.3, using the results of Subsection 4.3.

Lemma 1 $^i K_D$ *is controllable with respect to* $^i G$.

Proof Recall that $^i K_D = \overline{^i K_D}$ and $^i K_D \subseteq L(^i G)$. Let $w \in {}^i K_D$ obviously it holds that $w \in L(^i G)$. There are two cases. The first is $^i f^{|w|}(^i x_0, w) = (^i q_{1,1}, {}^i q_{2,k})$ and the second is $^i f^{|w|}(^i x_0, w) = (^i q_{1,2}, {}^i q_{2,k})$ where $k \in \{1, 2\}$. The value of the function $^i f^{|w|}(\cdot, \cdot)$ denotes the state of the composite

transition, triggered by the word, $w = e_{j_1} e_{j_2} \ldots e_{j_n}$, where $e_{j_1}, e_{j_2}, \ldots, e_{j_n} \in E^i$, i.e.,

$$^i f^{|w|}(^i x_0, w)$$
$$= {}^i f\big(\cdots \big({}^i f\big({}^i f(^i x_0, e_{j_1}), e_{j_2}\big)\cdots\big), e_{j_n}\big).$$

Next, the deactivation of the transitions triggered by the uncontrollable event $^i e_2$, being the only uncontrollable event of $^i G$, will be examined. In the first case, it holds that $w(^i e_2) \notin L(^i G)$. In the second case, it holds that $w(^i e_2) \in L(^i G)$. According to [11,15], $^i K_D$ is controllable with regard to $^i G$, if $\overline{^i K_D}{}^i E_{uc} \cap L(^i G) \subseteq \overline{^i K_D}$. Hence, to prove controllability of $^i K_D$ with respect to $^i G$, it suffices to prove that $(w(^i e_2) \in \overline{^i K_D}) \wedge (w(^i e_2) \in L(^i G))$. Since $w(^i e_2) \in L(^i G)$ and considering the analytic form of $L(^i G)$, it is observed that it is necessary for the word w to be of the form $w = \hat{w}(^i e_1)$, where $\hat{w} \in \overline{^i K_D}$. From the definition of $^i K_D$ it is observed that the appearance of $^i e_1$ is followed always by $\big((^i e_2)(^i e_1)\big)^*$. Clearly, $^i e_2$ is a prefix of every word that belongs to the regular expression $\big((^i e_2)(^i e_1)\big)^* - \varepsilon$. Hence, $w(^i e_2) \in \overline{^i K_D}$. $\qquad \square$

Lemma 2 $^i K$ *is controllable with respect to* G_e.

Proof Consider the language $^i K$ defined in Subsection 4.2. Let a word belonging to the first part of $^i K$, i.e., $w \in \overline{\big((E_{ca} - \{e_{ca,i}\} + E_l + E_a)^*\big)}$. Clearly, w contains only controllable events. From the form of $^i K$ it holds that $w \in \overline{^i K}$. Let an event e where $e \in E_{uc}$. Since $w e_{e,d} \notin L(G_e)$, it holds that $we \in \overline{^i K} \cap L(G_e)$ only if $e = e_{e,a}$. Also, since $w e_{e,a} e_{e,a} \notin L(G_e)$, it holds that $w e_{e,a} e \in \big(\overline{^i K} \cap L(G_e)\big)$ only if $e = e_{e,d}$. This holds true, since the uncontrollable events $e_{e,a}$ and $e_{e,d}$ take place in $L(G_e)$ and $^i K$ alternatively, i.e., in a way that after the appearance of one of them, its reappearance takes place only if the other uncontrollable event has taken place. Based on the above and the criterion of the controllability, the proof has been completed. $\quad \square$

A language is prefix-closed if it is equal to its prefix closure, see [11,15]. Since $\overline{^i K}$ and $L(^i G_e)$ are prefix-closed and controllable, then according to [15], page 152, the language $^i K_{D,e}$ is prefix-closed and controllable. We are now in position to prove the following Lemmas.

Lemma 3 *The desired language* $^i K_D$ *is* $L_m(^i G)$*-closed.*

Proof According to [11,15], the language $^i K_D$ is $L_m(^i G)$-closed if it is equal to the intersection of its prefix closure and $L_m(^i G)$. Since $^i K_D \subseteq L(^i G)$, $^i K_D = \overline{^i K_D}$ and $L_m(^i G) = L(^i G)$, it is observed that $\overline{^i K_D} \cap L_m(^i G) = \overline{^i K_D} \cap L(^i G) = \overline{^i K_D} = {}^i K_D$. $\qquad \square$

Lemma 4 *The desired language* $^i K_{D,e}$ *is* $L_m(G_e)$*-closed.*

Proof According to [15], page 155, the prefix closure of the intersection of two prefix-closed languages is equal to their intersection. Thus, the following equalities hold true:

$$\overline{^i K}_{D,e} \cap L_m(G_e)$$
$$= \overline{\overline{^i K} \cap L(G_e)} \cap L_m(G_e)$$
$$= \overline{^i K} \cap L(G_e) \cap L_m(G_e) = {}^i K_{D,e}.$$

$\qquad \square$

Lemma 5 *The m desired languages* $^i K_{D,e}$*, where* $i \in \{1, \ldots, m\}$ *are non-conflicting.*

Proof According to [11,15], two or more languages are non-conflicting if the prefix closure of their intersection is equal to the intersection of their prefix closures. Also, according to [15], page 155, the prefix closure of the intersection of two prefix-closed languages is equal to their intersection. Thus, $^i K_{D,e} = \overline{^i K}_{D,e}$ and consequently

$$\bigcap_{i=1}^m {}^i K_{D,e} = \bigcap_{i=1}^m \overline{^i K_{D,e}} = \bigcap_{i=1}^m \overline{^i K}_{D,e}. \qquad \square$$

4.4 The total desired behavior

To present the total desired behavior, two projection will be defined. Let $^i P_D$ be the projection of E^* to $^i E^*$ and $P_{D,e}$ be the projection of E^* to E_e^*, where $E = E_e \cup \bigcup_{\lambda=1}^m {}^i E$. According to [15], page 58, the inverse projection of a prefix-closed language is prefix-closed. Thus, since the languages $^i K_D$ and $^i K_{D,e}$ for $i \in \{1, \ldots, m\}$ are prefix-closed then the languages $^i P_D^{-1}(\overline{^i K_D})$ and $P_{D,e}^{-1}(\overline{^i K_{D,e}})$ are also prefix-closed.

The desired behavior of the total system, namely the manufacturing chains of all floors and the elevator system is expressed in the form of the following total desired language:

$$\Big(\bigcap_{i=1}^m {}^i P_D^{-1}(^i K_D)\Big) \cap \Big(\bigcap_{i=1}^m P_{D,e}^{-1}(^i K_{D,e})\Big).$$

5 Supervisor design

5.1 Local supervisors of the horizontal material handling system

The supervisor, realizing $^i K_D$, is proposed to be

$$^i S = (^i Q_S, {}^i E_S, {}^i f_S, {}^i H_S, {}^i x_{S,0}, {}^i Q_{S,m}).$$

The set of the states of the i-th supervisor is $^i Q_S = \bigcup_{\lambda=0}^{n_i} {}^{i,\lambda} Q_S$, where $^{i,j} Q_S = \{^{i,j} q_{S,1}, {}^{i,j} q_{S,2}, \ldots, {}^{i,j} q_{S,11}\}$ and $j \in \{0, 1, \ldots, n_i\}$. The alphabet of the i-th supervisor is $^i E_S = {}^i E$. The initial state is $^i x_{S,0} = {}^{i,0} q_{S,1}$ and the set of

the marked states is $^i\mathbf{Q}_{S,m} = {}^i\mathbf{Q}_S$. For $j \in \{0, 1, \ldots, n_i\}$, the active events sets are defined to be

$$^i\mathbf{H}(^{i,j}q_{S,1}) = \{^ie_{P,2j+1}, {}^ie_1\},$$
$$^i\mathbf{H}(^{i,j}q_{S,2}) = \{^ie_1\},$$
$$^i\mathbf{H}(^{i,j}q_{S,3}) = \{^ie_2, {}^ie_{T,2j+1}\},$$
$$^i\mathbf{H}(^{i,j}q_{S,4}) = \{^ie_3\},$$
$$^i\mathbf{H}(^{i,j}q_{S,5}) = \{^ie_4\},$$
$$^i\mathbf{H}(^{i,j}q_{S,6}) = \{^ie_1\},$$
$$^i\mathbf{H}(^{i,j}q_{S,7}) = \{^ie_2, {}^ie_{T,2j+2}\},$$
$$^i\mathbf{H}(^{i,j}q_{S,8}) = \{^ie_3\},$$
$$^i\mathbf{H}(^{i,j}q_{S,9}) = \{^ie_4\},$$
$$^i\mathbf{H}(^{i,j}q_{S,10}) = \{^ie_2, {}^ie_{T,2j+2}\},$$
$$^i\mathbf{H}(^{i,j}q_{S,11}) = \{^ie_1\}.$$

For $j \in \{0, 1, \ldots, n_i\}$, the transitions of the supervisor are

$$^i f(^{i,j}q_{S,1}, {}^ie_{P,2j+1}) = {}^{i,j}q_{S,2},$$
$$^i f(^{i,j}q_{S,1}, {}^ie_1) = {}^{i,j}q_{S,10},$$
$$^i f(^{i,j}q_{S,2}, {}^ie_1) = {}^{i,j}q_{S,3},$$
$$^i f(^{i,j}q_{S,3}, {}^ie_2) = {}^{i,j}q_{S,2},$$
$$^i f(^{i,j}q_{S,3}, {}^ie_{T,2j+1}) = {}^{i,j}q_{S,4},$$
$$^i f(^{i,j}q_{S,4}, {}^ie_3) = {}^{i,j}q_{S,5},$$
$$^i f(^{i,j}q_{S,5}, {}^ie_4) = {}^{i,j}q_{S,6},$$
$$^i f(^{i,j}q_{S,6}, {}^ie_1) = {}^{i,j}q_{S,7},$$
$$^i f(^{i,j}q_{S,7}, {}^ie_2) = {}^{i,j}q_{S,6},$$
$$^i f(^{i,j}q_{S,7}, {}^ie_{T,2j+2}) = {}^{i,j}q_{S,8},$$
$$^i f(^{i,j}q_{S,8}, {}^ie_3) = {}^{i,j}q_{S,9},$$
$$^i f(^{i,j}q_{S,10}, {}^ie_2) = {}^{i,j}q_{S,11},$$
$$^i f(^{i,j}q_{S,11}, {}^ie_1) = {}^{i,j}q_{S,10},$$
$$^i f(^{i,j}q_{S,9}, {}^ie_4) = \begin{cases} ^{i,j+1}q_{S,1}, & \text{if } j < n_i, \\ ^{i,0}q_{S,1}, & \text{if } j = n_i, \end{cases}$$
$$^i f(^{i,j}q_{S,10}, {}^ie_{T,2j+2}) = \begin{cases} ^{i,j+1}q_{S,1}, & \text{if } j < n_i, \\ ^{i,0}q_{S,1}, & \text{if } j = n_i. \end{cases}$$

The closed behavior and the marked behavior of the supervisor ^iS are proven to be equal to the desired language

$$L(^iS) = L_m(^iS) = {}^i\mathbf{K}_D.$$

Since $L(^iG) = L_m(^iG)$, $\overline{^i\mathbf{K}_D} = {}^i\mathbf{K}_D$ and the two languages have the same alphabet, the closed and marked behavior of the controlled system resulting after the application of the supervisor ^iS to ^iG are proven to be

$$L(^iS\|^iG) = L_m(^iS\|^iG) = L(^iG) \cap \overline{^i\mathbf{K}_D}$$
$$= \overline{^i\mathbf{K}_D} = {}^i\mathbf{K}_D, \quad (2)$$

where $\cdot\|\cdot$ denotes the synchronous product see [11] (or parallel composition see [15]) of the two (or more) argument automata.

The complexity of an automaton can be described by the triad including the number of the states, the number of the events and the number of the transitions of the automaton of the supervisor, indicatively see [21]. The complexity triad of the supervisor ^iS is

$$(11(n_i + 1), 3n_i + 7, 15(n_i + 1)).$$

Note that the components of the complexity triad of ^iS_e are proportional to the number of machines of the respective floors, thus revealing the low complexity of the supervisor with respect to the main parameters of the problem.

5.2 Supervisors of the vertical material handling system

The supervisor, realizing the language ^iK, is proposed to be

$$^iS_e = (^i\mathbf{Q}_{Se}, {}^i\mathbf{E}_{Se}, {}^if_{Se}, {}^i\mathbf{H}_{Se}, {}^ix_{Se,0}, {}^i\mathbf{Q}_{Se,m}).$$

The set of the states of the supervisor is

$$^i\mathbf{Q}_{Se} = \{^iq_{Se,1}, {}^iq_{Se,2}, {}^iq_{Se,3}, {}^iq_{Se,4}, \\ {}^iq_{Se,5}, {}^iq_{Se,6}, {}^iq_{Se,7}\}.$$

The alphabet of the supervisor is

$$^i\mathbf{E}_{Se} = \mathbf{E}_l \cup \mathbf{E}_a \cup \mathbf{E}_{ca} \cup \mathbf{E}_s \cup \{e_o, e_{cl}, e_g, e_{e,a}, e_{e,d}\}.$$

The initial state is $^ix_{Se,0} = {}^iq_{Se,1}$ and the set of the marked states is $^i\mathbf{Q}_{Se} = {}^i\mathbf{Q}_{Se,m}$. The active event sets are

$$^i\mathbf{H}_{Se}(^iq_{Se,1}) = (\mathbf{E}_{ca} - \{e_{ca,i}\}) \cup \mathbf{E}_l \cup \mathbf{E}_a \cup \mathbf{E}_s \\ \cup \{e_g, e_o, e_{e,a}, e_{ca,i}\},$$
$$^i\mathbf{H}_{Se}(^iq_{Se,2}) = \{e_{cl}\},$$
$$^i\mathbf{H}_{Se}(^iq_{Se,3}) = \{e_{e,d}\},$$
$$^i\mathbf{H}_{Se}(^iq_{Se,4}) = \{e_{l,1}\},$$
$$^i\mathbf{H}_{Se}(^iq_{Se,5}) = \{e_{l,i}\},$$
$$^i\mathbf{H}_{Se}(^iq_{Se,6}) = \{e_{a,1}\},$$
$$^i\mathbf{H}_{Se}(^iq_{Se,7}) = \{e_{a,i}\}.$$

The transitions of the supervisor are

$$^if_{Se}(^iq_{Se,1}, e) = {}^iq_{Se,1},$$
$$e \in (\mathbf{E}_{ca} - \{e_{ca,i}\} \cup \mathbf{E}_l \cup \mathbf{E}_a),$$
$$^if_{Se}(^iq_{Se,1}, e_o) = {}^iq_{Se,2},$$
$$^if_{Se}(^iq_{Se,1}, e_{e,a}) = {}^iq_{Se,3},$$

$$
\begin{aligned}
{}^{i}f_{Se}({}^{i}q_{Se,1}, e) &= {}^{i}q_{Se,4}, \ e \in \left(E_s \cup \{e_g\}\right), \\
{}^{i}f_{Se}({}^{i}q_{Se,1}, e_{ca,i}) &= {}^{i}q_{Se,5}, \\
{}^{i}f_{Se}({}^{i}q_{Se,2}, e_{cl}) &= {}^{i}q_{Se,1}, \\
{}^{i}f_{Se}({}^{i}q_{Se,3}, e_{e,d}) &= {}^{i}q_{Se,1}, \\
{}^{i}f_{Se}({}^{i}q_{Se,4}, e_{l,1}) &= {}^{i}q_{Se,6}, \\
{}^{i}f_{Se}({}^{i}q_{Se,6}, e_{a,1}) &= {}^{i}q_{Se,1}, \\
{}^{i}f_{Se}({}^{i}q_{Se,5}, e_{l,i}) &= {}^{i}q_{Se,7}, \\
{}^{i}f_{Se}({}^{i}q_{Se,7}, e_{a,i}) &= {}^{i}q_{Se,1}.
\end{aligned}
$$

The closed and marked behavior of the supervisor is

$$
L({}^{i}S_e) = L_m({}^{i}S_e) = \overline{{}^{i}K}.
$$

Since $L(G_e) = L_m(G_e)$ and the alphabet of $L(G_e)$ is equal to the alphabet of $\overline{{}^{i}K}$, the closed and marked behavior of the controlled system resulting the application of the supervisor ${}^{i}S_e$ to the automaton G_e are

$$
L({}^{i}S_e \| G_e) = L_m({}^{i}S_e \| G_e) = L(G_e) \cap \overline{{}^{i}K} = {}^{i}K_{D,e}.
$$

The complexity triad of the supervisor ${}^{i}S_e$ is

$$
(7, \ 4m + 7, \ 4m + 11).
$$

Note that the first component of the complexity triad of ${}^{i}S_e$ is independent from the number of floors while the second and the third are proportional to the number of floors, thus revealing the low complexity of the supervisor.

5.3 The modular supervisory control scheme

Since the mathematical models of all the subsystems of the manufacturing process have disjoin alphabet sets, it holds that the synchronous product of the automata is a shuffle [11]. This means that the automata of the horizontal handling system of each floor and the automaton of the vertical handling system do not affect one another. Obviously, the automata of the corresponding supervisors of horizontal handling systems and the vertical handling system do not affect the other subsystems.

Regarding the horizontal material handling system of the i-th floor, the supervisor ${}^{i}S$ is applied only to the automaton ${}^{i}G$ without affecting the automata of the other floors as well as the automaton of the elevator. According to [11,15], the supervisor ${}^{i}S$ is a nonblocking supervisor of ${}^{i}G$ as the language ${}^{i}K_D$ a) is controllable regarding the automaton ${}^{i}G$ and b) is $L_m({}^{i}G)$- closed. Also, as is shown in Subsection 5.1., it holds that $L({}^{i}S \| {}^{i}G) = L_m({}^{i}S \| f {}^{i}G) = {}^{i}K_D$.

Regarding the vertical material handling system, the supervisors ${}^{i}S_e$, where $i \in \{1, \ldots, m\}$, are applied to the

automaton G_e without affecting the automata of the horizontal material handling systems. According to [11,15], the modular supervisory control scheme of supervisors ${}^{i}S_e$, where $i \in \{1, \ldots, m\}$, is a nonblocking modular supervisory scheme of G_e as the languages ${}^{i}K_{D,e}$ where $i \in \{1, \ldots, m\}$, a) are controllable with respect to the automaton G_e (see Lemma 2), b) are $L_m(G_e)$-closed (see Lemma 4) and c) are non-conflicting among themselves (see Lemma 5).

Regarding the application of all supervisors ${}^{i}S_e$ to the elevator system G_e, where $i \in \{1, \ldots, m\}$, it holds that

$$
\begin{aligned}
&L\left(\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| G_e\right) \\
&= L_m\left(\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| G_e\right) = L_m\left(\overset{m}{\underset{i=1}{\|}} \left({}^{i}S_e \| G_e \right) \right) \quad (3) \\
&= \overset{m}{\underset{i=1}{\bigcap}} \left(\overline{{}^{i}K} \cap L(G_e) \right) = \overset{m}{\underset{i=1}{\bigcap}} {}^{i}K_{D,e}.
\end{aligned}
$$

For the derivation of the above equalities, the last relation in Subsection 5.2. has been used, where

$$
\overset{m}{\underset{i=1}{\|}} {}^{i}S_e = {}^{1}S_e \|{}^{2}S_e\| \cdots \|{}^{m}S_e.
$$

Before presenting the main result, the total automaton of the MFM process is presented to be $\left(\overset{m}{\underset{i=1}{\|}} {}^{i}G \right) \| G_e$ while the total supervisor is $\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| \left(\overset{m}{\underset{i=1}{\|}} {}^{i}S \right)$. Thus, the total controlled automaton, resulting after the application of the total supervisor to the total automaton of the MFM process, is

$$
\overset{m}{\underset{i=1}{\|}} \left({}^{i}S \|{}^{i}G \right) \| \left(\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| G_e \right).
$$

Theorem 1 *The closed and the marked behavior of the total controlled automaton are equal to the total desired language, i.e.,*

$$
\begin{aligned}
&L\left(\overset{m}{\underset{i=1}{\|}} \left({}^{i}S \|{}^{i}G \right) \| \left(\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| G_e \right) \right) \\
&= L_m\left(\overset{m}{\underset{i=1}{\|}} \left({}^{i}S \|{}^{i}G \right) \| \left(\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| G_e \right) \right) \\
&= \left(\overset{m}{\underset{i=1}{\bigcap}} {}^{i}P_D^{-1}({}^{i}K_D) \right) \cap \left(\overset{m}{\underset{i=1}{\bigcap}} P_{D,e}^{-1}({}^{i}K_{D,e}) \right).
\end{aligned}
$$

Proof From the properties of the closed behavior of the synchronous product (see [11]), it is observed that

$$
\begin{aligned}
&L\left(\overset{m}{\underset{i=1}{\|}} \left({}^{i}S \|{}^{i}G \right) \| \left(\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| G_e \right) \right) \\
&= \left(\overset{m}{\underset{i=1}{\bigcap}} {}^{i}P_D^{-1} \left(L({}^{i}S \|{}^{i}G) \right) \right) \\
&\quad \cap P_{D,e}^{-1}\left(L\left(\left(\overset{m}{\underset{i=1}{\|}} {}^{i}S_e \right) \| G_e \right) \right).
\end{aligned}
$$

Using the formulas in (2) and (3), it holds that
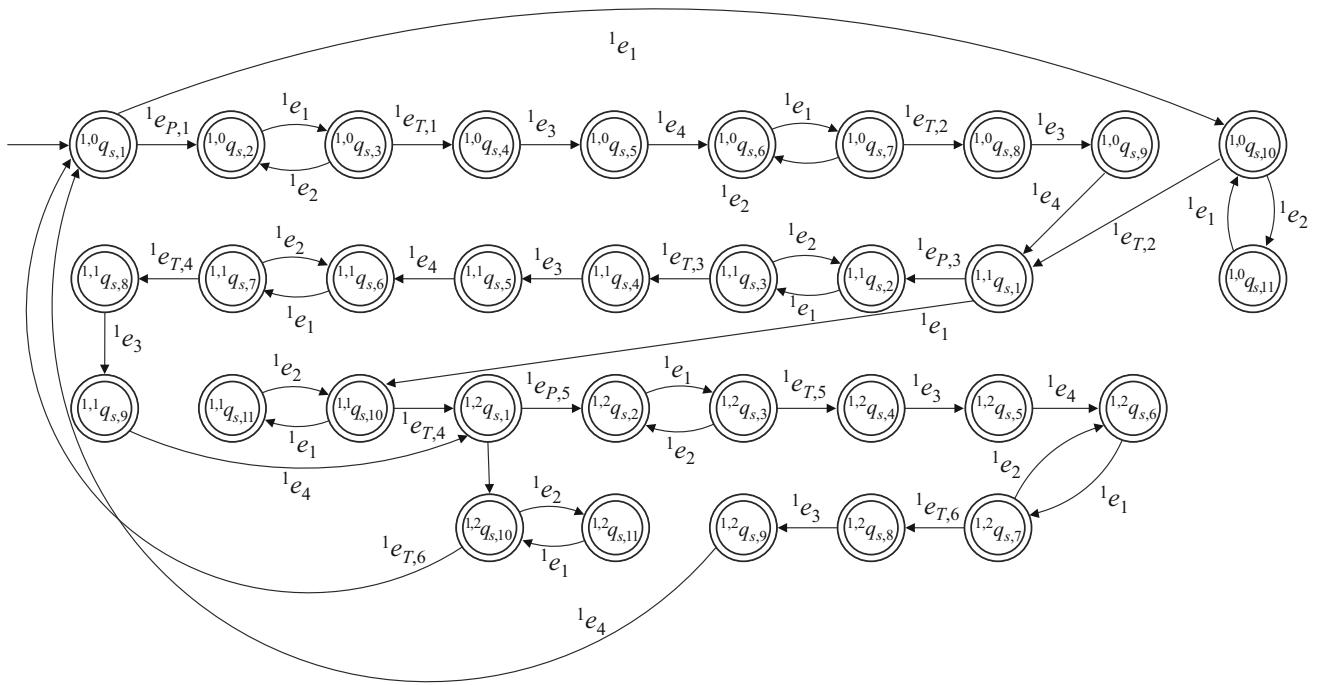
Fig. 3 State diagram of the automaton of the supervisor ^1S_e

$$\left(\bigcap_{i=1}^{m} {}^i P_D^{-1}\left(L({}^i S \| {}^i G)\right)\right) \cap P_{D,e}^{-1}\left(L\left(({\overset{m}{\underset{i=1}{\|}}} {}^i S_e)\|G_e\right)\right)$$
$$= \left(\bigcap_{i=1}^{m} {}^i P_D^{-1}({}^i K_D)\right) \cap \left(P_{D,e}^{-1}\left(\bigcap_{i=1}^{m} {}^i K_{D,e}\right)\right).$$

According to [pp. 58, 11], and since the languages ${}^i K_{D,e}$ are prefix-closed, it holds that

$$\left(\bigcap_{i=1}^{m} {}^i P_D^{-1}({}^i K_D)\right) \cap P_{D,e}^{-1}\left(\bigcap_{i=1}^{m} {}^i K_{D,e}\right)$$
$$= \left(\bigcap_{i=1}^{m} {}^i P_D^{-1}({}^i K_D)\right) \cap \left(\bigcap_{i=1}^{m} P_{D,e}^{-1}({}^i K_{D,e})\right).$$

Finally, it is noted that since all states of the automata and the supervisors are marked, the equality between the closed behavior and the marked behavior is derived. $\quad\square$

The total complexity of the supervisory control scheme proposed here is computed to be of the following analytic form:

$$(18m + 11 \sum_{\lambda=1}^{m} n_\lambda, \, m(4m + 14) + 3 \sum_{\lambda=1}^{m} n_\lambda,$$
$$m(4m + 26) + 15 \sum_{\lambda=1}^{m} n_\lambda).$$

6 Illustrative example

Consider the special case of the MFM process, where the number of floors is $m = 2$, the number of manufacturing machines on the first floor is $n_1 = 2$ and the number of manufacturing machines on the second floor is $n_2 = 3$. The desired languages of the horizontal material handling system are specified as follows:

$$^1 K_D = \overline{\left({}^{1,0}K_D {}^{1,1}K_D {}^{1,2}K_D\right)^*},$$
$$^2 K_D = \overline{\left({}^{2,0}K_D {}^{2,1}K_D {}^{2,2}K_D {}^{2,3}K_D\right)^*},$$

where

$$^{1,0}K_D = {}^1 e_{P,1}({}^1 e_1)\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,1})({}^1 e_3)({}^1 e_4)$$
$$({}^1 e_1)\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,2})({}^1 e_3)({}^1 e_4)+$$
$$^1 e_1\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,2}),$$

$$^{1,1}K_D = {}^1 e_{P,3}({}^1 e_1)\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,3})({}^1 e_3)({}^1 e_4)$$
$$({}^1 e_1)\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,4})({}^1 e_3)({}^1 e_4)+$$
$$^1 e_1\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,4}),$$

$$^{1,2}K_D = {}^1 e_{P,5}({}^1 e_1)\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,5})({}^1 e_3)({}^1 e_4)$$
$$({}^1 e_1)\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,6})({}^1 e_3)({}^1 e_4)+$$
$$^1 e_1\left({}^1 e_2({}^1 e_1)\right)^*({}^1 e_{T,6}),$$

$$^{2,0}K_D = {}^2 e_{P,1}({}^2 e_1)\left({}^2 e_2({}^2 e_1)\right)^*({}^2 e_{T,1})({}^2 e_3)({}^2 e_4)$$
$$({}^2 e_1)\left({}^2 e_2({}^2 e_1)\right)^*({}^2 e_{T,2})({}^2 e_3)({}^2 e_4)+$$
$$^2 e_1\left({}^2 e_2({}^2 e_1)\right)^*({}^2 e_{T,2}),$$
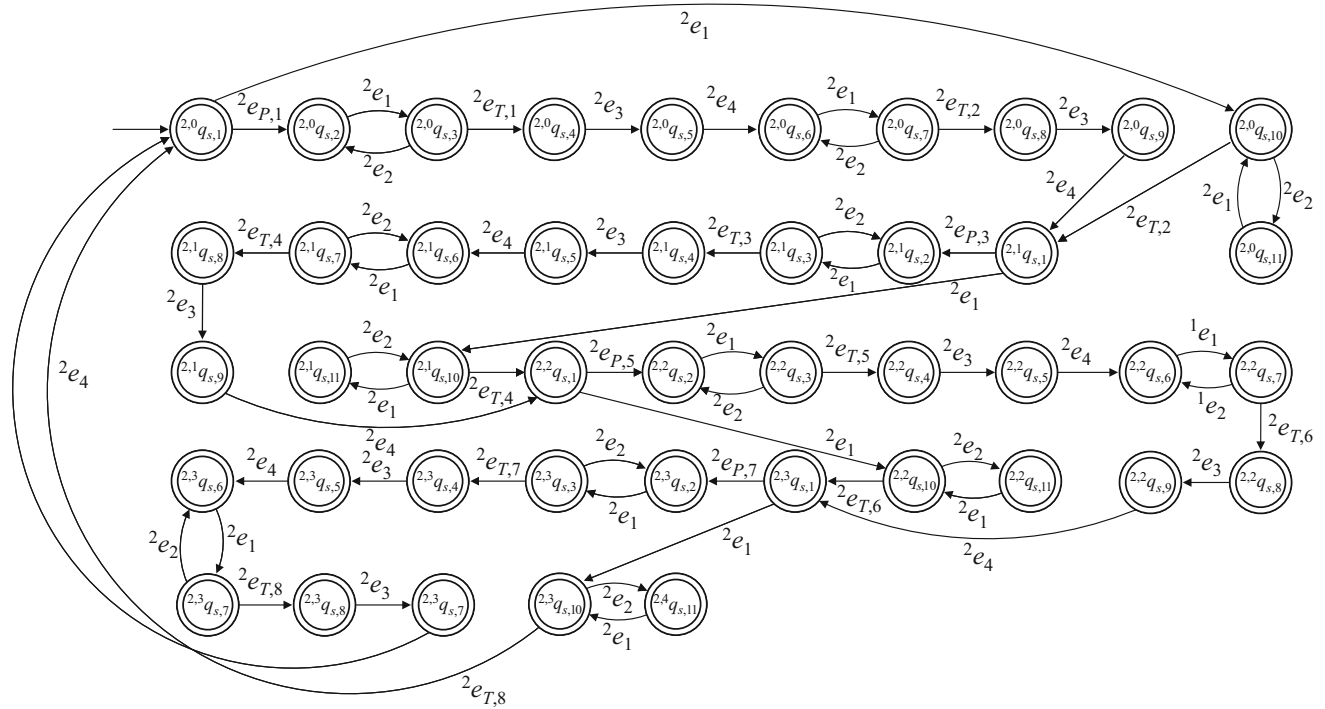
Fig. 4 State diagram of the automaton of the supervisor $^2 S_e$

$$
\begin{aligned}
^{2,1} K_D = {}& ^2 e_{P,3} \left(^2 e_1\right)\left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,3}\right)\left(^2 e_3\right)\left(^2 e_4\right) \\
& \left(^2 e_1\right)\left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,4}\right)\left(^2 e_3\right)\left(^2 e_4\right) + \\
& ^2 e_1 \left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,4}\right),
\end{aligned}
$$

$$
\begin{aligned}
^{2,2} K_D = {}& ^2 e_{P,5} \left(^2 e_1\right)\left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,5}\right)\left(^2 e_3\right)\left(^2 e_4\right) \\
& \left(^2 e_1\right)\left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,6}\right)\left(^2 e_3\right)\left(^2 e_4\right) + \\
& ^2 e_1 \left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,6}\right),
\end{aligned}
$$

$$
\begin{aligned}
^{2,3} K_D = {}& ^2 e_{P,7} \left(^2 e_1\right)\left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,7}\right)\left(^2 e_3\right)\left(^2 e_4\right) \\
& \left(^2 e_1\right)\left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,8}\right)\left(^2 e_3\right)\left(^2 e_4\right) + \\
& ^2 e_1 \left(^2 e_2 \left(^2 e_1\right)\right)^* \left(^2 e_{T,8}\right).
\end{aligned}
$$

For the present special case, the alphabets of the elevator are

$$
E_l = \{e_{l,0}, e_{l,1}, e_{l,2}\}, \ E_a = \{e_{a,0}, e_{a,1}, e_{a,2}\},
$$
$$
E_{ca} = \{e_{ca,1}, e_{ca,2}\}, \ E_s = \{e_{s,1}, e_{s,2}\},
$$

and the desired languages of the vertical material handling system are specified as follows:

$$
\begin{aligned}
^1 K = {}& \big(e_{ca,2} + e_{l,0} + e_{l,1} + e_{l,2} + e_{a,0} + \\
& e_{a,1} + e_{a,2}\big)^* \big(e_o e_{cl} + e_{e,a} e_{e,d} + (e_{s,1} + \\
& e_{s,2} + e_g) e_{l,0} e_{a,0} + e_{ca,1} e_{l,1} e_{a,1})\big)^*,
\end{aligned}
$$

$$
\begin{aligned}
^2 K = {}& \big(e_{ca,1} + e_{l,0} + e_{l,1} + e_{l,2} + e_{a,0} + \\
& e_{a,1} + e_{a,2}\big)^* \big(e_o e_{cl} + e_{e,a} e_{e,d} + (e_{s,1} + \\
& e_{s,2} + e_g) e_{l,0} e_{a,0} + e_{ca,2} e_{l,2} e_{a,2})\big)^*.
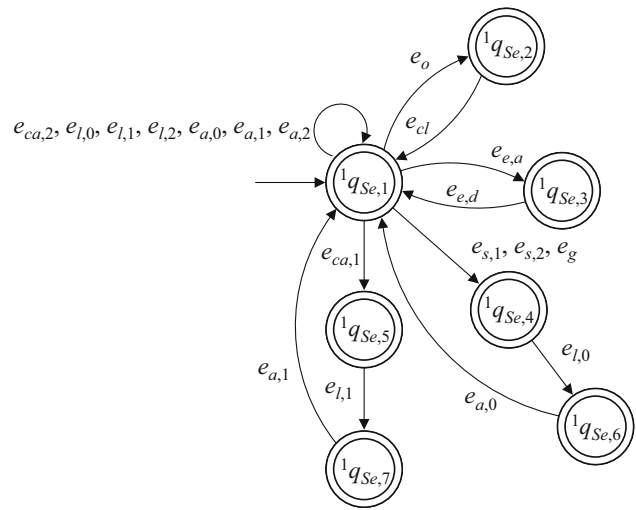\end{aligned}
$$

Fig. 5 State diagram of the automaton of the supervisor $^1 S$

For the present MFM process case, the supervisors $^1 S_e$, $^2 S_e$, $^1 S$ and $^2 S$, are specified as in Figs. 3, 4, 5 and 6, respectively.

7 Conclusion

The coordination in a MFM process has been achieved by the design of appropriate nonblocking supervisors in the form of finite deterministic automata. The model of the MFM process and the modular supervisors are parametric with respect
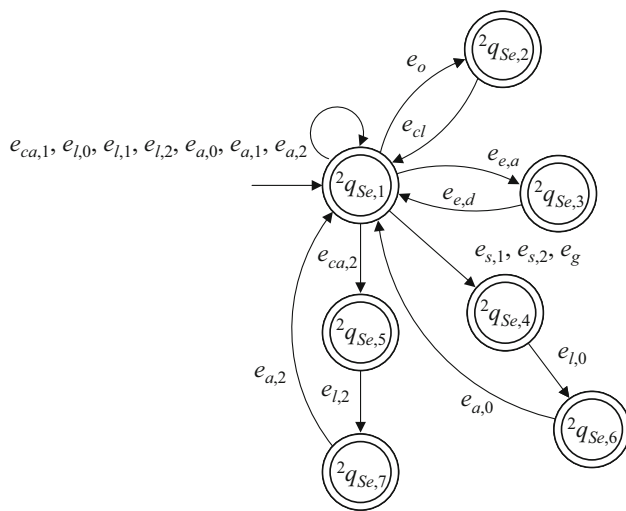
Fig. 6 State diagram of the automaton of the supervisor ^2S

to the number of floors and the number of manufacturing machines on each floor. Thus, generality and flexibility of the model and the controlled performance of the manufacturing process, are provided.

A future perspectives of the present research is the extension of the present results to include the automata describing the manufacturing machines and the automata describing the transfer of products from the machines to the stations and vice versa. Another future perspective is to include models covering the case where possible production faults may take place.

References

1. Dudek, T., Dzhuguryan, T., & Lemke, J. (2019). Sustainable production network design for city multi-floor manufacturing cluster. *Procedia Computer Science, 159*, 2081–2090.

2. Wisnicki, B., & Dzhuguryan, T. (2019). Integrated sustainable freight transport system for city multi-floor manufacturing clusters. *Multidisciplinary Aspects of Production Engineering, 2*(1), 151–160.

3. Fujita, M., & Thisse, J.-F. (2002). *Economics of Agglomeration: Cities, Industrial Localization, and Globalization*. Cambridge, Massachusetts: Cambridge University Press.

4. Lorenzen, M. & Frederiksen, L. (2008). Why do cultural industries cluster? Localization, urbanization, products and projects. *Creative cities, cultural clusters and local economic development*, Edward Elgar, Cheltenham, pp. 155–179.

5. Westkamper, E. (2014). *Towards the Re-Industrialization of Europe. A Concept for Manufacturing for 2030*. Berlin: Springer, Germany.

6. Khaksar-Haghani, F., Kia, R., Mahdavi, I., Javadian, N., & Kazemi, M. (2011). Multi-floor layout design of cellular manufacturing systems. *International Journal of Management Science and Engineering Management, 6*, 356–365.

7. Kia, R., Khaksar-Haghanib, F., Javadianc, N., & Tavakkoli-Moghaddam, R. (2014). Solving a multi-floor layout design model of a dynamic cellular manufacturing system by an efficient genetic algorithm. *Journal of Manufacturing Systems, 33*, 218–232.

8. Bernardi, S., & Anjos, M. F. (2013). A two-stage mathematical-programming method for the multi-floor facility layout problem. *Journal of the Operational Research Society, 64*(3), 352–364.

9. Anjos, M. F., & Vieira, M. V. C. (2017). Mathematical optimization approaches for facility layout problems: The state-of-the-art and future research directions. *European Journal of Operational Research, 261*(1), 1–16.

10. Sua, R., van Schuppen, J. H., & Rooda, J. E. (2012). Maximally permissive coordinated distributed supervisory control of nondeterministic discrete-event systems. *Automatica, 48*(7), 1237–1247.

11. Wonham, W. M., & Kai, C. (2019). *Supervisory control of discrete-event systems*. Springer International Publishing.

12. Koumboulis, F. N., Fragkoulis, D. G., & Kyriazis, N. C. (2016). A two layer supervisor scheme for ring manufacturing production chains. *South East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM),* Kastoria, Greece, pp. 48–52.

13. Koumboulis, F. N., Fragkoulis, D. G., & Michalainas, V. C. (2017). Modular supervisory control for mining process elevator systems. *International Conference on Modern Circuits and Systems Technologies (MOCAST)*, Thessaloniki, Greece. https://doi.org/10.1109/MOCAST.2017.7937614.

14. Ahmad, F., Fakhir, I., Afzal Khan, S., & Daanial Khan, Y. (2014). Petri net-based modeling and control of the multi-elevator systems. *Neural Computing and Applications, 24*, 1601–1612.

15. Cassandras, C. G., & Lafortune, S. (2008). *Introduction to Discrete Event Systems* (2nd ed.). New York: Springer.

16. Wonham, W. M., & Ramadge, P. J. (1988). Modular supervisory control of discrete-event systems. *Mathematics of Control, Signals, and Systems, 1*, 13–30.

17. Vieira, A. D., Santos, E. A. P., de Queiroz, M. H., Leal, A. B., de Paula Neto, A. D., & Cury, J. E. R. (2017). A method for PLC implementation of supervisory control of discrete event systems. *IEEE Transactions on Control Systems Technology, 25*(1), 175–191.

18. Cengic, G., Akesson, K., Lennartson, B., Yuan, C., & Ferreira, P. (2005). Implementation of full synchronous composition using IEC 61499 function blocks. *IEEE International Conference on Automation Science and Engineering*, Edmonton, AB, Canada, pp. 267–272.

19. Koumboulis, F. N., Fragkoulis, D. G., & Ioannou, K. A. (2018). Control of router nodes in production manufacturing processes. *2018 7th International Conference on Systems and Control (ICSC),* Valencia, Spain, pp. 372–377.

20. Koumboulis, F. N., Fragkoulis, D. G., & Menexis, A. N. (2017). Supervisory control for flexibility of production manufacturing processes. *IEEE 21st International Conference on Intelligent Engi-*

neering Systems 2017 (INES 2017), Larnaca, Cyprus. https://doi.org/10.1109/INES.2017.8118550.

21. Guo, L., Vincentelli, A. S., & Pinto, A. (2013). A complexity metric for concurrent finite state machine based embedded software. *8th IEEE International Symposium on Industrial Embedded Systems (SIES)*, Porto, Portugal, pp. 189–195.

Fotis N. Koumboulis is Professor of Robotics and Industrial Automation, Department of Digital Industry Technologies, National and Kapodistrian University of Athens, Greece. He has served as the Director of the Post Graduate Program "Advanced Control Systems and Robotics". He has served as Head of the Department of Automation and Dean of the School of Technological Applications. He has published over 300 scholarly articles in journals, conference proceedings and edited books. His current interests include industrial control, robot control systems, multivariable control, distributed control systems, networked control systems, robust control, supervisory control, switching systems and cyberphysical systems. He received the 1991/92 IEE Mather Premium Award and the 2002/03 Coaless Award. He is included in the Node of Academic and Scientific Excellence of the Ministry of Education, Research and Religious Affairs, Greece. He is a Senior member of IEEE.

Dimitrios G. Fragkoulis is with the Core Department of the National and Kapodistrian University of Athens, where he is Assistant Professor of Modelling, Control and Fault Diagnosis in Production Processes. He holds a Bachelor's degree in Automation from Technological Institute of Piraeus, an M.Sc. degree in Automatic Control, Informatics and Decision Support Systems and a Ph.D. degree in Automatic Control from Paul Sabatier University, Toulouse, France. He has published many scholarly articles in journals and conferences. His current research interests include fault diagnosis, supervisory control, industrial control, and process modeling.

Aristides A. Michos is with the Core Department of the National and Kapodistrian University of Athens. He holds an M.Sc. degree in "Advanced Control and Robotics" of the National and Kapodistrian University of Athens. He holds a Bachelor's degree in Electrical Engineering from Technological Institute of Patras. His research interests include discrete event systems and supervisory control.