International Journal of Innovative Computing, Information and Control
Volume **8**, Number **10(B)**, October **2012**

ICIC International ©2012 ISSN 1349-4198
pp. 7199–7215

PRODUCT DRIVEN DISTRIBUTED CONTROL SYSTEM FOR AN EXPERIMENTAL LOGISTICS CENTRE

Javier de las Morenas, Andrés García Higuera and Pablo García Ansola

Autolog Labs
School of Industrial Engineering
University of Castilla-La Mancha
Avda Camilo José Cela S/N, 13071, Spain
{ javier.delasmorenas; andres.garcia; pablo.garcia }@uclm.es

Received June 2011; revised January 2012

Abstract. *The ever-growing complexity of production systems has long made centralised factory control unpractical. Distributed solutions, where control responsibilities are shared among different decisional entities, are now common. Different distributed architectures have been proposed in literature, where heterarchical systems, in which no central coordination is required, are also being explored. The use of distributed control systems makes possible to reach flexible and agile solutions that rapidly react to changes in the environment. However, distributed decision-making is usually prone to the unalignment of the system, where the previous defined guidelines of the enterprise are not adequately followed by all decisional entities at all times. This paper presents a distributed control system that is structured into two layers. The first layer is constituted by a deliberative level related to the management of the system, while the second layer contains the reactive level, related to the control of the system in the shop floor. This solution uses an alignment method based on a two-step priority calculation in combination with Radio Frequency Identification. The implementation of this solution has been applied to an experimental platform representing the highly automated distribution centre existing at the facilities of a local bottling company.*
Keywords: Distributed control, RFID, Alignment, Logistics, Experimental platform

1. **Introduction.** An increasing necessity for flexible and reconfigurable solutions is becoming apparent for the coordination between production and distribution. The need for production to adapt to customers requests and the pace in which incoming orders have to be fulfilled, has led to the development and application of new production automation and control paradigms. A promising approach is to consider production entities as a conglomerate of distributed, autonomous, intelligent and reusable units, which operate as a set of collaborating entities [1]. As a result, systems can be formed by collaborative units, which can dynamically interact with each other to achieve both local and global objectives [2]. Following these principles, different approaches have been proposed to define reconfigurable production systems, such as Multi-Agent Systems (MAS) [3], Multi-disciplinary Design Optimization (MDO) [4] and Holonic Manufacturing Systems (HMS) [5].

Enterprises operate in a highly competitive environment, which forces them to improve day by day, in terms of quality, response, agility and flexibility. In the current economic situation, this is crucial in order for companies to stay in business [6]. From the client's point of view, traceability is seen as an added attribute to the base product. For this reason, products without a clear process of fabrication and traceability are becoming almost worthless [7]. Identification technologies, such as RFID, provide the eyes of distribution systems that face a growing necessity to align business processes and to introduce new concepts such as interoperability and integration in the business models [8].

The aim of the present research is the development of more efficient control and management methodologies, supported by the visibility offered by RFID, and focused on its future application in factories with high levels of production and large distribution or logistic centres. As to try different control approaches in a factory under production is impractical, it has been necessary to build an experimental platform, which closely reproduces the facilities of an existing distribution centre, to test these new technologies.

The next section will introduce a definition for the concept of agents to then provide some background for this work and comment on how RFID technology efficiently complements Multi-Agent Systems (MAS). Section 3 presents some aspects of the application of Intelligent Management Systems enhanced with Radio-Frequency Identification (RFID-IMS) for the management of a distribution centre represented on an experimental platform. This experimental platform will be described to then use Section 4 to explain the implementation of its control. It will be explained how the system is aligned and driven by the priority values presented in this same section. Section 5 reports the evaluation of the proposed methodology using discrete-event simulation. Finally, some conclusions are extracted from the performance of this platform in Section 6.

2. **Background.** Agents are the software elements that best describe the always-changing situations that are common in industrial environments. They have the capacity to negotiate and coordinate to achieve a common goal. The use of agents forces the adaptation to new technologies, which should be able to supply the system with the current information necessary to take the required decisions. Product-identification using radiofrequency (RFID) [9] has become the technology of choice to ensure visibility. This way, the control can be performed through negotiation as is typical in agent-based systems; without the traditional drawback of the disparities between the information available for the different control elements (agents).

An agent characterizes a software body with a sturdy, adaptable architecture that can function in different environments or computational platforms in an intelligent and autonomous manner. The agent is also capable of carrying out different objectives through an information exchange about the environment, with other human agents or software agents [10]. Another definition is that proposed by Franklin and Graesser [11], who define an agent as a system situated inside of, and in a part of, an environment, that perceives this environment and acts on it over time, making an effort to achieve its goals and this way, affecting it on what it perceives in the future. In spite of the diverse definitions that can be found in literature, and the slight conceptual differences they may introduce, there is one that has been widely accepted: this is that presented in Wooldridge and Jennings [12] defining an agent as a self-contained problem-solving entity. The characteristic of flexibility implies that these agents have a reactive character – as they can perceive the environment and act to meet the changes in conditions; they also have a pro-active character – as their behaviour is oriented towards the achievement of predefined objectives; and they have social capabilities – as they can interact with other agents through negotiation processes in pursuit of their respective objectives [13]. This negotiation is usually applied to decision-making problems, and has a heterarchical nature; in other words, there is no central decision entity. This feature allows the distribution of complex systems in small parts, making easier the control of the entire system.

A multi-agent system (MAS) can, then, be defined as a set of agents that represent the elements of a system, and are capable of interacting in order to achieve their individual goals, even when they do not have enough knowledge and/or skills to achieve individually their objectives [6]. The spreading of the use of multi-agent systems in many research branches lays on the capability of negotiation they offer. Consequently, the main application of agents is the development of distributed systems. However, obviously, there are also issues to overcome for the final adoption of agent structures, such as information integration, the problem of alignment, real world implementation, robustness, agility, and flexibility.

Different studies on decentralized control systems have been conducted to obtain robust and interconnected systems able to cope with uncertainties. Also, the robustness of multi-agent system under external disturbances has been analyzed. This way, it is possible to face the myopic behaviour [14] presented by agents, and therefore the un-alignment of the systems. In [15] this problem is faced by combining the multi-agent system with a hierarchical coordination. As has been seen un-alignment is a problem that arises commonly during the performance of multi-agent system.

The problem of the integration of the information among planning and manufacturing control and execution in enterprises was dealt with in [16]. In that paper, the information gap problem is considered for the case of a specific manufacturing enterprise located in China. That work proposed a three layers framework driven by the bill of materials in order to achieve a hybrid push-pull control strategy.

There are some initiatives for the implementation of MAS in industry. Some of the most representatives are MAST [17], ADACOR [18], JACK [19] or DACS [20]. Nevertheless, the use of agents in industrial application is still limited due to the complexity of the frameworks, making difficult a real-time performing [21]. These solutions require a considerable processing power, memory and bandwidth. Moreover, it is difficult to find approaches where real control equipment, as Programmable Logical Controllers (PLCs), is taken into consideration. An exception is [22].

Some examples of the recent application of agent technology can be found in literatures, where the different actors involved in a global company (that is, production, handling and distribution) are modelled using agents in order to forecast the behaviour of the company. [23] uses multi-agent systems to design a negotiation based policy for order acceptance following due date negotiation strategies. That work takes also in consideration the study of some variables, such as profit and long-term customer behaviour. Another approach in the negotiation domain has been presented in literature, where the conditions necessary to achieve consensus in a multi-agent system are analysed.

MASs have the inconvenience of an excessive dependency on the availability and reliability of information. Therefore, they can strongly benefit from the deployment of RFID systems and, more specifically, with Radio Frequency Identification enhanced Information System (RFID-IS) – the combination of an information management system with RFID. Thus, RFID-IS technology applied to MASs provides flexibility and intelligent control for its application in production, distribution, storage, etc. Therefore, while the MAS provides on-line control of the system, RFID-IS generates and manages updated information. This new solution, RFID-IS/MAS, can be called: Intelligent Management Systems enhanced with Radiofrequency Identification (RFID-IMS), a detailed description can be found in [24]. This idea was first time introduced by Garcia et al. in [25].

The best way to demonstrate the benefits of using these solutions is to apply them to a real company; but, naturally, this is impractical. Soylemezoglu et al. [26] presented a test-bed to experiment with EPC-IS [27] (a particularization of RFID-IS technology). They integrated the decision-making model with a network, using a 3-level integrated model: controller simulation, distributed controller simulation, and distributed controller simulation with hardware-in-the-loop. Vrba et al. [22] presented a simulation of the integration of RFID technology with a multi-agent system in some scenarios.

As has been shown so far, agents are a promising technology but, unfortunately, there is a reduced number of real implementations in place. In this paper, we present the implementation of MAS-DUO [28], a multi-agent system architecture, on a test bench representing the installations of a real enterprise. Here we focus on the low levels of the architecture, where resides the shop floor control. The problem of alignment between layers, the industrial implementation plus the flexibility and information integration are also analysed in this paper. The next section shows how RFID-IMS is modelled in an experimental platform. Later, additional control questions will also be discussed, so as to make possible the introduction of priority considerations.

3. **RFID-IMS Applied to an Experimental Platform.** Being able to experiment with a similar environment to the real facilities, but controlled in a laboratory is essential [29]. Thus, an experimental platform (Figure 1) has been developed to serve as test-bench for the technologies introduced so far. This experimental platform, which has been set-up in the AUTOLOG laboratory within the UCLM, integrates RFID-IMS technology in a storage and distribution facility with a production area. The design used is very similar to the lay-out in the real plant of a local company that distributes food and beverages.

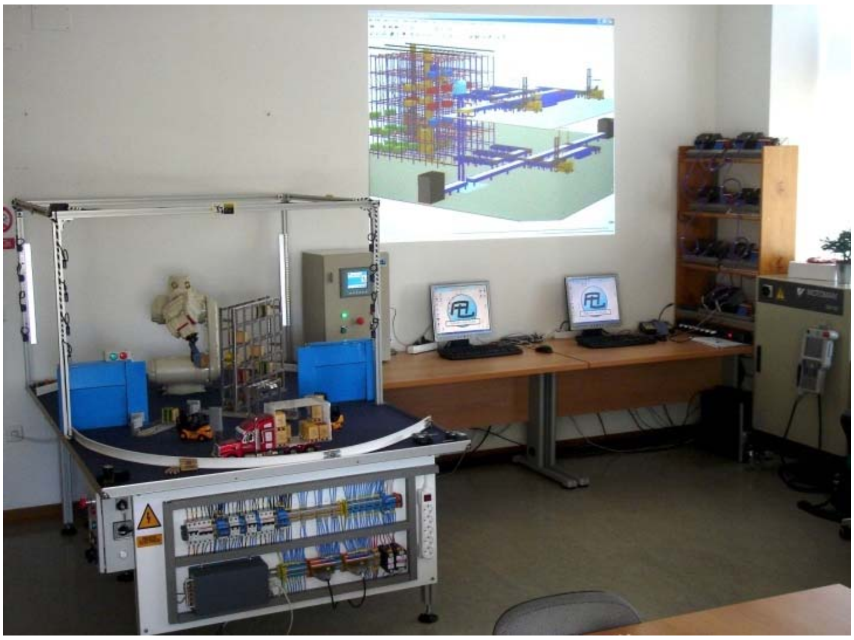

Figure 1. Experimental platform

The layout of the automated warehouse and distribution centre has three corridors served by as many automatic cranes. This system is situated between the factory where production takes place and the docks. The distribution centre is divided into two levels: a ground level by the docks for inputs to the warehouse and an upstairs level for outputs and picking operations.

In this experimental platform a physical part and a simulated one are combined. The simulation – that here will rather be a 3D visualization of a virtual plant – represents the processes of storage and distribution, whereas the physical part does the same for the loading/unloading processes taking place at the docks. To accurately match the real environment, the control philosophies have been tested on a network of Siemens PLCs connected with PROFIBUS DP that has been deployed to control the simulated factory. The system is composed of six PLCs SIMATIC S7 200 CPU 224 which act as slaves, and a SIMATIC S7 300 CPU 313C-2DP acting as master. An additional S7-1200 controls the production zone over Industrial Ethernet. A detailed description of this experimental platform was given in Encinas et al. [30]. The following sub-sections provide an overview of this platform.

3.1. **Miniature physical model.** The physical platform (miniature model) should be understood as the tangible part of the distribution centre, which also allows for a greater flexibility and adaptability of the system. This makes possible the incorporation of real RFID technology and introduces the possibility of errors due to human intervention, as would be the case in a real facility.

The model represents the loading and unloading docks of the distribution centre. It is composed mainly by:

- A set of miniature pallets provided with RFID tags and distributed in four categories/colours: red pallets represent those with high rotation, those with medium rotation are yellow and green stands for those with low rotation while the blue ones represent those that have already been allocated to an order. Blue pallets need to be specially treated because their destination cannot be changed as they may be the result of picking and contain a specific set of different products. Due to practical capacity limits, all pallets with the same rotation category (the same colour) are identical for the miniature part (their Electronic Product Code or EPC is cropped), but every pallet is unique for the 3D visualization (they keep their complete EPC).
- An RFID reader with two antennas. The antennas are situated: one on the input conveyor belt by the entrance, and the other by the exit door of the facilities making it possible to check the load of the outgoing lorries.
- A Motoman HP3 Robot is in charge of managing the miniature pallets that enter the warehouse. It is worth noting that there is no direct relation between the storage of pallets on the physical miniature platform and the simulated model.
- Two conveyor belts. One at the entrance and another at the exit with lateral loading and unloading.
- Two automatic gates, one for entrance and the other for exit to the distribution centre. These doors are provided with open/close detectors.
- A rack or shelf with 36 positions, which is where the miniature pallets are stored.
- Automatic guided vehicles (AGVs) representing the lorries: these are the vehicles in charge of transporting the pallets to and from the system.
- R/C forklift trucks to move loads between the conveyor belts and the AGVs.
- A control panel, formed by actuators, light indicators, buttons and a security system, which assist the operation of the platform.

3.2. **3D visualization.** For a 3D modelling and visualization of the distribution centre, the Grasp10 software by BYG Systems has been used. This software is specifically suitable for the simulation of robots, conveyor belts and any sort of discrete processes. But the main reason to use it is that BYG has provided us with a socket connection module that allows a direct link between the control system and the 3D visualization. This way, the different parts of the factory are actually controlled by their corresponding real PLC.

This visualization has been programmed to accurately reflect a real distribution centre. The simulated virtual factory is divided into: an input area at ground level, an area on the second floor for picking and outputs, and the automated warehouse itself with its three S/R (storage/retrieval) machines (or cranes). Each of these areas and the cranes are controlled by separate S7 200 PLCs. As mentioned, Grasp10 allows communication via sockets. Thanks to this, the simulation works according to the instruction given by the

PLCs through the use of the programmed Java interface, which will be explained later. There are a series of tracks – or sub-modules – in the simulation defining movements. These tracks are always invoked; therefore, they are active at all times and expecting to be launched by PLCs.

4. **Management and Control.** One of the objectives of MASs is for agents to negotiate among themselves to tackle situations of conflict [31]. In this case, RFID-IMS manages the negotiations that take place at specific decision points, such as, conveyor belt junctions, warehouse input/outputs (allocation of Storage Keeping Units or SKUs), and order priority management. It also manages the plan execution, scheduling and planning.

4.1. **Multi-agent architecture.** In practice, for most applications, it is usual to have agents defined from two different viewpoints. Some agents are closely related to the physical world, as they command machinery or moving objects; associated with the reactive agents. While others are more related with management strategies and scheduling, and therefore associated with the deliberative agents. One of the most representative architectures is MAS-DUO [28].

MAS-DUO proposes two autonomous platforms of agents: a physical platform implemented on the physical elements – these agents get the beliefs from the elements of the plant – and the Information System (I.S.) platform that take the beliefs from the RFID readers or other external information sources. This division makes possible for the physical platform to work in an autonomous way by providing robustness in case of errors or communication failures. Based on the concepts defined in [28], a multi-agent architecture has been designed for its application at the Autolog Platform. The system is structured into two levels: reactive and deliberative (Figure 2).

Figure 2. Multi-agent architecture

4.1.1. *Reactive level.* These agents have the ability to perceive changes in the environment and react automatically to them. In this level, the orders are executed without any reasoning. This allows agents to provide fast responses to situations that are critical or too simple to require further analysis. The following agents have been defined taking in mind the lay-out and equipment of the experimental platform.

Production agent: this is the element that controls the production area in the premises. It deals with the completion of the production scheduling and sends the final products to the Distribution Centre. It has been implemented on a PLC S7-1200 capable to produce (in a simulated way) the four different types of items (pallets of the different colours), following the indications of the deliberative level. The production agent receives an order composed by several products and manufactures them accordingly. This production agent starts with the highest rotation products and introduces/manages the corresponding production times before submitting the new pallet to make it appear in the 3D virtual plant. Therefore, once a product is manufactured, the production agent sends it to the distribution centre, controlled by the PNA presented below.

PLC Network Agent (PNA): At this level the PLCs control the physical model and the 3D virtual plant. The PNA receives information from the deliberative level and stores it in data blocks. Floor-level decision-making, traceability of pallets and movement of the loads are made by the PNA. These actions are directly reflected in the 3D virtual plant. The components of this module are:

- Master PLC: This component is responsible for the connection of the PLC's network to the LAN. It manages the incoming data from the deliberate level and assigns it to the corresponding slave PLC. Also, when each pallet of the work order is completed, the master PLC reports this to the ERP agent.
- PLC 1: This component manages the ground level of the simulated plant (conveyors). It communicates with the 3D virtual plant through the Master PLC, acting as a local controller for its zone.
- PLC 2: The same as the previous agent, but for controlling the second floor of the plant (outputs and picking areas).
- PLC 3 to 5: They control each of the storage and retrieval (S/R) cranes in the automated warehouse.
- PLC 6: It controls the physical platform (miniature model) in coordination with the 3D virtual plant.

RFID agent: This module assigns a unique identification to each incoming pallet and sends that identification to the deliberative level. This agent has two RFID readers of passive tags (working under EPCglobal Class 1 Gen 2 standards [32]). It also supplies the required information to check the outgoing orders.

Robot: This component is responsible for the movement of miniature pallets from the dock to the physical shelf. Besides, it manages the SKUs there in coordination with PLC 6.

4.1.2. *Deliberative level.* These agents have the ability to receive/process information about their environment and, according to their beliefs and intentions, decide on the best plan of action to reach the specified objectives. At this level it is possible to identify the following agents:

ERP Agent: This module hosts the processing, questioning, and evaluating of actions that should be taken to reach the objectives. It receives orders from customers, generates the scheduling in coordination with the WMS agent and calculates the priority levels of the orders. This agent has been implemented in Velneo ® 6.4.1 for our platform.

WMS Agent: This agent manages the simulated automatic warehouse. It allocates the pallets in the automatic warehouse shelves according to different factors like rotation grades, priority orders, gathering orders, sales promotion etc.

Tracking Agent: This agent is in charge of following the state of the orders. This way, if there is a delay that renders impossible to complete an order on time, priority levels of each pallet can be modified in real time to complete the order on schedule.

4.2. **Priority values as a mechanism of alignment.** RFID Technology is adopted to make the system capable of identifying all the products arriving and leaving the facilities. Moreover, the deliberative agents (e.g., those related to ERP and WMS) negotiate the location of goods and the priority of production orders supported by RFID information. Priorities are represented as dynamic values reflecting the current guidelines of the management system. These values are calculated including different factors, as customer relevance, delivery date, arriving date, and completeness of orders. The deliberative level assigns different values to these factors and allocates a weight coefficient to them, depending on the policy of the organization. As a result, the system can follow a strategy that gives more importance to a specific client, to overall customer satisfaction, or to the order completeness, among others. Sharing the calculated priority values immediately influences the decision making process at the control system (reactive agents), which tends to follow the upper level strategies, and thereby maintain the system aligned.

4.2.1. *Data packet.* It has been necessary to define a data packet template to make possible the sharing of priority values between agent layers. The data packet is sent to all the elements of the system concerned with processing a specific instruction and it forms the basis of the semantics used in the management and control system. Therefore, every component is capable of reading the related data packet and extracting the necessary information to complete a task. The data packet model represents the instructions generated by the deliberative level. These instructions have a determined structure as shown in Figure 3, in which the necessary data appears coded in an efficient way. A 4-Byte binary code is used to incorporate the identification of the pallet within the control system, the origin and destination of the pallet, a series of flags to specify possible errors that could take place during the fulfilment of the order and an extra space reserved to define the priority of each instruction. This 4-bit priority data reflects the priority calculation detailed in the next section. As can be seen, only the necessary information for the correct operation of the warehouse is incorporated in the instruction/data packet. Each pallet-agent in the installation carries this information.

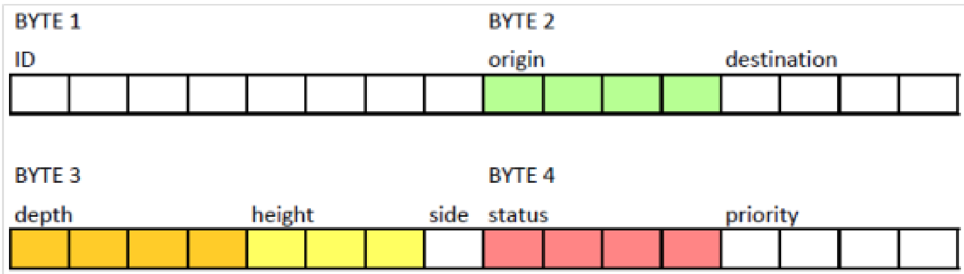

FIGURE 3. Data packet template

4.2.2. *Priority calculation.* The decision of sending a pallet to a certain location, or giving it a specific completion priority, results from the negotiation process between the deliberative agents at the top layer of the MAS architecture. For instance, if a pallet k_{ji} from an order K_j and a pallet k_{rl} from order K_r arrive to a conveyor belt crossing point at the same time, the PLC (bottom layer) performs the decision-making operation based on priority values. Priority values arise from a two-step process.

a) First step for the calculation of the priority

RFID-IMS through its implementation at the ERP at the deliberate level, assigns priority levels to pallets attending to business guidelines. Matrix (1) shows the list of orders and the pallets that compose every one.

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_1 \\ \vdots \\ \mathbf{K}_m \end{bmatrix} = \begin{bmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & & \vdots \\ k_{m1} & \cdots & k_{mn} \end{bmatrix} \tag{1}$$

The priority is determined by Equation (2).

$$P_{k_{ji}} = w_c C_{\mathbf{K}_j} + w_d D_{\mathbf{K}_j} + w_t T_{\mathbf{K}_j} + \frac{\sum_{i=1}^{n} S_{k_{ji}}}{n} \tag{2}$$

where $P_{k_{ji}}$ denotes the priority. $C_{\mathbf{k}_i}$ is the importance degree of a consumer as provided by the company with a value in the range $0 \le C_{\mathbf{k}_i} \le 1$. $D_{\mathbf{k}_i}$ is the value assigned to the delivery date that is defined by the company where $0 \le D_{\mathbf{k}_i} \le 1$. $T_{\mathbf{k}_i}$ denotes the value assigned to the arriving date that is defined by the company with values $0 \le T_{\mathbf{k}_i} \le 1$. The last term of the equation represents the completeness of the order that is obtained by the addition of $S_{k_{ji}}$ factors divided by the total number of pallets. $S_{k_{ji}}$ is a binary variable that is 0 when the pallet k_{ij} has not been shipped, and 1 in other cases.

Finally, w_c, w_d and w_t are weights assigned to $C_{\mathbf{k}_i}$, $D_{\mathbf{k}_i}$ and $T_{\mathbf{k}_i}$ respectively. With this, the priority value can be adjusted to the needs of the organizations. For example, if a company considers that all the costumers are in the same category, then $w_c = 0$. The same may occur with w_d and w_t. The weights may, therefore, have values ranging from 0 to 1 depending on the distribution centre policy. Once deliberative agents negotiate these weights and calculate the corresponding priority, this value is shared with the reactive level.

These equations have been based on a model that calculates the value of reputation/confidence of an agent in a MAS [33]. That model measures the performance of communities of practice in organizations, which are concerned with knowledge sharing by a social approach. On our work, the model has been adapted to calculate the priority level of a pallet inside the community of pallets that complete an order.

The virtual mark V_x that activates the instruction, at the reactive level, to allow pallet k_{ij} to move on to the next process is given by Equation (3).

$$V_x \begin{cases} P_{k_{ji}} \ge P_{k_{rl}} \Rightarrow V_x = 1 \\ P_{k_{ji}} < P_{k_{rl}} \Rightarrow V_x = 0 \end{cases} \tag{3}$$

Therefore, according to this example, the corresponding agent carries out the decision on what pallet has priority in a crossing point based on the values of V_x. This way, an agent will be able to take the adequate decisions by knowing the level of priority associated to each of the pallets in the zone. These values are found in the corresponding data block of the PLC hosting the specific agent. The data can be updated through Ethernet following instructions coming from the deliberative level. This way the system stays aligned through the tuning of control and management layers.

b) Second step for the calculation of the priority

As a first approach for the decision-making process taking place at a conveyor belt junction, a simple comparison of the priorities of the immediately incoming pallets could be a correct solution. However, as can be seen in Figure 4, this procedure could not be the best method taking into account the situation in the plant at that specific instant. The instruction to enter the junction in Figure 4 attending to the priority values given by (3) would be, in that case, processed as follows: First pallet at line 1 (L1), second pallet at L1, first pallet at line 2 (L2) and, finally, the second pallet at L2. As can be seen, the pallet with the highest priority would be the last in crossing the junction when using this simple method. A better solution can be obtained by launching a new negotiation process that would then be the second step of the priority calculation. It consists on a weighting/adjustment of the priority taking into account the complete state of the plant – i.e., considering all the pallets approaching the junction – for what the use of an adapted exponential smoothing equation is proposed.

The exponential smoothing is a technique usually applied in Business Administration to perform forecasts. It estimates the next value of a series of data by applying a constant α to weight the variables in the equation. Taking as model the standard exponential

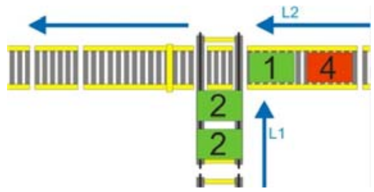

FIGURE 4. Possible situation at a junction showing priority values

smoothing equation [34], our adapted standard exponential is as shown in (4).

$$P^*_{k_{ij}} = (1-\alpha)^{n+1}\frac{n_l}{n_{l\,\max}}P_{\max} + \left[\sum_{i=1}^{n}\alpha(1-\alpha)^{i-1}P_i\right] \tag{4}$$

where $P^*_{k_{ij}}$ denotes the weighted priority; whose value is $0 \le P^*_{k_{ij}} \le 4$. $P_{\max}$ represents the maximum value that can take the priority, in this case 4. P_i is the priority that has the pallet placed at the position i in the conveyor belt L. n_l is the number of pallets loaded in L. $n_{l\,\max}$ is the maximum number of pallets that can be loaded in L. α is the smoothing factor; whose value is $0 \le \alpha \le 1$. According to the value of α, the weighted priority gives preeminence to moving first the pallets with higher priority (high α) or to avoid the congestion of the conveyors belts (low α), so as not to block other junctions upstream.

The correct choice of α is a key factor for the correct completion of orders in the platform. It has been observed that for α equal to 0.4, the system operates near to an optimum performance. However, a fixed α is neither compulsory nor desirable. It is rather recommendable for each agent controlling a conveyor belt junction to have a custom α, attending to its beliefs in order to adequately get its desires or goals.

4.3. **Information flow at the reactive level.** As it was argued before, there are not many MAS implementations over PLCs. This section shows the way in which this has been done for our platform in which a specific instruction is generated for each complete pallet movement – from an origin to a destination – that has to take place at the distribution center. This instruction is sent to the PLC network for its execution, which drives the 3D model. The PLC Master is in charge of collecting the data packets (instructions) coming from the management system and of sending them to the corresponding slave PLC (Figure 5). Additionally, the master is also in charge of collecting messages (instructions at different processing stages) as they are returned by slave PLCs, once their respective parts of the process have been finished, and of redirecting the instructions to the following link in the processing chain (another PLC). Once all execution steps have been processed, the master PLC uses the communication interface to inform the deliberative level about the completion of the instruction (data packets include a status field to report possible incidences). In this way, this master PLC controls and manages the traffic of instructions through the PLC network. Incidentally, for this specific set-up the master PLC is also the link between the virtual I/Os of the PLCs in the network and the interface with the 3D visualization of the simulated plant they have to control.

Figure 5 shows the way in which the master PLC receives and analyzes the incoming instructions and sends them to the specific slave PLC within the Profibus DP network.

A Colored Petri Net (CPN) has been used to model the way in which instructions are sent from the S7-300 (master PLC) to the corresponding S7-200 (slave PLCs) as shown in Figure 6. For each master-slave link the model consists of three parts: S7-300, Profibus-DP, and S7-200. The S7-300 part has two transitions, which can *Send Instructions* and

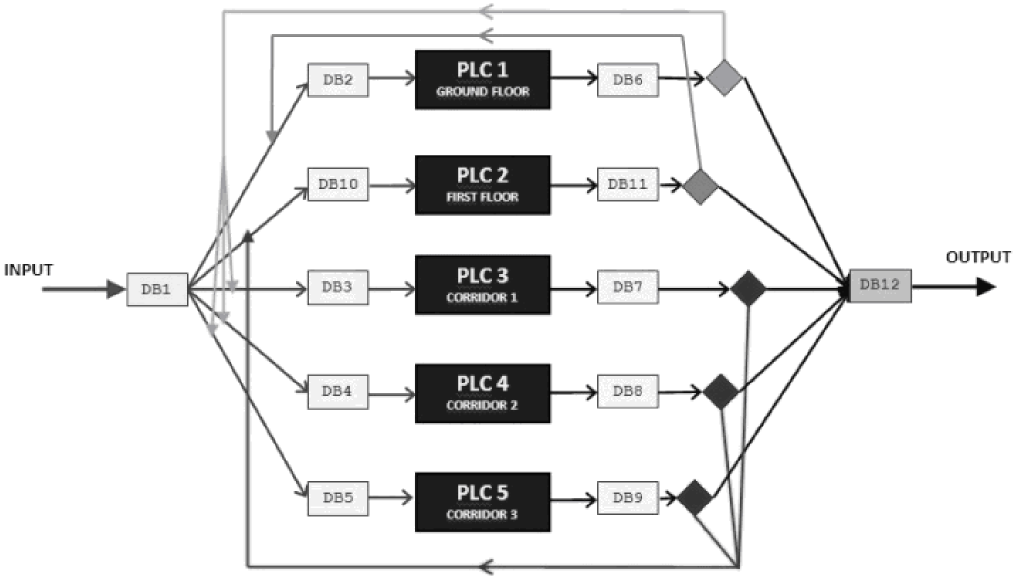

Figure 5. Information flow at the master PLC

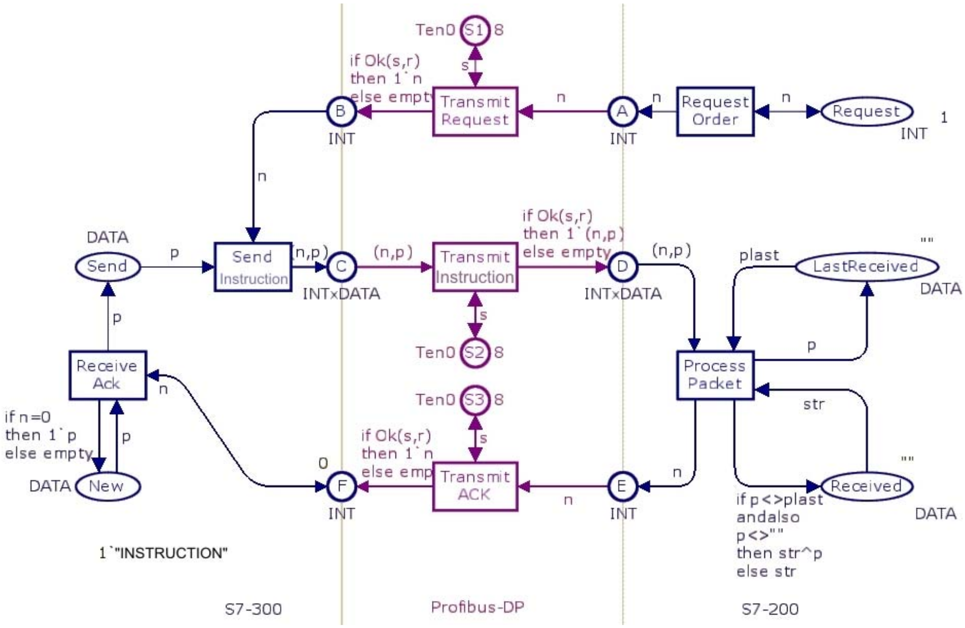

Figure 6. CPN of the communication between PLCs

Receive ACKs (transmission codes for the acknowledgement of a reception). The Profibus-DP network has three transitions which can *Transmit a Request*, *Transmit an Instruction* or *Transmit an ACK*. Finally, the Receiving part has two transitions which can *Process an Instruction* or *Request an Instruction*. The interface between the S7-300 and the Profibus-DP contains places B, C and F (i.e., it can take these estates), while the interface between the Profibus-DP and the S7-200 contains places A, D and E.

Incoming instructions arrive from the deliberative level through place "*New*" and before being transmitted through the *Profibus-DP* network they are stored at place "*Send*". An instruction goes from "*New*" to "*Send*" when the value of the ACK is zero, that is, when there is no pending acknowledgment of reception. An instruction is sent if there is a *Request* from the S7-200 (right-up corner in Figure 6).

Received messages are stored at the place *Process Instruction*. At this point, the incoming instruction is checked to make sure it is different from the last one processed; in which case it is sent to *Received*, where it is stored in the input-job buffer of the S7-200. Then, this S7-200 processes the instruction beginning by checking whether this is a new one through the *LastReceived* place. Any time a new instruction is received, the S7-200 sends and ACK to the S7-300. If the S7-300 receives the ACK, it clears the current instruction from the input buffer and continues with the next one. As can be observed, the next instruction will be sent as soon as the S7-200 makes a request. The fact of erasing an instruction is reflected with transition "*Receive ACK*". This model has been developed using CPN tools [35].

The protocol followed to send an instruction copes with the constraints of the Profibus-DP network by providing a pull strategy for requesting instructions. This way, as a slave S7-200 pulls instructions from the master S7-300, it allows the deliberative level to negotiate the sequence in which instructions are processed until an instruction is requested, enabling a dynamic reconfiguration of the scheduling without interfering in the normal operation of the plant. This helps all the PNAs to perform their role effectively while making them capable of managing unexpected situations.

5. **Evaluation.** Simulation is currently the only way to evaluate the emergent (aggregate) dynamic behaviour of the global agent-based industrial solutions [36]. At this point it is important to note that the word "simulation" has many uses. Whereas before we some times used the words "simulation/simulator" for the software providing the 3D visualization of our virtual plant, we will now be discussing discrete-event simulation. As, in order to demonstrate the effectiveness of the proposed methodology to provide an appropriate use of resources, a thorough analysis has been carried out using the simulation software Witness. This is a simulation environment of the type generally used for performance analysis in industrial environments (Figure 7).

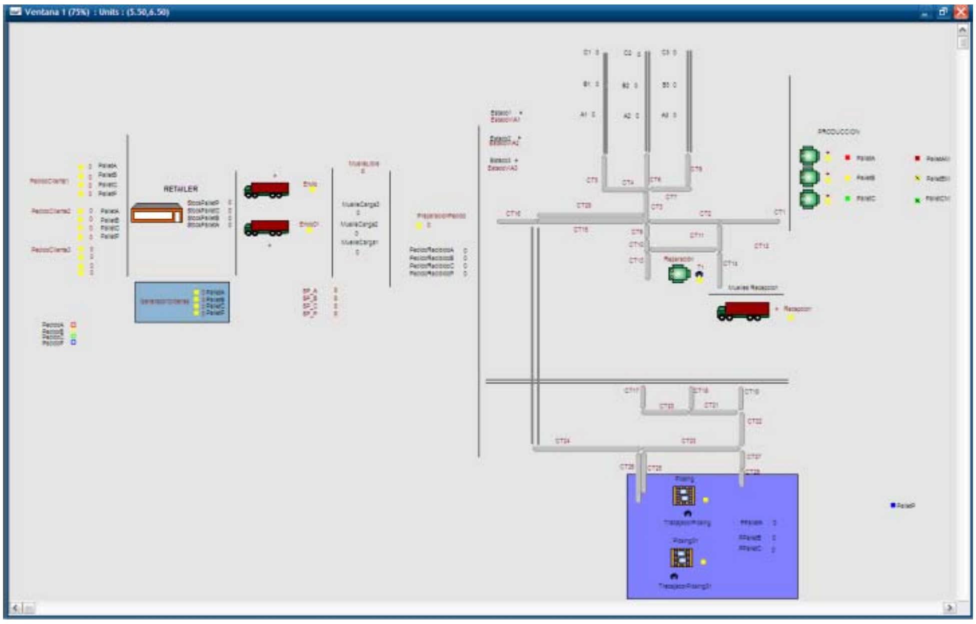

FIGURE 7. Discrete-event simulation in witness

The company is employing for processing the orders the criteria of First Input First Output (FIFO), or First Come First Served (FCFS). These methodologies perform adequately provided that unexpected changes do not occur. However, in current industrial environments this is an unrealistic assumption, given the dynamic behavior of the markets. In Table 1, the characteristics of traditional scheduling and RFID-IMS are summarized.

Table 1. Traditional scheduling vs. RFID-IMS

	Reference Scheduling	RFID-IMS Scheduling
Priority Rule	FIFO	Customized
Priority Value	Fixed	Dynamic
Agility	Limited	High
Throughput	Fixed	Variable

5.1. **Example.** A very simple example is now presented to highlight the effectiveness of the methods proposed here. The following scenario is analyzed for the plant represented in the platform: 3 incoming orders are received having different priority levels. The sequence of arrival for the orders (all in the same day) is: order 1 arrives at 08:00 with low priority, order 2 arrives at 08:30 with intermediate priority, and the last one (order 3) arrives at 09:30 with high priority.

The orders are composed by different amounts of A, B, C and P products. The last one represents the mixed pallets containing A, B and C goods. Table 2 summarizes this situation.

Table 2. Description of the orders

	Order 1	Order 2	Order 3
Units	12A, 8B, 7C, 6P	11A, 9B, 9C, 4P	12A, 10B, 6C, 5P
Incoming date	19/09/11	19/09/11	19/09/11
Incoming hour	08:00	08:15	08:30
Delivery date	19/09/11	19/01/11	19/01/11
Delivery hour	–	–	09:15

The total order processing times (from order arrival to the expedition of goods) are:

– Order 1: 36' 32". Picking: 34,3'
– Order 2: 23' 40". Picking: 23,2'
– Order 3: 30' 39". Picking: 29,5'

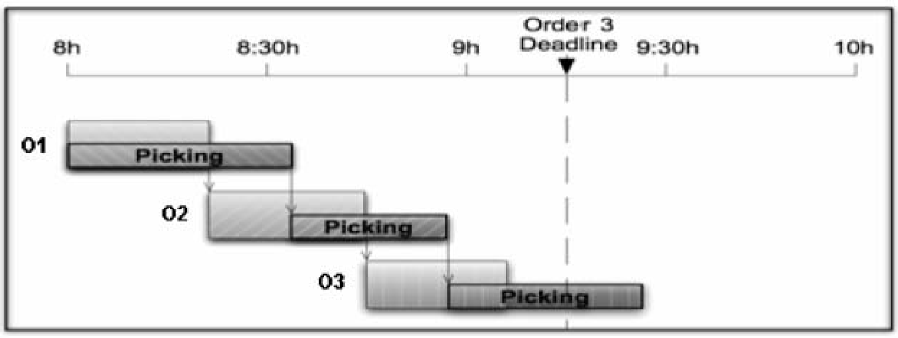

Figure 8. Traditional planning

Figure 8 shows the Gantt diagram for order 3 with a traditional planning method on a FIFO basis. It can be seen that this order is not completed on time.

Figure 9 shows the order processing for the same order but using the methodology proposed here. The following results can be observed:

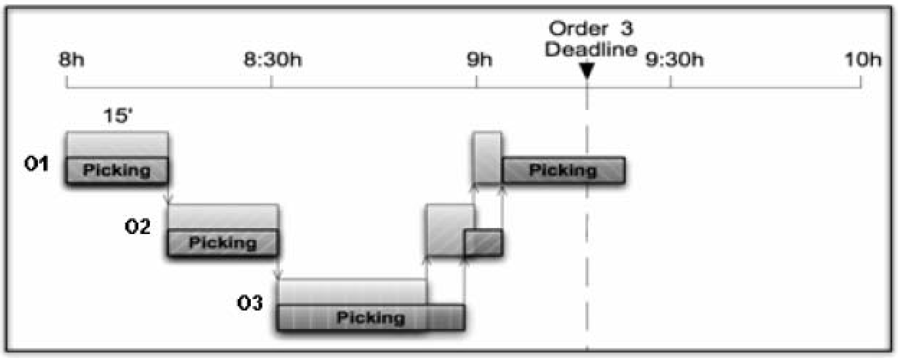

Figure 9. RFID-IMS planning

- There is a decrease in the total elaboration time: from 1 hour and 10 minutes to 1 hour and 2 minutes. Saving around 9% of preparation time.
- It is possible to fulfill the 3rd order within the delivery time.
- Picking is a bottleneck for the installations.

These results are possible due to the decisions made by the management system, RFID-IMS, based on priority levels allocated to pallets and orders. It provides the flexibility necessary for the system to deal with different work orders at the same time. The local control provide by the reactive level of RFID-IMS manage instructions faster than a unique centralized one. A considerable improvement in the use of resources is also achieved.

It can be argued that the traditional methodology used as reference for the comparison is too simple. However, in real systems, the increase in complexity is matched by an increase in the occurrence of perturbations, which a negotiation system is in a much better situation to tackle. As complexity increases, scheduling becomes more difficult and demanding if not faced by a system specifically designed to deal with it.

5.2. **Outcome of the proposed strategy.** By establishing fixed, common, and simple semantics for the definition of the instructions, it has been possible to achieve the required integration and interoperability between agents in the system. The structure defined for the data packets provides an efficient way of sharing information and ensures the correct alignment of the system as, while its composing parts can work independently, they always follow the general politics of the company. This has been achieved by introducing a field to reflect the priority of the instruction that is taken into account along all the decision-making processes and that can be modified dynamically. The deliberative level allocates priorities to orders attending to the guidelines of the company. The reactive level works with the priorities established at the higher level, but adapts them to the real state of the plant. Thus, by sharing a little amount of data, the system is aligned at all levels, from management to operations control. Moreover, the deliberative level acts as a dynamic layer that renegotiates the schedule attending to new orders, guidelines, offers, and delays, among others. This produces as a consequence a change of priorities that spreads dynamically over specific instructions along the whole system to ensure a correct alignment at all times. To achieve this at reactive level, and more particularly on the PLC network, priorities are updated dynamically; which requires the master PLC to keep track of what instruction is in what slave PLC at all times. This way, when the priority of an instruction changes, the master PLC sends an instruction to the specific PLC that is processing that instruction at that specific time, to modify the priority in the corresponding data packet.

The combination of established priorities and RFID provides the system with all the information necessary to track and trace resources and monitor the plant. This improved visibility restrains the deliberative level of overloading the lines of communication with constant messages requesting changes of priorities in the system. This adds efficiency to communications due to the fact that priority modifications are not always effective as, because of the specific lay-out of the plant, it may happen that at some points there is no option for the system to react – not having by-passes and/or the corresponding load being already too close to the output and waiting for external operations. In these cases, communications requesting a change in priority within the system must be avoided and the corresponding external agents notified instead. Placing RFID readers at strategic points that are directly connected to the deliberative level makes communications between deliberative and reactive levels far more efficient.

6. **Conclusions.** A control architecture has been presented that endows the system with an inherent capability to react to unexpected changes in a flexible manner. This reactiveness is achieved through an agent-based control relying on negotiation. The information management system provides accurate information about products, identified by RFID, to all the agents. This way, deliberative agents can negotiate the location and priority of products supported by RFID information. The priorities thus allocated to different operations are dynamic values that reflect the guidelines established for the management system. These values are calculated including different factors, as customer relevance, delivery date, arriving date, and order completion. The deliberative level assigns different values to these factors and allocates a weight coefficient to them, depending on the policy of the organization. As a result, the system can follow a strategy that gives more importance to a specific client, to overall customer satisfaction, or to the completion of orders, among others.

This methodology has been implemented on the control system of an experimental platform. The resulting test-bench is driven by a PLC network where each PLC is in charge of controlling a determined area of the plant. This closely matches the structure of the control system on a real factory in which this test-bench has been inspired. With this platform, it has been possible to analyse at leisure realistic situations that are common in the factory.

Through the performance of this test-bench it has been found that sharing the calculated priority values immediately influences the decision making process at the control system (reactive agents), which tend to follow the upper level strategies and, thereby, maintain the system aligned. In some cases, it is necessary to adapt order priorities to the current state of the installation, so as to avoid bottlenecks. This is mainly done by re-calculating weights again, attending to priorities based on the workload. For this reason, this method has been called: two-step priority calculation. The experimental platform has allowed testing the system in different complex situations while results obtained by discrete-event simulation ratify the efficiency of the proposed method.

Acknowledgment. This work is partially supported by the Spanish Ministry of Education and Science trough grant DPI2011-26438, and to the Consejería de Educación y Ciencia of the Junta de Comunidades de Castilla-La Mancha trough grants PII2I09-0058-9107 and PII1I09-0265-4337. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] M. J. Marco, P. Leitão, A. W. Colombo and F. Restivo, Service-oriented control architecture for reconfigurable production systems, *IEEE Int. Conf. Ind. Informat.*, 2008.

[2] M. Wooldridge, G. Weiss and P. Ciancarini, *Agent-Oriented Software Engineering II, LNCS*, Springer-Verlag, Berlin, Germany, 2002.

[3] M. Pechoučk and V. Mařík, Industrial deployment of multi-agent technologies: Review and selected case studies, *Auton Agents Multi-Agent Syst.*, vol.17, no.3, pp.397-431, 2008.

[4] Z. Ren, F. Yang, N. M. Bouchlaghem and C. J. Anumba, Multi-disciplinary collaborative building design – A comparative study between multi-agent systems and multi-disciplinary optimisation approaches, *Automat. Constr.*, vol.20, no.5, pp.537-549, 2011.

[5] R. Brennan, W. Gruver and K. Hall, Forward-special issue on industrial applications of Holonic manufacturing systems, *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol.41, no.1, pp.1-3, 2011.

[6] P. Leitão, Agent-based distributed manufacturing control: A state-of-the-art survey, *Eng. Appl. AI*, vol.22, no.7, pp.979-991, 2009.

[7] G. Morel, P. Valckenaers, J.-M. Faure, C. E. Pereira and C. Diedrich, Manufacturing plant control challenges and issues, *Control Eng. Pract.*, vol.15, no.11, pp.1321-1331, 2007.

[8] H. Panetto and A. Molina, Enterprise integration and interoperability in manufacturing systems: Trends and issues, *Comput. Ind.*, vol.59, no.7, pp.641-646, 2008.

[9] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, John Wiley & Sons, Ltd., 2003.

[10] F. J. Garijo, Tecnología de agentes: Experiencias y perspectivas para el desarrollo de nuevos servicios y aplicaciones, *Boletin tic nº 24*, 2002.

[11] S. Franklin and A. Graesser, Is it an agent, or just a program: A taxonomy for autonomous agents, *Proc. of the 3rd Internation Workshop on Agent Theories, Architectures, and Languages*, 1996.

[12] M. Wooldridge and N. Jennings, Intelligent agents: Theory and practice, *Knowl. Eng. Rev.*, vol.10, no.2, pp.115-152, 1995.

[13] P. Valckenaers, S. Hadeli, B. S. Germain, P. Verstraete and H. Van Brussel, MAS coordination and control based on stigmergy, *Comput. Ind.*, vol.58, no.7, pp.621-629, 2007.

[14] D. Trentesaux, Distributed control of production systems, *Eng. Appl. AI*, vol.22, no.7, pp.971-978, 2009.

[15] A. Damba and S. Watanabe, Hierarchical control in a multiagent system, *International Journal of Innovative Computing, Information and Control*, vol.4, no.12, pp.3091-3100, 2008.

[16] B. Li and C. Yang, Production logistics system based on a hybrid push/pull control strategy in make-to-order environments, *International Journal of Innovative Computing, Information and Control*, vol.5, no.5, pp.1343-1350, 2009.

[17] M. Merdan, T. Moser, P. Vrba and S. Biffl, Investigating the robustness of re-scheduling policies with multi-agent system simulation, *Int. J. Adv. Manuf. Tecnol.*, vol.55, no.1-4, pp.355-367, 2011.

[18] P. Leitão, A holonic disturbance management architecture for flexible manufacturing systems, *Int. J. Prod. Res.*, vol.49, no.5, pp.1269-1284, 2011.

[19] R. Evertsz, M. Fletcher, R. Jones, J. Jarvis, J. Brusey and S. Dance, Implementing industrial multi-agent systems using JACK, *Program Multi-Agent Syst.*, pp.18-48, 2004.

[20] S. Bussmann, N. Jennings and M. J. Wooldridge, *Multiagent Systems for Manufacturing Control: A Design Methodology*, Springer-Verlag, New York, 2004.

[21] S. Theiss, V. Vasyutynskyy and K. Kabitzsch, Software agents in industry: A customized framework in theory and praxis, *IEEE Trans. Ind. Informat.*, vol.5, no.2, pp.147-156, 2009.

[22] P. Vrba, F. Macurek and V. Marík, Using radio frequency identification in agent-based control systems for industrial applications, *Eng. Appl. AI.*, vol.21, no.3, pp.331-342, 2008.

[23] B. Behdani, A. Adhitya, Z. Lukszo and R. Srinivasan, Negotiation-based approach for order acceptance in a multiplant specialty chemical manufacturing enterprise, *Ind. Eng. Chem. Res.*, vol.50, no.9, pp.5086-5098, 2011.

[24] A. García and A. Cenjor, Sistema heterárquico de control basado en agentes para sistemas de fabricación, *Revista iberoamericana de Automática e Informática Industrial (RIAI)*, vol.4, no.1, 2007.

[25] A. Garcia, A. Cenjor, Y. Chang and M. de la Fuente, An agent-oriented design methodology for RFID improved manufacturing control, *Proc. of ETFA*, Prague, 2006.

[26] A. Soylemezoglu, M. Zawodniok, K. Cha, D. Hall, J. Birt, C. Saygin and J. Sarangapani, A testbed architecture for auto-ID technologies, *Assembly Autom.*, vol.26, no.2, pp.127-136, 2006.

[27] M. Harrison, D. McFarlane, A. K. Parlikad and C. Y. Wong, Information management in the product lifecycle – The role of networked RFID, *IEEE Int. Conf. Ind. Informat.*, pp.507-512, 2004.
[28] P. García Ansola, J. de las Morenas, A. García and J. Otamendi, Distributed decision support system for airport ground handling management using WSN and MAS, *Eng. Appl. AI*, vol.25, no.3, pp.544-553, 2011.
[29] A. Thorne, D. McFarlane, S. Hodges, S. Smith, M. Harrison, J. Brusey and A. Garcia, *White Paper: The AutoID Automation Laboratory: Building Tomorrow's Systems Today*, University of Cambridge. 2003.
[30] J. Encinas, A. García and J. de las Morenas, Improvements in operations management applied to a 3D simulation connected to a physical platform, *J. Intell. Manuf.*, vol.23, no.3, pp.845-856, 2010.
[31] M. J. Wooldridge, *An Introduction to Multiagent Systems*, Wiley, 2002.
[32] GS1, *The EPCglobal Architecture Framework*, http://www.gs1.org/gsmp/kc/epcglobal/architecture /architecture_1_4-framework-20101215.pdf, 2010.
[33] J. Soto, A. Vizcaino, J. Portillo, M. Piattini and O. Kusche, A two-layer multi-agent architecture to facilitate knowledge sharing within communities of practice, *Inteligencia Artificial*, vol.13, 2009.
[34] J. E. S. Gardner, Exponential smoothing: The state of the art – Part II, *Int. J. Forecasting*, vol.22, no.4, pp.637-666, 2006.
[35] K. Jensen, L. M. Kristensen and L. Wells, Coloured petri nets and CPN tools for modelling and validation of concurrent systems, *Int. J. Softw. Tools Tech.*, vol.9, no.3, pp.213-254, 2007.
[36] D. McFarlane, V. Marík and P. Valckenaers, Intelligent control in the manufacturing supply chain, *IEEE Int. Syst.*, pp.24-26, 2005.