Logist. Res. (2011) 3:133–143
DOI 10.1007/s12159-011-0052-y

ORIGINAL PAPER

Agent-based decision-making process in airport ground handling management

Pablo García Ansola · Andrés García Higuera · José Manuel Pastor · F. Javier Otamendi

Received: 31 August 2010 / Accepted: 6 March 2011 / Published online: 26 March 2011

Abstract Recent research on highly distributed control methods has produced a series of new philosophies based on negotiation, which bring together the process engineering with computer science. Among these control philosophies, the ones based on Multi-agent Systems (MAS) have become especially relevant to address complex tasks and to support distributed decision making in asset management, manufacturing, and logistics. However, these MAS models have the drawback of an excessive dependence on up-to-date field information. In this work, a theoretical and experimental MAS, called MAS-DUO, is presented to test new strategies for managing handling operations supported by feedback coming from radio frequency identification (RFID) systems. These strategies have been based on a new distributed organization model to enforce the idea of division between physical elements and information and communication technologies (ICT) in the product scheduling control. This division in two platforms simplifies the design, the development, and the validation of the MAS, allowing an abstraction and preserving the independency between platforms. The communication between both platforms is based on sharing the parameters of the Markov reward function. This function is mainly made up of the field information coming from the RFID readers incorporated as the internal beliefs of the agent. The proposed MAS have been deployed on the Ciudad Real Central Airport in Spain in order to dimension the ground handling resources.

Keywords MAS · Decision making · Airport handling · RFID · Planning

1 Introduction

Airport handling operations involve the sequencing, control, and optimization of the operations related to assist the airplanes at their chosen parking positions. The sequencing tasks are double-fold. On the one hand, medium and long-term scheduling statically determines the staffing requirements, which include both material and human resources dimensions. On the other hand, it is necessary to take short-term and even online decisions whenever deviations or disturbances from plans arise [1]. These assignment tasks are much more related to the review, control, planning, updating of information, and 24-h data availability. Both scheduling tasks are usually centralized at a control department based on software tools with a great lack of field communication. To adapt these tools, new ideas and concepts coming from the intelligent manufacturing systems (IMS) can be applied in distributed planning and decision making [2]. This is the case of the Ciudad Real Central Airport in Spain that was searching for tools that will help with its handling scheduling and control operations.

The focus of this article is to introduce the development framework of a distributed decision support system (DSS) in the assignment of resources to assist incoming flights.

P. García Ansola (✉) · A. García Higuera
School of Industrial Engineering,
University of Castilla-La Mancha, Ciudad Real, Spain
e-mail: Pablo.garcia@uclm.es
URL: http://autolog.uclm.es

J. M. Pastor
Polytechnic School, University of Castilla-La Mancha, Cuenca, Spain

F. J. Otamendi
Applied Economics I Department, Universidad Rey Juan Carlos, Madrid, Spain

More specifically, the part of the application that is highlighted is the development of a decision-making MAS that has been designed to cover the online shop floor necessities prior to the updating of data using RFID. What started as a pure scheduling tool based on traditional techniques in which the centralized control was taking the staffing decisions has evolved into a sophisticated MAS in which the resources and departments are also allowed to participate in a distributed decision-making process. The shift came after understanding that the different resources were competing for the scarce tasks on an airplane basis in today's competing markets in product-driven supply chain [3]. The uncertainties and disturbances of normal operations (flight delays, employees' illnesses, weather conditions, machinery breakdowns etc.) provoked so many deviations and disturbances from the original schedules, deviations that needed to be sorted via an online negotiation process. This new idea transmitted good vibes to the airport staff, although a word of caution arouse due to possible problems at implementation of these new strategies. Software and hardware had to be developed from scratch and particularized for the airport in hand. For that reason, the decision was a pre-validation environment that allowed for testing the proposed MAS and the combination of ICT before the real deployment, which was presented as a demo show in the airport.

What follows on the paper is the explanation of the proposed DSS framework for shop floor scheduling, which will be detailed under several points of view: architecture, modeling, and methodology. The following paragraphs of this section explain new specific schedule requirements because of the appearance of new production models in GH operations, mainly the "common use" model. Section 2 proposes different manufacturing concepts that can be applied to help the current GH situation, and introduces MAS features that will improve the product scheduling control. Section 3 defines the proposed MAS methodology and the steps followed while modeling and during its implementation. Section 4 explains the proposed Markov interaction protocol between agents, and Sect. 5 introduces the results including its real and simulated components in the Ciudad Real Central Airport. Section 6 is used to conclude and discuss about future research opportunities and also about the expected implementation problems.

Modern airports can be extremely complex organizations, often involving governmental organizations, private companies, airlines, aircraft operations, and airport operators [4]. Current situation in the world economic markets have especially forced air transport agents to modify the way they are operating, the results take to light the need to improve productivity and reduce costs by using new strategies. New ideas like "*common use*" are arising in airport operators to gain flexible control over facilities and services, increasing passenger-processing options, and acquiring shared use efficiencies [5]. Airport common usable space is defined as the place in which any airline may operate, that is, there is no space specifically dedicated to any single airline. In this situation, all airline usable airport space is available for use by any airline. The goal of the full "*common use*" model consists of minimizing the amount of time given to a specific airline, as well as maximizing the full use of the airport. Airports benefit from the improved productivity based on the increased utilization of existing resources. The airlines are assigned with no preferences given to any individual airline, which suppose conflicts in the plan of resources in case of resource allocation, delays, or breakdowns. The common use model is business as usual in many airport operations: The implementation results of common use terminal equipment (CUTE), common use self-service kiosks (CUSS), information displays, baggage, or sorting have been successful. [5]. A natural next step of the common use is to be deployed in the GH operations. It is matching up with the increasing of outsourcing GH operations, especially in Europe, which at least define a new agent in the air transport operations. The definition of the GH operations as common use equipment forces a redefinition in the work habits. In order to plan shared resources, the interrelations between all the GH elements are critical and need constant updating from the information flows of the air transport agents: airports, airlines, ATC (air traffic control) and CFMU (central flow management unit). These companies need to share real-time information to be able to define a shared dynamic picture of the airport made up by all the key performance indicator (KPI) elements [6]. The amount of the available data between agents is huge, and its flow is quick and non-stopping: The target off-block time (TOBT), the estimated in-block time (EIBT), the estimated landing time (ELDT), the gate allocation, the taxi way, or the flight plans (Fig. 1). Eurocontrol projects like SESAR (Single European Sky ATM Research Programme) or Airport CDM (collaborative decision making) are spending huge investment in this global integration of data, which also define rules and protocols for sharing data between air transport agents, governments, and regulatory associations. Instead of the company approach, there are also several academic scheduling DSS close to the handling operations, which require to be integrated with the existing systems to establish accurate estimations in in-block/off-block time, gate assignment, or taxi times [7, 8].

Even if this information is accessible, the GH complex schedule processes need computer software and systems put together in place to perform complex calculations and previsions, in order to allow real-time negotiation, to monitor resource usage, and to provide status reporting [9]. In the "*common use*" setting, the GH planning become

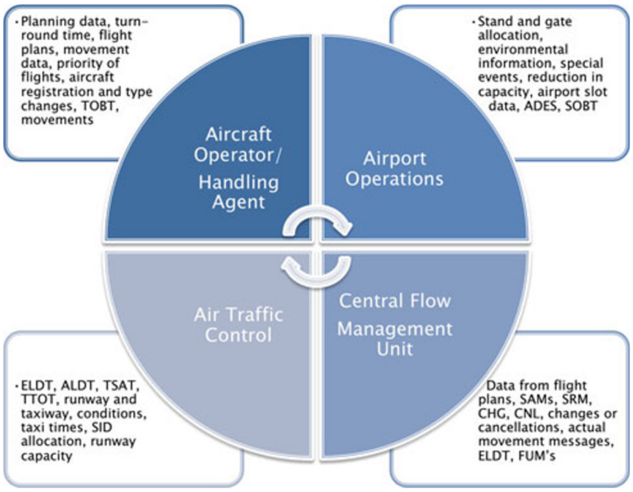

Fig. 1 Information flow in airport operations

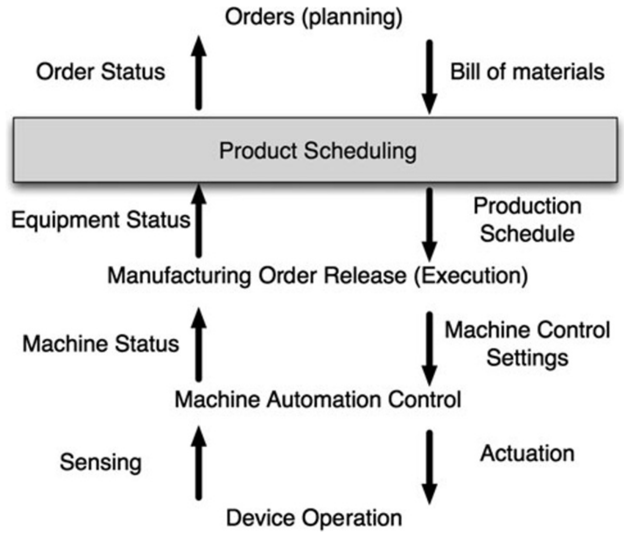

Fig. 2 The central position of product scheduling in the manufacturing control

complex since they have to provide service to several different types of incoming flights, airlines, passengers, or cargo services as is detailed by Dorndorf [10]. Apart from working in a dynamic and shared environment, these companies need to reduce costs while fulfilling the service contract with the airlines in order not to pay fines because of delays. The way to reduce costs is to improve the productivity of the airport by properly scheduling the resources that assist the incoming flights; this idea also goes in the direction of reducing carbon emission by sustainability policies in green airports. This work presents a DSS software framework based on the customization of the manufacturing control to schedule the GH operations.

2 Applying manufacturing concepts in shop floor scheduling control

There are manufacturing concepts like lean strategies, holon, and/or agent-based control [11], product scheduling [12], business to manufacturing (B2M) [13], environmental alignment [14], or distributed control techniques that can be applied to other areas such as shop floor scheduling or asset management. Especially, the manufacturing control and its product schedule can align the shop floor decisions on GH by using the global company policy because of its central position [3] (Fig. 2). The phase of product scheduling is where the plan gets operational and is divided in executable packages as it breaks the long-term policies and orders into scheduled operational decisions at shop floor. This component of the manufacturing control can address the company strategic alignment level in every shop floor decision, which exists when the information system decisions are in concordance with business organization's goals. But to achieve a well-formed organizational DSS, a constant feedback of information is needed, which requires a second level of alignment, called "environmental alignment" [4]. This second alignment level takes into consideration the external disturbances and assumes that the information system (IS) has to integrate real-time features for assessing this environment [5]. This work tackles the lack of alignment in two ways: On the one hand, the strategic alignment can be faced using agent/holonic-based control strategy. And on the other hand, the environmental alignment is achieved by getting an accurate picture of the shop floor by using new visibility frameworks as electronic product code information services (EPCIS).

The combined use of this visibility framework with new paradigms of information management, such as MAS, has been used for short-term planning by several authors [6, 7]. In the present case, the information system is based on a division of the product scheduling level into two agent platforms: One built as an agent structure related to the physical elements of the plant and directly connected to the physical world and the other one as an interface with the upper IS of the company. This division puts special emphasis on applying strategy policies (IS platform) while using information being fed from the environment (physical platform) that includes constant disturbances from the initial schedule. This situation matches scenarios like the common use model on airports. These disturbances at shop floor level require taking decisions not well defined at strategic levels, which produces uncouples in the global decision-making process. This situation can only be addressed when a flexible information update related to disturbances is guaranteed, as is the case when service-oriented architectures like EPCIS are deployed [15]. This junction of technologies is tested in the GH operations due to the specific situation of the sector.

In the required environmental alignment at the shared environment presented in the previous section, three management/decision problems have been detected in the day-to-day planning at the manufacturing product schedule level. Problems that are common in big industries, factories, or distribution centers:

- *The usage of the resources*: It not easy to know when a resource is available or it is busy. The information feedback about the state of the resources is complex with a high dependency of the surrounding elements and physical details.
- *The expected duration of the processes or tasks*: Each one has different lengths depending on the external and internal conditions. The process cannot be completely defined as a mathematical static function or in a template; the real situation is in constant change.
- *The coordination of the involved elements in a task or process*: It is not easy to define the rules of negotiation in a collaborative environment and combine the strategic policies in every single operational decision.

Looking at these problems, there are no mathematical static models that allow for the proper assignment of resources, because there are constant negotiations between different elements, discovery of new services, and real-time disturbances, which cannot be easily reflected or connected on a mathematical function. The resulting classic schedules can incur in delays, over costs, waste of time, unproductive resources, vague previsions, and out of control situations. The need for a flexible reasoning engine in manufacturing/logistics planning has been tested by several authors [16, 17]. Instead of classical tools, new software techniques related to the distributed artificial intelligence (DAI) like MAS are perfectly focused on dynamic reasoning engines like beliefs-desires-intention (BDI) [18], subordinated to a real-time feedback of information [19].

Agent technology is an area of distributed artificial intelligence (DAI) that is already applied in industrial environments with successful results [11]. It is hard to find a definition of the term "*agent*" to which all authors can agree. Fisher [20] defined an agent as an encapsulated entity with traditional AI capabilities; while Jennings and Wooldridge defined an agent as a self-contained problem-solving entity [21]. In general, it can be said that the agents are entities of the software system, which are able to perform autonomous actions in a flexible way in pursuit of established objectives. 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 behavior is oriented toward 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 [18].

A MAS can therefore be defined as a set of specialized agents, which are capable of interacting in order to achieve their individual and cooperative goals, even when they do not have enough knowledge and/or skills to individually achieve their objectives [10]. These properties make the MAS structure applicable to highly dynamic situations, which turn them into promising candidates in providing a management solution [22]. Moreover, software agent technology can monitor and coordinate events and disseminate information [4] improving visibility and creating organizational memories [1]. An important issue is that agents can learn from their own experience, receive information about their environment (keeping updated their beliefs), and adapt themselves to be closer to the solution of the present work at each moment. Unexpected fluctuations or variations in the surrounding environment can be taken into account immediately and acted upon in real time; thus creating an autonomous system that is able to operate without user intervention [5]. The main reasons for using MAS in online scheduling situations are:

- *High capacity of negotiation*: The agents can be published as services in public director facilitators, which allow dynamic interaction driven by discovery services.
- *Quick learning and deployment of decisions*: Each entity (agent) can apply internal pre-planned intentions to the global negotiation, and it usually adds necessary information to a global process of negotiation by using its beliefs-desires-intention (BDI) reasoning [23].
- *Designed for distributed environments*: The agents can be placed in any computer, device, or industrial hardware connected to the network, even in different areas, departments, or companies.
- *Easy to get online real data of the environment*: The agent can be implemented to respond to environmental decisions via hardware connections, web services, or with a specific middleware. In the next section, an integration of the well-known BDI agent with the electronic product code (EPC) infrastructure is presented.
- *Scalability*: The MAS can implement multiple agents in several layered logic platforms, this allows specializing reasoning of each one.

3 MAS-DUO architecture: product-driven BDI reasoning in shop floor scheduling

As a new information system paradigm, agent/holonic-based systems were developed as an extension of previous philosophies but with a high degree of specialization/autonomy in manufacturing. Among the main methodologies used on manufacturing planning, the PROSA methodology by Van Brussel et al. [24], the plant automation based on distributed systems (PABADIS) project of Lüder et al. [25], or ExPlanTech by the Gerstner Laboratory [26] focus the manufacturing decision making using distributed reasoning systems based on an agent approach as our proposed solution. Other agent-based systems appeared as extensions of the knowledge-oriented methodologies such as: The MAS-Common KADS developed by Iglesias et al. that can be found in the book by Singh et al. [27] on intelligent agents; or the CoMoMAS proposed by Glaser [28]. There are also general agent methodologies based on roles like MASE [29] or GAIA [30] based on a macro level (social) and micro level (agent) that could be applied in manufacturing. Another example, proposed by Zhang [31], comes from the supply chain environment and is based on simulation and decision-making tools. All of these developments include the combined use of emerging technologies in manufacturing/logistics as tools for agent-based technologies, like MAS planning [18, 32], Markov decision process application [33], game-theory techniques [34], and Bayesian networks [35]. This list of methodologies highlights that, to apply MAS, ad hoc software and hardware frameworks are necessary to properly address the complexity of the analysis of real systems.

Following most of the previous revised agent-based manufacturing methodologies, real systems tend to match an agent superstructure where two kinds of agents are detected. In the research control approach, it is usual to have agents closely related to the physical world representing assets, products, or resources, as they command machinery or moving objects, while others elements are more related to management strategies, ERPs, or ICTs. So, it can be adequate to maintain that differentiation in the MAS structure as well. Besides, the BDI approach can be a feasible way to implement both types of agents, which form two different agent platforms: physical agents and information system agents. Some models, like PROSA, allow a common definition of all these physical agents (product, resources, and orders at the same physical level). This is an effective initial approach, but for the fact that, after that, the implementation needs a specialization of the reasoning to support interaction with machinery, PLCs, or robots. A service-oriented implementation of this communication with the plant appears to be especially relevant by using MAS features. In the previous section, an easy online feedback of data was emphasized as one of the benefits to use agent-based systems due to its flexibility. This characteristic goes in the way of the well-known problems of lack of field information in the upper logistics and manufacturing levels [13]. This feature is based on the adaptability of the BDI reasoning, which provides a mechanism for separating the activity of selecting a plan from obtaining beliefs of the environment and its execution. The division in two agent platforms allows defining specialized software units that better feed its internal beliefs during the reasoning in several ways. It supposes an adaptation to the real information flows at the manufacturing/logistics environment.

Therefore, based on this capacity of customization, a division of the manufacturing agent spaces is proposed. The agents closer to the physical environment, which can be named BDI physical agents, are most likely directly benefited from visibility frameworks through a software interface as the standard EPC information services (EPCIS) infrastructure. This work only analyses the communication of this platform with RFID frameworks, future proposals will demonstrate its communication with other devices such as PLCs. This type of agents takes charge of the real-time information coming from the plant and its circumstances, reporting treated information to upper levels of agents. In practice, these agents are constituted by sets of software and hardware interfaces defining a specific functionality of the plant. Beliefs of the agent are refreshed by the EPCIS through its XML specification, generating a proper picture of the real state of the resource. With an EPC subscription process, any event of the resource in the tracking area will be submitted to the specific agent. This allows an abstraction with minimum coupling to specific physical details of the environment. The package has the EPC and extra information like position, timestamp, and context. This information can place request for questions such as: What entities are the subjects of the event? When did the event occur? Where did the event occur? And as a conclusion: Why did a particular event happen? For example, in the tested case of the GH operations, the subscription can respond to what GH resources are the subject of the event, the time when took place, the location where the resources are situated on the terminal, and what is the state or the business context of the resource, whether the resources is free, busy, or in transit.

The physical BDI agents constitute a procedural reasoning system (PRS) with an RFID event actualized database. It is a framework for constructing real-time product-driven reasoning systems that can perform decision making in dynamic environments. From a coordination point of view, the physical agent platform is a product-driven approach, where the products compete for shared resources or processes, and this competition requires a negotiation. These low level actions taken by the physical agent after competitive coordination define a plan; these reactive actions have a constant great dependency of the state of the products. Therefore, the static paradigms as object oriented or structure programing seem inadequate because the requirements of constant negotiation and information feedback between the elements. After the information is collected, the decisions during the plan could be based on intelligent techniques as Bayesian networks or Markov decision process. Besides, these intelligent plans can generate new suitable beliefs depending on the chosen options generated in the selection of the plan and its growing knowledge of the system. These new internal beliefs are really useful during the decision process because they are the result of the reasoning process while striving to meet a goal. The obtained beliefs can be directly applied as decision-making parameters and provide information focused to possible solutions, even when using data coming from other operators. Furthermore, these revised beliefs simplify the following deliberations. Some examples of possible internal beliefs are the cost of the plan (financial results), the energy consumed during the selected execution (sustainable policies), or the client valuation in the customer relationship management (CRM). The following algorithm shows the details of the customized BDI reasoning in the physical BDI agents. The first steps are the initialization and the subscription of tracking the linked resource (point 1 and 2). Then, if there is any EPC event, which means a "what?", "where?", "when?", or "why?" variation, the loop executes several instructions: The generation of new options (point 3.a) and a deliberation based on Markov decision process (MDP) (point 3.b). The confirmation and execution of the selected options (point 3.c and 3.d) generates new external events based on its own intentions (point 3.e). And finally, the unsuccessful and impossible attitudes are dropped from the option list of instructions in the next loop (point 3.f and 3.g) (Fig. 3).

On the upper agent level, BDI information system (IS) agents are closer to the management of the company and to the ICT. Their main concerns are not as much related to the physical performance of the plant as with business decisions and strategies. They relate to the fulfilment of orders including resource allocation, order preparation, and plant supply. They feed information systems supplied by the

1. Initialize-state ();
2. Subscription (Electronic Product Code);
3. Repeat while new event
   a. Options: option-generator (instructions-queue);
      i. Reception of new Instructions
      ii. Instruction-queue by Reward ->Where? When? What? Why?
   b. Selected-options: deliberate (options);
      i. New Generated Beliefs -> Cost? Energy? Client? Safety?
      ii. New Reward Selection -> Markov Parameters & New Beliefs
   c. Update-intentions & Confirmation (selected options);
   d. Execution & Results (selected options);
   e. Get-new-external-events ();
      i. Where? When? What? Why?
   f. Drop-unsuccessful-attitudes ();
   g. Drop-impossible-attitudes ();
4. End repeat

Fig. 3 Customized BDI reasoning in product scheduling

management information systems and the processed information coming processed by the physical agents. This platform works as interface between the physical MAS and the manufacturing/logistics ICT like enterprise resource planning (ERP), customer relationship management (CRM), database, warehouse management system (WMS), expert systems.

Based on the division of the organization model shown in the previous section in two platforms of agents, a methodology of MAS development is presented. The methodology combines Commonkads [27], GAIA [36], and the agent unified modeling language (AUML) methods [37]. In GAIA, there are concepts that can be used inside the models of Commonkads like roles, permissions, activities, and properties, and this combination is improved with the interaction diagrams, use cases, and activity diagrams supported by AUML. The complexity of the design of the MAS can be better understood and addressed if the system is subdivided into the following models:

- *Organization model*: The organization model is the basis of the MAS since it includes the definition of the physical platform as well as the information system platform. It therefore affects the behavior of the rest of the models that are software based in nature. Moreover, a correct organization model makes the MAS easier to develop, to test, and to maintain. It represents a divided product scheduling on the manufacturing control (Fig. 4). For example, in GH management test case, all the resources are represented in the physical platform and its state of free or busy. If the new task is to attend an incoming Boeing 737, the business model defined in the task model establishes the relationship between agent and its dependency and coordination. The result of this negotiation in the physical platform is based on the global policy parameters defined by the IS platform, and its results are validated or reject with new parameters by the platform.

- *Agent model*: It specifies the agent characteristics: reasoning capabilities, skills (sensors/effectors), services, agent classes, and hierarchies; in the physical platform, this process is realized with every linked resource. The agent model begins with the specification of its beliefs, desires, and intentions, and then adds new functionalities like roles, permissions, responsibilities, and compositions. This process, called agentification, determines the set of agents that adequately represents the involved elements and their specific environment. The second step consists of identifying and shaping the classes of physical agents; based on the selecting agents, it will be necessary then to identify the basic functional agent groups that will constitute agent

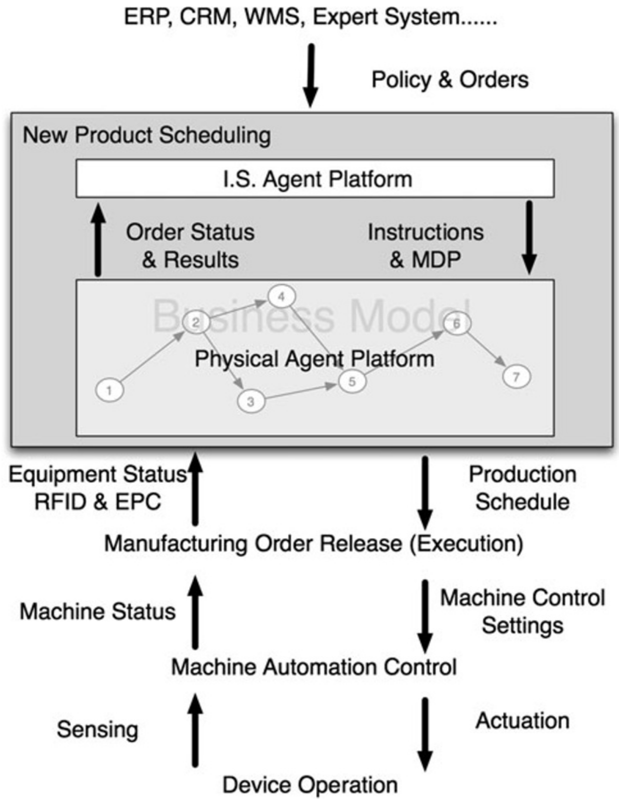

Fig. 4 MAS-DUO model organization. New product scheduling

classes. Agents can be aggregated or specialized depending on their functionality components. This is the current way of work in airport operations where GH resources are classified by their functionality as: stairs, conveyors, carts, employers, tracks, or buses. But inside every class, there are specializations because of its specific attributes; for example in the case of stairs, there are: fast stairs, electrical stairs, driven stairs, or manual stairs.

- *Task model:* It describes the tasks or the activities that the agents can carry out goals or final states, decompositions, and initial states. These tasks define the interaction protocol that will be carried out by the MAS. In the test case, several tasks related to the assistance of the incoming Boeing 373 were defined; it includes subtasks realized by physical agents like luggage handling, air cargo handling, catering trucks, refueling, ground power, or wheelchair lifts.
- *Learning model:* It describes the knowledge needed by the agents to achieve their goals and to gain knowledge; the details of this model will be shown in the following section.
- *Coordination model:* It describes the conversations between agents, their rules, protocols, and required capabilities. The coordination models are well defined by the foundation for the intelligent physical agents (FIPA). The FIPA specifications include agent communication language (ACL), interaction protocols, speech act theory-based communicative acts, and content language representations.

Within each one of the previous models, there are software development tools that can be reused in MAS programing. Currently, the main ideas are being focused in programing languages like Java with a framework for MAS, in order to reuse all the existing powerful libraries with the agent point of view. Some examples are: JADE [38], JADEX [39], JACK (Zeus Agent Toolkit) [40], or the Cougaar agent architecture. As information exchange technologies, there are some approaches to offer a semantic interoperability of agents: ACL, DARPA agent markup language (DAML), and other standards of XML like the web ontology language (OWL) [41].

4 Agent platform interaction protocol based on the Markov parameters

The physical agents, which are in a constant product-driven negotiation, provide their outcomes by using a decision-making technique based on MDP (Markov decision process). This paper proposes an adaptation of the MDP, in which its reward function reflects a strategic decision making in every single operational decision of the physical elements. The value function $V^{\pi}: S \rightarrow \Re$ specifies the value of the policy Π in the state S. This policy will define the decision-making process in every state of the system, even when disturbances occur. This internal reasoning for the selection and execution of the intentions is based on a set of states (s), a set of actions (a) for transaction between states, a collection of probability functions of state transactions (P) that have been successful before, the discount rate (γ), and the reward function (R) defined by the company (Eq. 1).

$$Q^{\pi}(s,a) = R(s) + \gamma \sum_{s \in S} P(s'|s,a) V^{\pi}(s')$$

$$V^{\pi}(s) = \max_{a \in Ap(s)} [Q^{\pi}(s,a)]$$

$$R(s) = \begin{pmatrix} A & B & C & D \end{pmatrix} \begin{pmatrix} \mathrm{Delay}(s) \\ \mathrm{Cost}(s) \\ \mathrm{QoS}(s) \\ \mathrm{Energy}(s) \end{pmatrix} \quad (1)$$

This equation is the basis of the dynamic programing, where function Q represent the reward value of the chosen option in the space of possible actions and all its previously selected steps. Function V is the election of the maximum reward in the space of possible options obtained by function Q. The reward function R is obtained by using the parameters submitted by the ERP systems (A, B, C, D) and the continuously generated beliefs of the physical agents (delay, cost, quality of service, energy). For example, in the GH test case, electrical stairs can obtain a better reward than manual stairs as the required time is shorter, and the cost is adequate to assist an incoming Boeing 737 in the parking 1, following the reward parameters defined by the company.

Combining the BDI and MDP techniques, the internal beliefs situate the agent in one of the defined states in the MDP. The intentions of the agent specify the actions that will define state transitions; they allow the generation of a plan based on the Markov parameters defined by strategic levels. The desires are the goal state or final state on the MDP that the agents want to achieve. Therefore, there is an internal MDP in each physical agent, which chooses those actions with the maximum reward while taking it closer to the goal state. The reward function of the MDP can be customized to the needs of the company, which can be based on the importance of the delays, on the costs of the process, on the importance of the client, on power consumption, or on a combination of all of them; with a specific quantity parameter to reach a selected value in every decision. In the proposed test case, the reward function depends on the value of 4 parameters defined by the company. The average reward is defined as the limit of the sum of rewards divided by the number of states (s) (Eq. 2):

$$\begin{aligned} R(s) &= A * \mathrm{Time}(s) + B * \mathrm{Cost}(s) \\ &\quad + C * \mathrm{QoS}(s) + D * \mathrm{Energy} \\ &\mathrm{Limit\_Reward} : \lim_{T=1}^{S} \frac{\sum_{t=1}^{\infty} R^{(t)}}{T} \end{aligned} \tag{2}$$

With the internal decision policy of the company in mind, if the firm wants to reduce costs it will increase B, and it will give more importance to the cost in every single action, instead of focusing the actions in reducing the delays (A) or improving the results for a specific client (C). With a correct customization of the parameters, the company can define a dynamic decision policy during operations. This method provides well-formed decisions in the operational level with a limited communication with the neighbor nodes of the supply chain management (SCM). Every single decision is validated by the information system platform using the propose interaction protocol defined by FIPA. The results, which are the reward value and the new beliefs, are sent to the ERP for their validation using FIPA proposal messages.

The proposal is negotiated using the contract net protocol between the agents of the I.S. platform. In the test case, there were three agents: the ERP agent, the CRM agent, and an expert system agent. If the proposed solution is accepted, the physical agents execute the plan. Otherwise, the proposal is rejected and the physical agent has to recalculate the plan with new Markov parameters provided by the I.S. platform. The information flow between platforms is detailed in the Fig. 5.

5 Test case in the ground handling management

One test bench of the proposed MAS architecture is an application for the management of the resources that assist the incoming flight at the Ciudad Real Central Airport. On that regard, the proposed architecture is also evolving into a distribution center platform in order to test new ideas about agent-based organization models [42]. This new airport was looking for tools to help the management of resources that are used to provide service to the GH operations. This airport started operations in 2008. To get an idea about the complexity of the scheduling and service problem faced by the GH management of any airport, the large list of tasks that have to be performed for each combination of incoming-outbound flight follows guiding the aircraft into and out of the parking, towing with pushback tractors, lavatory drainage, water cartage, air conditioning, air start units, luggage handling, air cargo handling, catering trucks, re-fueling, ground power, passenger stairs, wheelchair lifts, hydraulic mules, deicing, catering, passenger service, check-in counter services, gate arrival and departure services, staffing the transfer counters, and field operation service. The corresponding physical and human resources that have to be controlled by the system are as follows: chocks, bag carts, dollies for containers and pallets, re-fuellers, tugs and tractors, ground power units, buses, container loader, transporters, air starter, potable water trucks, lavatory service vehicles, catering vehicle, belt loaders, passenger boarding stairs, pushback tugs and tractors, de/anti-icing vehicles, operations (load

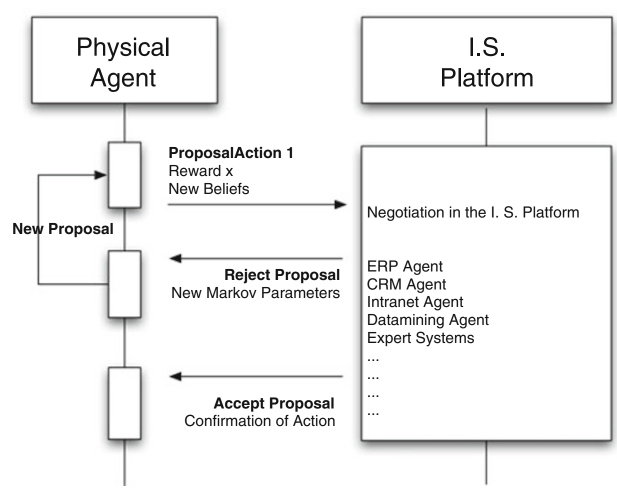

Fig. 5 Platform interaction protocol

control) agent, warehouse agent, crew chief, ramp agent, transfer agent, inbound runner, lavatory agent, mail/freight agent, bag room agent, and drivers.

The proposed software tool allocates and plans the previous mentioned resources to assist incoming flights supported by using RFID and an implementation of MAS-DUO. This MAS-DUO deployment has been developed over Java and JADE in the Eclipse software environment. The data thus obtained are presented in user interfaces that can run/show the simulations in different ways and the possible decisions. This visualization and simulation tool helps the airport management to identify possible conflict situations. The system will communicate the current status of the resources to the visualization tool that supports planning, so that unforeseen changes on the real system will be considered in short-term planning. In the case of the Ciudad Real Central Airport, it has four parking positions for passenger planes; it involves an interface divided into four hubs (Fig. 6). The resources that have to attend the flight surround the aircraft are shown on the GUI, which works as an operation picture to the foreman. On the groundwork, the GUI informs the missing resource to support the process.

The airport management staff fixed the ERP parameters of the reward function to reduce delays, direct costs, and consumed energy. QoS is not represented in the reward function because initially the airport staff did not want to give priority to any specific airline. The reward function is represented in the Eq. 3.

$$R(s) = 0.5 * \mathrm{Time}(s) + 0.4 * \mathrm{Cost}(s) + 0.1 * \mathrm{Energy} \quad (3)$$

The tests that were launched during the demo show were based on the scheduling of a standard workday in the airport. The GH tasks correspond to the definition of processes by the Ciudad Real Central Airport in the attendance of a Boeing B737 during the standard 45-min scale. Besides, the probability of just in time resources was obtained from the information of other Spanish airports as Malaga Airport. The results provided a better fit than the initial dimensioning performed by a major consultancy because there was a reflection of the environmental details in every decision. The comments of the airport staff was that this DSS really takes into account the physical details of the terminal, and it makes possible to place the focus on the accurate dimensioning of the resources with the real requirements of the airport. The comparative of the results

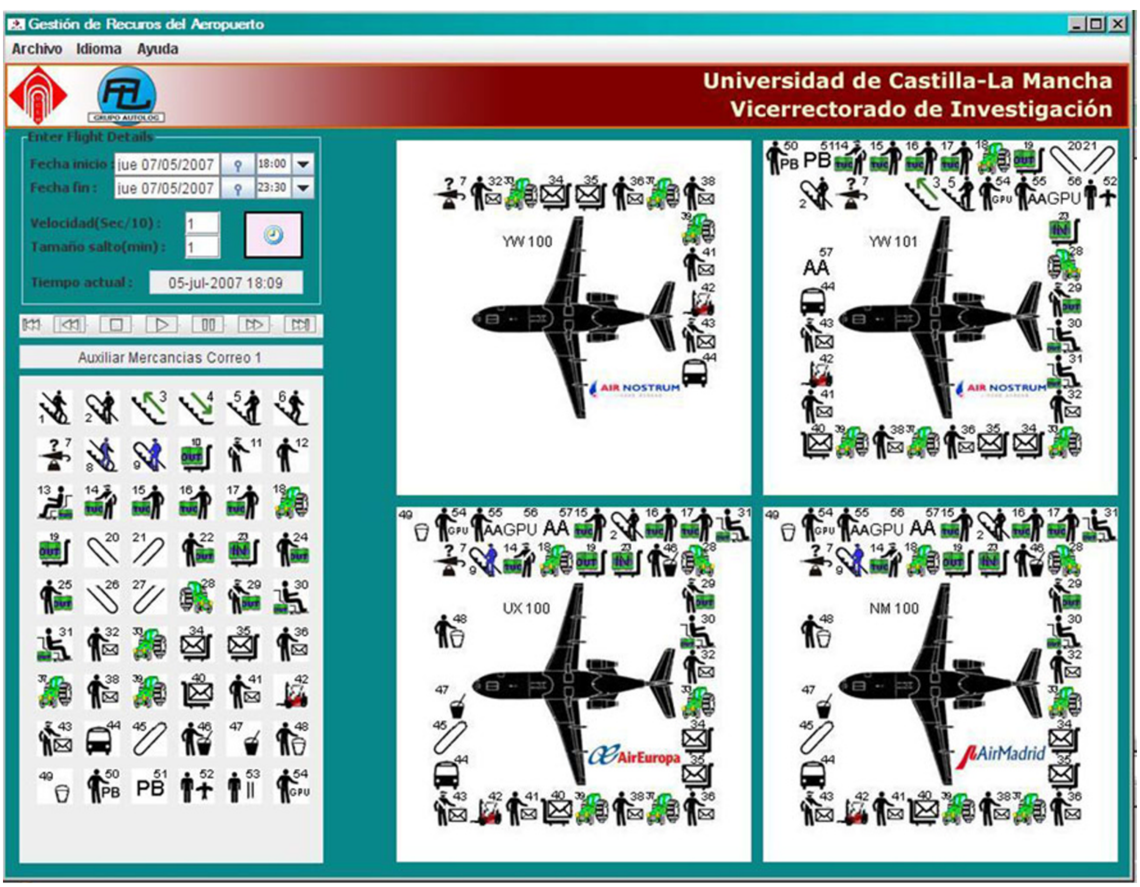

Fig. 6 Ground GUI

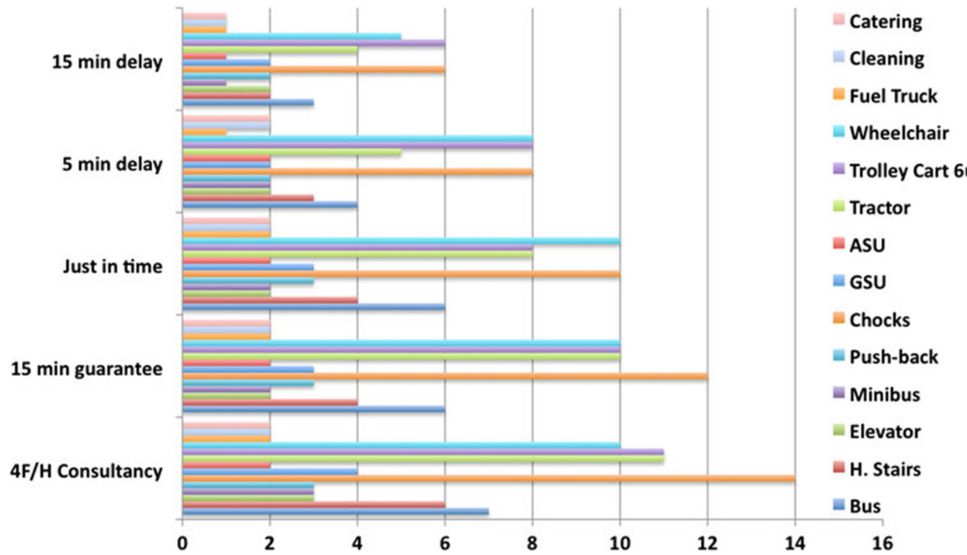

Fig. 7 Results of the GH dimension

based on the usage of resources is analysed in 4 flights per hour, since this is the expected frequency of flights. In Fig. 7, four tests are represented as performed by the MAS, comparing several configurations with the results obtained by the consultancy. The results are divided by several expected delays in the overall attendance process as 5-min guarantee, just in time, or with a specified delay. Before the real deployment, these results help airport management to realize an accurate dimensioning of the required resources in defined situations, so as to be able to attend the incoming flights following the established policy as expressed in the definition of the decision-making process.

These results have been obtained in a small airport, but the scalability of the MAS systems makes possible its deployment in bigger airports. Therefore, the MAS retain its capacity to represent complexity during analysis and development while the well-formed DSS capacity grows exponentially in bigger and more complex cases.

6 Conclusions and future research

Agent-based technologies look appropriate to address complex tasks of planning operations along the supply chain in both the manufacturing and the logistics. The main reasons are as follows: its scalability, its capacity of negotiation, and its ability to implement artificial intelligent techniques. More specifically, the combination of agent based and identification technologies has been considered as a proper way to manage the short-term planning of assigning multiple resources, in highly dynamic environments.

In this paper, a new MAS methodology based on a division of the organization model in two platforms, following the real workflows on the companies, is presented as MAS-DUO. Physical and information system agents are treated separately, facilitating both the understanding of the system, the design and the development of the MAS. This separation allows strategic policies to be reflected on the physical decisions and informs to the upper information system about physical disturbances as well. The problem of communication between platforms is solved using an interaction protocol based on sharing parameters of the Markov reward function. This definition allows an abstraction between platforms, achieving maximum cohesion, and minimum coupling.

One of the first test cases of the MAS-DUO is the GH operations at the Ciudad Real Central Airport. This airport is being used as a test bench to validate the design and development of MAS. The system helps the scheduling duties even when disturbances occur, filling the void of classical scheduling tools that relate more to medium and long-term planning. The article shows the development of such a complete system that was validated by the airport management staff during the demo show.

Acknowledgments This research has been partially supported by the Junta de Comunidades de Castilla-La Mancha, through the research grants PBI08-0267-5500 (Mercury) and PBI06-0152 (AeroLog).

References

1. Schönberger J, Kopfer H (2007) On decision model adaptation in online optimization of a transport system. In: Hans-Otto G, Dirk C.M, Leena S (eds) Management logistischer Netzwerke. Physica-Verlag, HD, pp 361–381
2. Abramovici M, Filos E (2009) Industrial integration of ICT: opportunities for international research cooperation under the IMS scheme. J Intell Manuf :1–8
3. Wang G, Huang S, Dismukes J (2004) Product-driven supply chain selection using integrated multi-criteria decision-making methodology. Int J Prod Econ 91(1):1–16

4. Ashford N, Stanton H, Moore C (1998) Airport operations. McGraw-Hill Professional, New York
5. Belliotti R (2008) Synthesis 8. Common use facilities and equipment at airports. Airport cooperative research program. Transportation Research Board, Washington
6. Thorne A, Barret D, McFarlane D (2007) Impact of RFID on aircraft operations processes. White papers Aero-ID Cambridge. University of Cambridge, Cambridge
7. Atkin J, Burke E, Greenwood J, Reeson D (2008) On-line decision support for take-off runway scheduling with uncertain taxi times at London Heathrow airport. J Scheduling 11(5):323–346
8. Lim A, Wang F (2005) Robust airport gate assignment. ICTAI ′05 Proceedings of the17th IEEE international conference on tools with artificial intelligence (ICTAI′05) Hong Kong, China
9. Eurocontrol (2003) Airport collaborative decision making implementation. Eurocontrol airport operations programme. Eurocontrol, Brussels
10. Dorndorf U (2002) Project scheduling with time windows: from theory to applications. Physica Verlag, New York
11. Leitão P (2009) Agent-based distributed manufacturing control: A state-of-the-art survey. Eng Appl Artif Intell 22(7):979–991
12. Gudehus T, Kotzab H (2009) Planning and scheduling production systems from a logistics perspective. Logistics Research 1(3):163–172
13. Panetto H, Molina A (2008) Enterprise integration and interoperability in manufacturing systems: trends and issues. Comput Ind 59(7):641–646
14. Camponovo G, Pigneur Y, Lausanne S (2004) Information systems alignment in uncertain environments. International conference on Decision Support Systems (DSS) Prato, Italy
15. EPCGlobal (2007) EPC Information Services (EPCIS) Version 1.0.1. EPCglobal Inc
16. Weiming S, Lihui W, Qi H (2006) Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey. Syst Man Cybern C Appl Rev IEEE Trans 36(4):563–577
17. Karageorgos A, Mehandjiev N, Weichhart G, Hämmerle A (2003) Agent-based optimisation of logistics and production planning. Eng Appl Artif Intell 16(4):335–348
18. de Weerdt M, Clement B (2009) Introduction to planning in multiagent systems. Multiagent Grid Syst 5(4):345–355
19. Garcia A, McFarlane D, Thorne A, Fletcher M (2003) The impact of Auto-ID technology in material handling systems. 7th IFAC conference of Intelligent Manufacturing Systems (IMS′03) Budapest, Hungary
20. Burmeister B (1996) Models and methodolgy for agent-oriented analysis and design. Workshop on agent-oriented programming and distributed systems (KI 96) Saarbrücken, Germany
21. Wooldridge M, Jennings N (1995) Intelligent agents: theory and practice. Knowl Eng Rev 10(2):115–152
22. Windt K, Becker T, Jeken O, Gelessus A (2010) A classification pattern for autonomous control methods in logistics. Logistics Res 2(2):109–120
23. Rao AS, Georgeff MP (1995) BDI agents: From theory to practice. Paper presented at the proceedings of the first international conference on Multi-Agent Systems (ICMAS-95), San Francisco, USA
24. Van Brussel HWJ, Valckenaers P, Bongaerts L, Peeter P (1998) Reference architecture for holonic manufacturing systems: PROSA. Comput Ind 37(3):255–274
25. JP LüderA, Sauter T, Deter S, Diep D (2004) Distributed intelligence for plant automation based on multi-agent systems: the PABADIS approach. Production Plannning and Control Taylor & Francis 15(2):201–212
26. Pechoucek M, Riha A, Vokrinek J, Marik V, Prazma V (2002) ExPlanTech: applying multi-agent systems in production planning. Int J Prod Res 40(15):3681–3692
27. Iglesias CA, M. Garijo et al (1996) A methodological proposal for multiagent systems development extending commonkads. Proceedings of 10th KAW (KAW 96) Banoe, Canada
28. Glaser N (1997) The CoMoMAS methodology and environment for multi-agent system development. Multi-Agent Syst Methodol Appl 1286(1997):1–16
29. Zhang C, Lukose D (1998) Multi-agent systems: methodologies and applications. Springer, New York
30. Wooldridgey M, Ciancarini P (2001) Agent-oriented software engineering: the state of the art. Agent-Oriented Softw Eng 1957(2001):55–82
31. Zhang C (2007) Design and simulation of demand information sharing in a supply chain. Simul Model Pract Theory 15(1):32–46
32. Pechoucek M, Rehak M, Charvat P, Vlcek T, Kolar M (2007) Agent-based approach to mass-oriented production planning: case study. Syst Man Cybern C Appl Rev 37(3):386–395
33. Puterman ML (1994) Markov decision processes. Wiley, New York
34. Krajewska M, Kopfer H, Laporte G, Ropke S, Zaccour G (2008) Horizontal cooperation among freight carriers: request allocation and profit sharing. J Oper Res Soc 59(11):1483–1491
35. Shoham Y, Leyton-Brown K (2009) Multiagent systems. Algorithmic, game-theoretic, and logical foundations. Cambridge University Press, New York
36. Wooldridge M, Jennings N, Kinny D (2000) The Gaia methodology for agent-oriented analysis and design. Auton Agent Multi Agent Syst 3(3):285–312
37. Bauer B, Odell J (2005) UML 2.0 and agents: how to build agent-based systems with the new UML standard. Eng Appl Artif Intell 18(2):141–157
38. Bellifemine F, Caire G, Poggi A, Rimassa G (2003) JADE—a white paper. EXP Search Innov 3(3):6–19
39. Braubach L, Pokahr A, Lamersdorf W (2003) Jadex: implementing a BDI infrastructure for JADE agents. EXP Search Innov 3(3):76–85
40. Howden N, Rönnquist R, Hodgson A, Lucas A (2001) JACK Intelligent Agents: Summary of an agent infrastructure. In: Wagner T, Rana O (eds) The 5th international conference on autonomous agents, workshop on infrastructure for agents, MAS and Scalable MAS (Agent 2001) 251–257 Montreal, Canada
41. Haarslev V, Racer MR (2003) An OWL reasoning agent for the semantic web. In: Proc. of the international workshop on applications, products and services of web-based support systems (WSS03), pp 91–95
42. Garcia P, Garcia A, Encinas J, De las Morenas J (2009) Experimental platform for the analysis of the RFID enhanced agent based management system of a distribution center. Paper presented at the MITIP, Bergamo, Italy