Multi-Agent System for On-demand Production Integrating Production and Quality Control

Paulo Leitão[1,2], and Nelson Rodrigues[1]

[1] Polytechnic Institute of Bragança, Campus Sta Apolonia, Apartado 1134,
5301-857 Bragança, Portugal
`{pleitao,nrodrigues}@ipb.pt`
[2] Artificial Intelligence and Computer Science Laboratory, R. Campo Alegre 102,
4169-007 Porto, Portugal

Abstract. Multi-agent systems is being pointed as particularly suited to design and engineer a new class of control systems to operate at the factory plants addressing the current requirements of modularity, flexibility and re-configurability. This paper introduces the main principles of a multi-agent system approach to support the integration of production and quality control processes in washing machines production lines that is being developed under the EU FP7 GRACE project.

Keywords: Multi-agent systems, manufacturing control, product-driven systems.

1 Introduction

Production engineering is evolving rapidly and manufacturing processes are becoming more and more complex, oriented towards customized and flexible production, facing the economic driving forces towards the improvement of production process efficiency and customer demands. This reality is aligned with the strategic research agenda made by the Manufuture European Technology Platform [1] that points out the need for enabling technologies, oriented to flexible and intelligent processes, to address the strong market pressures in terms of cost, quality, customization and time to market. In this emergent and very volatile environment, responsiveness is one of the main challenges on which every manufacturing company stands to be/remain competitive [2]. Under these circumstances, manufacturing companies may consider structures that guarantee the dynamic re-configurability and adaptability at the factory level with a strong emphasis on the goal of realizing fault-free products, fully compliant to specifications, with the aim of improving customer satisfaction. An example of this new manufacturing reality is the sector of home appliances (e.g., washing machines and cooking) that is continuously under the pressure of consumer demand, exhibiting the referred requirements.

The traditional control systems in manufacturing environment are typically large monolithic and centralized applications that do not cope efficiently with the current requirements imposed to manufacturing systems, namely in terms of flexibility, responsiveness and re-configurability. In fact, these approaches are built upon

centralized and hierarchical control structures that present good production optimization, but a weak response to change due to the rigidity and centralization of their control structures. Such centralized hierarchical organization normally leads to situations where the whole system is shutting down by single failures at one point of the system hierarchy [3].

The current trend is to build modular, intelligent and distributed control systems, which exhibit innovative features, like the agile response to the occurrence of disturbances and the dynamic re-configuration on the fly, i.e. without stopping, re-programming or restarting the process. The multi-agent systems paradigm and technology [4-5] is particularly suited to implement this new kind of control systems to operate at the factory level. In fact, the distributed nature of the multi-agent systems approaches suggests the definition of distributed control functionalities based on autonomous agents that account for the realization of efficient, flexible and robust overall plant control, exhibiting modularity adaptation and re-configurability. Several applications of multi-agent systems at factory level were reported in the literature (e.g., see [6] and [7] and the references therein).

The EU FP 7 GRACE (inteGration of pRocess and quAlity Control using multi-agEnt technology) project (http://www.grace-project.org) is studying and developing a modular, flexible, adaptive and reconfigurable architecture using the multi-agent systems principles to operate at factory-level, handling the process control and the quality control in an integrated way, allowing the improvement of the production system performance [8]. The multi-agent architecture is designed for a factory operating with an on-demand production, producing washing machines, where production processes are subject to planned changes of set-points and to unexpected changes of process variables (e.g., disturbances and fluctuations of process/product parameters) [8]. The objective of the paper is to introduce the main principles of the designed multi-agent system to support the integration of quality and process control in washing machines production lines. The proposed architecture uses as source of inspiration the principles established in previous manufacturing multi-agent systems architectures, namely PROSA [9], ADACOR [10] and PABADIS-PROMISE [11], adapted according to the particularities of the on-demand production process.

The paper is organized as follows. First, Section 2 specifies the individual components of the multi-agent system architecture, namely the architecture for a generic agent, the identified types of agents and the individual agent's behaviour. Section 3 describes how the multi-agent system emerges from the interaction of individual agents and Section 4 analyses the mechanisms to integrate process and quality control processes. Finally, Section 5 rounds up the paper with the conclusions.

2 Individual Components of the Multi-Agent System

The proposed multi-agent collaborative architecture integrating the production and quality control processes, involve a society of distributed, autonomous and cooperative agents representing the manufacturing components. As suggested by [12], the idea is to preserve the current low-level control that already use state of art industrial control based on PLCs running IEC 61131-3 control programs, guaranteeing real-time responsiveness, and create an agent layer on top of it to provide intelligence and adaptation.

2.1   Generic Architecture of an Agent

In the proposed society of agents forming the multi-agent manufacturing control system, each individual agent has a partial view of the system and behaves according to a small number of simple rules that constitutes its behavioural repertoire, being defined by the tuple $\{O_i, P_i, B_i, I_i, S_i, A_i, K_i\}$, where:

- O_i is the agent's objective, described by an mathematical expression to be maximized or minimized.
- P_i is a plan, i.e. a sequence of actions leading to the realization of the goal.
- B_i is a set of behaviours to execute the agent's specific functions.
- I_i is a set of interfaces to connect the agent to its environment.
- S_i is the state of the agent, including the resources owned and its configuration.
- A_i is a set of attributes describing the agent's skills.
- K_i is the knowledge of the agent, consisting of a set of statements or facts.

The production, assembly and quality control stations of the home appliance production line will be treated as intelligent agents, but also the products being produced or other logical entities without physical representation. In spite of the particularities of each agent representing different (physical or logical) objects, a conceptual structure for a generic agent of the multi-agent system is established and comprises three main components: inter-agent communication, internal behaviour and interfaces, as illustrated in Fig. 1.
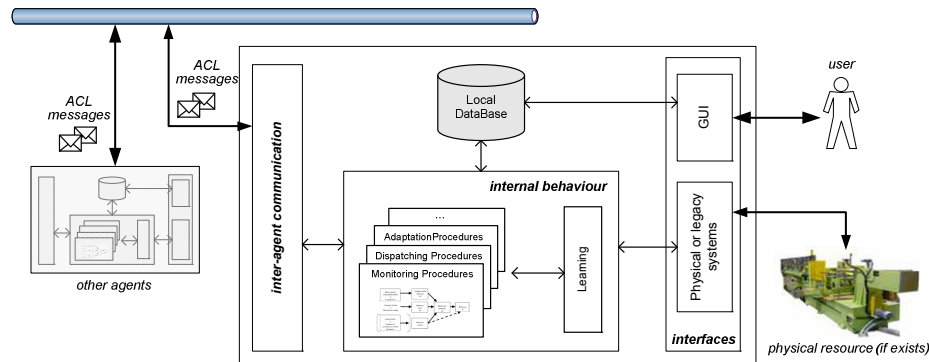

Fig. 1. Conceptual Model for a Generic Agent

The *inter-agent communication* component is responsible for the interaction with the other agents, making transparent the data exchanged to support the cooperation among the agents. The *internal behaviour* component is responsible for the several control functions that regulate the agent's behaviour, namely the monitoring, control, adaptation and learning. The *interfaces* component is responsible to support the interaction of the agent with the physical equipment (e.g., robots and machines) and/or legacy systems (e.g., MES - Manufacturing Execution Systems), if they exist. This component may also comprise a graphical user interface (GUI) to support a

direct interaction with the user. A local data base is used to store the agent's information and knowledge, e.g., the historical production information.

In terms of functionalities, the designed agents exhibit some important capabilities that guarantee the achievement of:

- *Modularity*, i.e. plugging in or out autonomous components/agents.
- *Adaptation*, i.e. applying local self-adaptive and self-organization concepts to adapt the system behaviour according to the environmental changes.
- *Reconfiguration*, i.e. adding, removing or changing a component on the fly, i.e. without the need to stop, re-program and re-initialise the other components.
- *Responsiveness*, i.e. a better response to changes, such as failures, by using distributed control structures.

The agents may also possess *learning capabilities* to support the dynamic improvement of the agent behaviour in the future based on the knowledge gathered during its life-cycle.

2.2 Identification of the Individual Agents

The designed multi-agent system architecture distributes the manufacturing control functions by several agents. In such distributed environment, several types of agents were identified according to the process to control and to their specialization [8]: product type agents (PTA), product agents (PA), resource agents (RA), quality control agents (QCA) and independent meta agents (IMA).

The *product type agents* represent the catalogue of products that can be produced by the factory plant and contains the process and product knowledge required to produce the product, namely the product structure and the process plan. The *product agents* manage the production of product instances in the factory plant. They possess a process plan to produce the product and interact with the RA agents for the production control and with the QCA agents for the quality control.

The *resource agents* are associated to the physical equipments of the production line, such as robots and machines, and manage the execution of their production/transportation/assembly operations in the production line. The *quality control agents* are associated to the quality control stations and manage the execution of quality control operations in the production line. The results provided by the QCA agents allows closing the control loops, making available reliable information to be used by other agents for adaptation of factory processes.

The *independent meta agents* will introduce a kind of hierarchy in the decentralized system, allowing the implementation of global supervisory control and optimized planning and decision-making mechanisms, e.g. defining and adapting global policies for the system.

2.3 Individual Agent Behaviours

The structure and behaviour of each agent is modelled by using the Petri nets formalism [13-14] to understand and synthesise the system specifications and to validate the correctness of those models. Petri net is a formal modelling tool, based on

a powerful mathematical foundation, adequate to model and analyse the behaviour of complex event-driven systems characterised as being concurrent, asynchronous, stochastic and distributed, such as flexible manufacturing systems are.

As an example, the Petri nets model representing the behaviour of the product agent is illustrated in the Fig. 2, containing several timed transitions (i.e. blue boxes in the model) representing complex functions that can be exploded into a more detailed sub- Petri nets model. This allows a top-down methodology to model the logic control structure of the agent's behaviour, refining step by step some timed transitions to include enough system operation details for purpose of the hardware implementation.
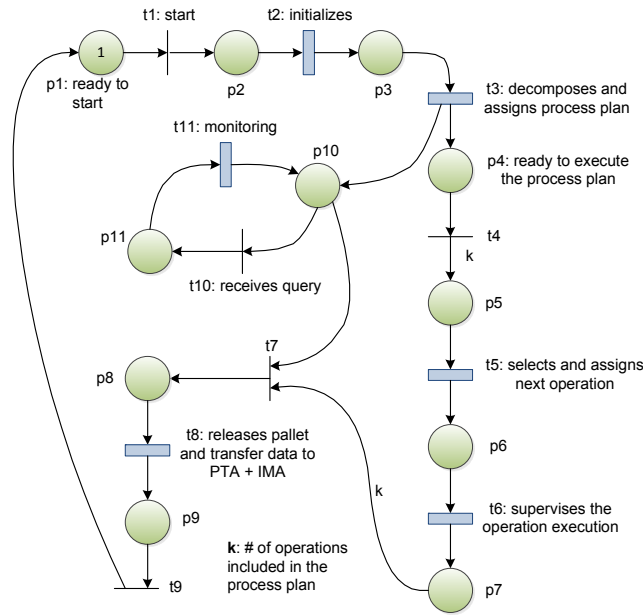

Fig. 2. Behaviour Model for the Product Agent

The PA agent is launched to manage the production of a product instance, being provided with a proper process plan and the *moby* identifier of the associated pallet. After the initialization, two sub-behaviours are launched, namely the monitoring and the process execution, being handled asynchronously in parallel, so that the execution of one process doesn't block the execution of the other process. In the process execution, the PA agent interacts with the RA and QCA agents that are able to execute the *k* individual operations belonging to the process plan. At the end of one operation execution, the next operation to be executed is selected by the function represented by the timed transition t_5, followed by the execution of the "supervises the operation execution" function represented by the timed transition t_6. This transition can be expanded into a refined Petri nets model containing the details about the functions to supervise the execution of the operation. After the execution of all *k* operations, the PA agent executes a set of actions, represented by the transition t_8, related to release the pallet and to transfer the relevant information of the product

execution to the PTA and IMA agents for posterior analysis to perform adaptation in the future process plans.

During the execution of the process plan, the monitoring requests about the state of the product execution, requested by other agents, are handled in parallel and not affecting the process execution.

3 Emergent Behaviour of the Multi-Agent System

The overall behaviour of a multi-agent system emerges from the interactions among individual agents, each one contributing with its local behaviour, recalling the concept of emergent behaviour. This phenomenon can be found in nature where complex systems are built upon entities that exhibit simple behaviours and have reduced cognitive abilities, with a small number of rules generating systems of surprising complexity [15]. As example, an ant or a bee present very simple behaviour but their colonies exhibit a smart and complex behaviour.

The production process will be supervised and controlled through the integrated and coordinated operation of a network of collaborative individual agents, each with its own objectives and behaviours, possessing its own perceptive and cognitive capabilities. In fact, in spite of individual agents being able to perform individually their control functions using proper embedded algorithms, they have a partial knowledge of the problem and need to interact each other when they haven't enough knowledge or skills to perform these functions alone. Fig. 3 represents the multi-agent system architecture emerging from the interaction among the individual agents.
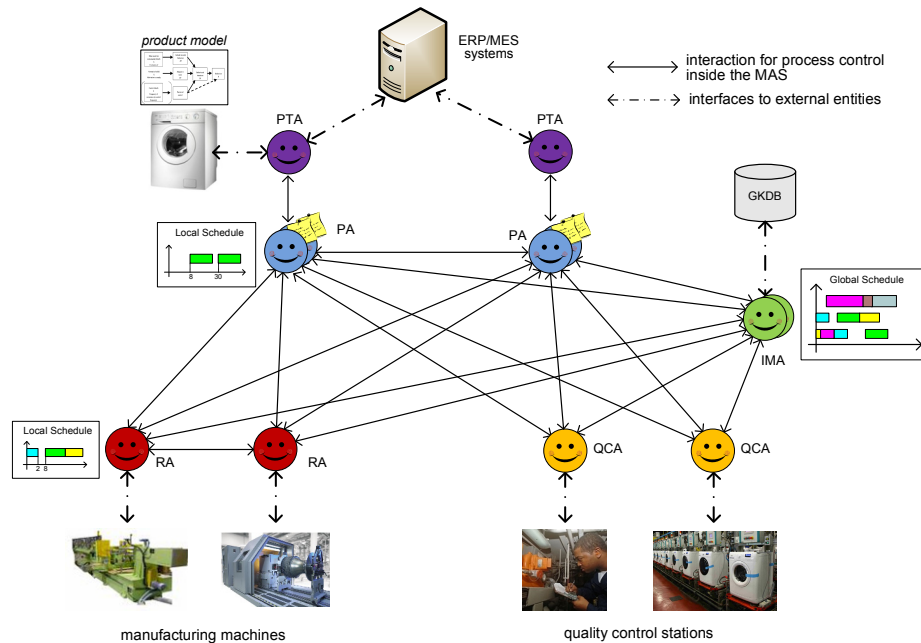

Fig. 3. Multi-agent System Architecture for the Production Line

The PTA agents receive orders from the ERP/MES system and launch PA agents to execute the production requests, exchanging product and process planning information. The PA agents interact with the RA and QCA agents during the execution of the process plan, re-routing the pallets according to the current production line conditions and querying about the progress of the plan execution. The PA, RA and QCA agents interact with the IMA agents to provide feedback information about the production execution and to receive optimized guidelines to improve their execution. The presence of IMA agents is optional and aims to provide global decision-making strategies based on data analysis methods which exploit information collected from individual agents. The RA agents interact between themselves during the physical synchronisation of production activities, such as for the transportation of the pallets between workstations.

Interaction patterns are required to model the cooperation among agents and to co-ordinate their actions to produce a product, enhancing the integration and adaptation of the production and quality control processes. As an example, Fig. 4 illustrates the interaction protocol to be followed by the agents during the execution of an operation.
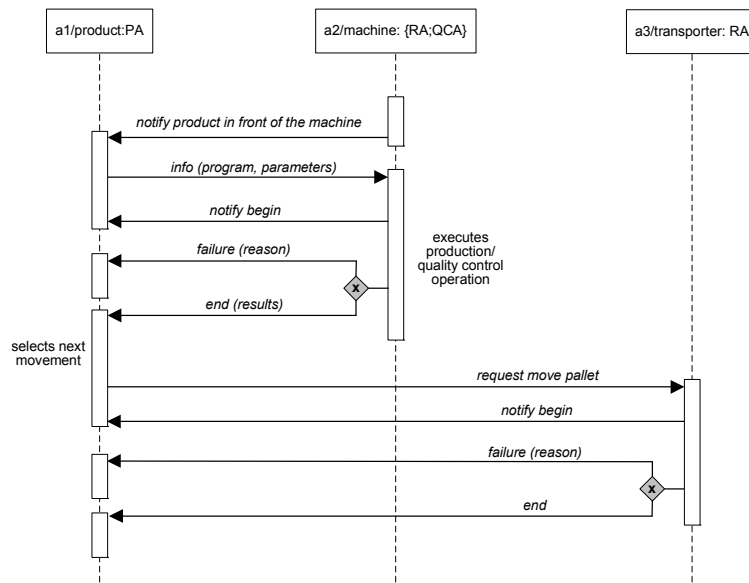

Fig. 4. Interaction Diagram for Operation Execution Process

Initially, the RA agent detects that a pallet has arrived to its machine station by reading the *moby* identifier associated to the pallet, and notifies the PA agent associated to the pallet. The PA agent determines the program and the parameters to be passed to the RA agent for the operation execution, based on the process plan and the information about the execution of previous operations. After finishing the execution of the operation, the RA agent notifies the PA agent about the results of the operation execution: *end* if it is concluded with success or *failure* if any problem had occurred during the execution. At this moment, the PA agent determines the next

operation and requests a movement of the pallet to the next station to a transport resource agent, which will deliver the pallet in the target station.

The design of other interaction patterns requires a more complex behaviour, namely the re-routing of pallets to face the occurrence of unexpected scenarios. For this purpose, the application of concepts inherited from the theory of complexity, emergence behaviour and self-organization will be considered.

4   Mechanisms for Integrating Process and Quality Control

The proposed multi-agent system approach wants to make a step forward by integrating production and quality control, implementing procedures to adapt the production process. For this purpose, self-adaptation and self-optimization mechanisms will consider the information collected locally by the agents in charge for the production and quality control, namely PAs, RAs and QCAs agents, and globally by the IMA agents through a multiple information fusion.

This section describes one example of this type of mechanisms, namely the interaction pattern to adjust the processing parameters of the machines and quality control stations based on the information of provided by the quality control tests. Fig. 5 illustrates the sequence of interactions among the distributed agents to achieve this behaviour based on the contribution of individual agents.
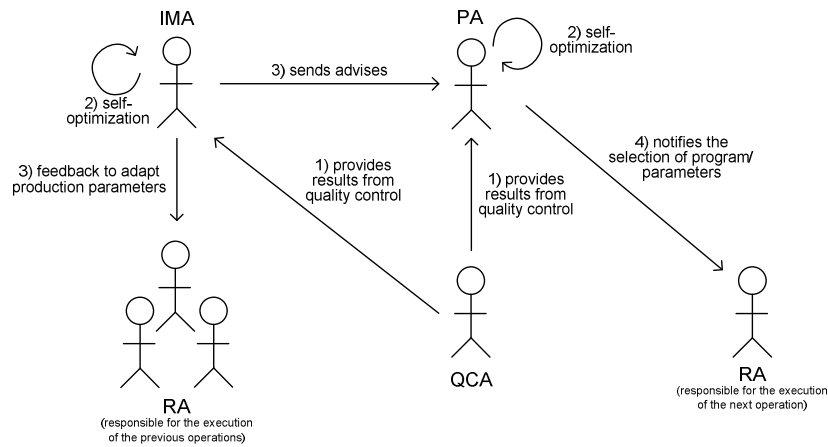

Fig. 5. Adjusting the Processing Program/Parameters of the Manufacturing Machines

Briefly, the QCA agent after performing the inspection tests over a product instance provides the results to the PA and IMA agents. The IMA agent can use this information to analyze possible deviations by using proper adaptation/optimization mechanisms, providing feedback to the RA agents responsible for the execution of the previous operations (i.e. responsible for the detected quality problem). These agents may use this information to improve the execution of future operations, correcting the detected problems/deviations, e.g., by calibrating the program parameters. The IMA

agent also provides feedback to the PA agent, namely advises about new strategies and policies for the production execution, aiming to increase the overall quality and performance of the productive process.

The PA agent, also using proper embedded adaptation mechanisms, uses the feedback information from the quality control operations to select the machining program, to adjust the parameters of the program or even to select the components to be used during the execution of the next operation. For this purpose, the adaptation mechanism embedded in the PA agent reasons by using the historical data and a proper adaptation algorithm. The achieved results are used by the PA agent to notify the RA agent associated to the machine responsible to perform the next operation about the adequate program/parameters/components to be used during the execution of the operation.

A similar procedure is used to adjust the configuration of the quality control tests to be executed by the quality control stations, by selecting the proper algorithms and/or testing parameters. In this case, the PA agent uses the feedback from the previous quality control tests and the advises provided by the IMA agent to decide about the best algorithms and/or parameters to be used in the next quality control operation. A practical example of this mechanism is the customization of the set of functional tests to be performed at the end of the production line by the produced product according to the results of the previous quality control tests performed during along the product production.

The achievement of adaptation mechanisms by integrating production and quality control encompasses a strong effort in designing proper optimization/adaptation mechanisms to be embedded in individual agents and in designing cooperation mechanisms to support the combination of the local and global knowledge/adaptation behaviours, which is a real challenge for the near future in the complete specification of the multi-agent system for the home appliance production system.

5   Conclusions and Future Work

The paper describes a multi-agent system approach for on-demand production of home appliances that integrates production and quality control processes, aiming to improve the production efficiency, reduce the production-line down-time and reduce the non-conformities by improved quality control of products. For this purpose, a multi-agent system was introduced by describing the two architectural levels: the micro and macro levels. The micro level was described by presenting the internal architecture of a generic individual agent, the functions of the identified classes of agents and the behaviour of the individual agents. The macro level is described by discussing how the multi-agent control system emerges from the interaction among individual agents, being a special attention devoted to the mechanisms to integrate production and quality control processes, pursuing the feedback control loops.

At this moment, a prototype of the agents' skeleton, namely for the PTA, PA, RA and QCA agents, was implemented by using the JADE (Java Agent DEvelopment Framework) platform [16]. The testing of this prototype is allowing to verify the correctness of the designed multi-agent system for the production of appliances (being possible to observe the interaction patterns during the production life-cycle of a product) and to support the refinement of the multi-agent system specification.

Future work is related to the further specification of the multi-agent architecture concerning the interfaces, namely the interaction among agents by using proper ontologies and the interaction with physical equipments and legacy systems, and the development of adaptation/optimization behaviours to be embedded (plugged-in) in the system agents.

References

1. MANUFUTURE: Strategic Research Agenda Manufuture (2006)
2. ElMaraghy, H.: Flexible and Reconfigurable Manufacturing Systems Paradigms. Int'l. Journal of Flexible Manufacturing Systems 17, 261–271 (2006)
3. Colombo, Schoop, R., Neubert, R.: An Agent-based Intelligent Control Platform for Industrial Holonic Manufacturing Systems. IEEE Transactions on Industrial Electronics 53(1), 322–337 (2006)
4. Wooldridge, M.: An Introduction to Multi-Agent Systems. John Wiley & Sons, Chichester (2002)
5. Ferber, J.: Multi-Agent Systems, An Introduction to Distributed Artificial Intelligence. Addison-Wesley, Reading (1999)
6. Leitão, P.: Agent-based Distributed Manufacturing Control: A State-of-the-art Survey. Engineering Applications of Artificial Intelligence 22(7), 979–991 (2009)
7. Monostori, L., Váncza, J., Kumara, S.: Agent-Based Systems for Manufacturing. Annals of the CIRP 55/2, 697–720 (2006)
8. Castellini, P., Cristalli, C., Foehr, M., Leitão, P., Paone, N., Schjolberg, I., Tjønnås, J., Turrin, C., Wagner, T.: Towards the Integration of Process and Quality Control using Multi-agent Technology. In: Submitted to the 37th Annual Conference of the IEEE Industrial Electronics Society, Melbourne, Australia, November 7-10 (2011)
9. Brussel, H.V., Wyns, J., Valckenaers, P., Bongaerts, L.: Reference Architecture for Holonic Manufacturing Systems: PROSA. Computers in Industry 37(3), 255–274 (1998)
10. Leitão, P., Restivo, F.: ADACOR: An Agile and Adaptive Holonic Architecture for Manufacturing Control. Computers in Industry 57(2), 121–130 (2006)
11. PABADIS'PROMISE Consortium: Structure and Behaviour of a PABADIS'PROMISE System. White Paper (2008)
12. Vrba, P., Tichy, P., Marik, V., Hall, K., Staron, R., Maturana, F.: Rockwell Automation's Holonic and Multi-agent Control Systems Compendium. IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews 41(1), 14–30 (2011)
13. Murata, T.: Petri Nets: Properties, Analysis and Applications. IEEE 77(4), 541–580 (1989)
14. Zurawski, R., Zhou, M.: Petri Nets and Industrial Applications: A Tutorial. IEEE Transactions on Industrial Electronics 41(6), 567–583 (1994)
15. Holland, J.: Emergence: from Chaos to Order. Oxford University Press, Oxford (1998)
16. Bellifemine, F., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE. Wiley, Chichester (2007)