Chapter 21
Review of Industrial Applications of Multi-agent Technologies

Pavel Vrba

Department of Cybernetics, Faculty of Electrical Engineering,
Czech Technical University in Prague, Czech Republic
pavel.vrba@fel.cvut.cz

Abstract. The intention of the manuscript is to give the overview of alternative control approaches that have been applied in industrial automation domain for more than two decades. Apart from more traditional centralized and hierarchical approaches the discussed ones are built on distributed, autonomous and intelligent entities that provide and consume services in networked environments. The key drivers have been holonic concept, multi-agent systems and recently service-oriented architectures. The talk discusses the major benefits as well as prevailing roadblocks hindering the widespread exploitation and deployment in real factories. It reviews the principle methodologies, architectures, tools and either pilot trials or commercially successful applications of industrial agent systems with major focus on achievements of the Rockwell Automation company.

Keywords: Manufacturing, Industrial Control, Multi-agent Systems, Holonic Systems, Ontology, Service-Oriented Architectures.

1 Introduction

For more than two decades many public research institutions as well as private companies try to enforce new technologies for making the automation solutions more open, flexible, and robust. In general, the key characteristics of these novel approaches are intelligence, distribution, and collaboration. The control system is no more perceived as a monolithic application with centric decision making, but rather as a network of self-contained intelligent components that pursue the globally defined goal by means of communication and cooperation.

The concepts of distributed control applications were initially developed within the framework of Holonic Manufacturing Systems (HMS) consortium, part of the Intelligent Manufacturing Systems (IMS) program. The term *holon*, devised by Arthur Koestler for description of a dual nature of units in living organisms, being simultaneously parts of larger wholes and wholes containing sub-parts, was reused for manufacturing. There it represents the basic unit of a manufacturing system having the same properties as for instance cells in the animal body – the substantial level of autonomy, yet being the subject of control from higher levels, and the ability to coordinate behaviour with the others. The prevalent technology for implementing the

T. Borangiu et al. (Eds.): *Service Orientation in Holonic and Multi Agent*, SCI 472, pp. 327–338.
DOI: 10.1007/978-3-642-35852-4_21

holonic principles have been for a long time the multi-agent systems (MAS) originating from the distributed artificial intelligence. Over the past several years the concepts of Service-Oriented Architectures (SOA) started to be exploited for the purposes of designing the distributed control applications. Both MAS and SOA are based on same principles – existence of self-contained loosely coupled entities that communicate in an orchestrated manner to achieve the high-level objectives.

The aim of this manuscript is to give the comprehensive overview of the application of *holonic*, *multi-agent*, and *service-oriented principles* in the industrial automation domain. Key attention is given to achievements of the Rockwell Automation, Inc., which has been acknowledged as pioneering company investing a large effort in researching into the alternative control solutions based on holonic and multi-agent systems. From first trials and simple prototypes created spontaneously and ad hoc, the gradual evolution resulted in a development of a comprehensive portfolio of methodologies, practices and tools that cover the whole life cycle of design, implementation, validation and deployment of *agent-based control systems*. The key point distinguishing Rockwell Automation's solutions from others is the consistent effort in integrating the alternative, agent-based solutions with the legacy automation control architectures. The main reasons are the conservativeness of the automation world and strict requirements for performance, robustness, and safety of the control solutions.

2 Methodologies and Architectures

2.1 Holonic Architectures

There were several holonic manufacturing methodologies proposed with the aim of formalizing types of holons, their behaviours and interaction scenarios. It is for instance well known PROSA – Product, Resource, Order, Staff Architecture [3], ADACOR – Adaptive Component Based Architecture [12], or HCBA – Holonic Component based Architecture [9].

Rockwell Automation has developed its own holonic architecture. The core concept of the architecture is the *workstation agent* (WA), which represents an aggregation of machines, equipment, and tools into a single entity, which is providing its services as manufacturing operations and/or material resources to other agents. Internally, it coordinates the execution of a complex operation by instructing the machine/equipment agents to perform their basic operations. The *product agent* (PA) establishes the concept of an intelligent product [19] that is able to manage its state and proactively coordinate its progress through the production process. Basically, the product agent, representing a single product instance, executes a production plan consisting of a series of operations that have to be performed to make the final product. The PA dynamically schedules execution of operations by negotiating with WAs about allocation of their time and resources. As shown in Fig. 1, there is the *order agent* (OA) representing the customer order for certain products. For each product in the order the OA creates corresponding PA, which in turn receives a tailored production plan. Such plan contains ordered list of operations for which a match with the operations provided by WAs is found during scheduling. Within the PA-WA

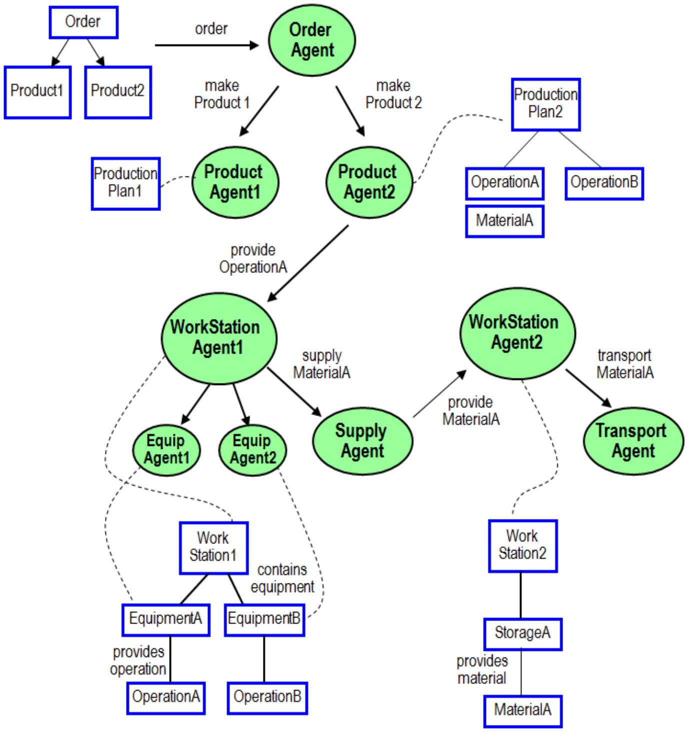

Fig. 1. Rockwell Automation's holonic architecture for agent-based production control systems

negotiation about operations scheduling the WA utilizes the help of *supply agent* (SA), whose task is to take care of the supply of all required material, including the product itself, to the workstation. The SA agent interacts with WAs that provide material and schedules its transportation to the workstation by help of transport agents.

Along with standardization aimed at holonic systems there was an intense effort to provide standards for MAS domain as well. The FIPA (Foundation for Intelligent Physical Agents) organization [11] produced sets of standards covering agent management, agent communication, and agent message transport. Variety of agent platforms providing the developers with the support for programming and running the agent applications were created: JADE [11], FIPA-OS, AGLOBE, MAdkit, or JACK.

2.2 Agents for Real Time Control

In the effort of applying agents at the factory floor level it is necessary to consider the requirement for both the ability of interacting with the physical equipment as well responsiveness under real-time constraints. This lead to a design of an aggregated architecture referred to as simply *holon* [24], *physical agent* [7], or *holonic agent*

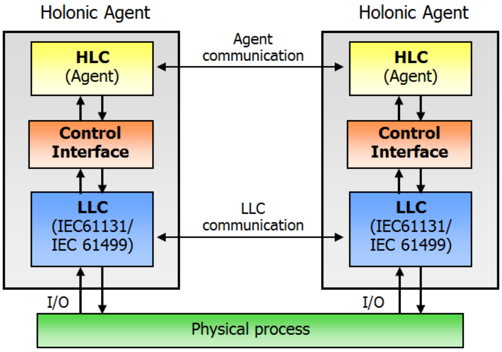

Fig. 2. Holonic agent architecture designed for application of agents in low-level real-time control tasks

[26]. As depicted in Fig. 2 it designed as a composition of high-level control part (HLC), low-level control subsystem (LLC), and the control interface for interactions between the HLC and LLC.

LLC module contains real-time control routines associated with the controlled physical process. To achieve seamless integration with legacy control architectures LLC can be implemented as a common control program running in PLC (Programmable Logic Controller) or other embedded controller. Rockwell Automation prefers to use IEC 61131 standard for programming PLCs to implement the LLC module. In other solutions, the IEC 61499 standard is used instead [8]. The HLC module contains the intelligent agent, which is responsible for autonomous decision-making and cooperation with other agents. Because of complexity of agent behavior the preferred programming languages for its implementation are C++ or Java.

The control interface serves as a communication link between HLC and LLC. Through this interface the LLC reports diagnostics information and other important events to the agent, like for instance the completion status of the product, detected failure, etc. On the other hand, the agent's decisions concerning the physical control are passed through the control interface to LLC. There are various ways of implementing the control interface – some used COM/DCOM or OPC technologies, in other cases sharing data on a black board was used, or PLC manufacturer's proprietary interfaces were applied [26]. In case of IEC 61499 used for LLC, its service interface function blocks are utilized [14]. The Rockwell Automation's approach is to share the data directly in the PLC memory in so called *tags*. The control interface is in form of C++ and Java API that allows the agents to read and write the tag values either in the real or emulated PLC and thus to directly interact with IEC 61131 control routines that utilize the same tags [26].

In order to leverage the object-oriented (OO) principles for designing agents for real-time control we have invented the concept of *holonic agent template* representing the class of the holonic agent according to the architecture displayed in Fig. 2. It obviously contains a class of the agent (HLC), which is easy to define as the object oriented languages, either C++ or Java, are used. The issue is with the definition of the class for the LLC module. It was already noted that we are using legacy IEC 61131 standards for LLC that is not object oriented. To overcome this issue we have developed the object-oriented extensions of the ladder logic (one language of IEC 61131) enabling the developer to define templates of LLC routines and instantiate them in particular control application in the same way as classes are instantiated in OO languages. The OO enhancements of IEC 61131 include technique called *indirect references*, which allows the developer to reference *this* instance's attributes. Other technique is *macro instructions* that give the possibility to iterate over a collection of components. Last is the *inheritance* that supports developer with the reuse of an existing LLC template to derive a specialized one.

2.3 Agent Communication and Organizations

Communication among agents is the crucial property of any multi-agent systems. The FIPA standards provide various communication protocols for querying, voting, negotiation, auctions, etc. [7]. The most popular one for agent negotiations is the *contract-net protocol* (CNP), in which the initiator agent asks a group of other agents for bids on provision of a particular service. When bids are received the initiator selects the best one(s) and delegates the contract to the chosen agent(s). If the queried agent cannot satisfy the request by itself it can decompose it and delegate execution to other agents by initiating another contract-net protocol.

We have extended CNP to provide more flexibility in terms of separating the planning, commitment and execution phases [11]. In the planning phase, similar to call for proposals of CNP, the requested agent does not allocate any resources, just gives a bid on their use. In the commit phase, it is confirmed that all the resources need to fulfil the task are available. It can happen that the agent that previously gave a bid can refuse the commitment because it could have already made commitment to the other agent. If this happens for all requested agents the planning phase has to be repeated. If the commit phase is successful the execution phase follows, in which the contracted agent executes the planned task. The benefit of separating these three phases is that there can be any long time periods between them. In certain cases it is useful to perform planning and commitment in advance and then after a specific period of time to start the execution of the requested service.

In order to find suitable agents that provide requested services the multi-agent system contains a specific agent called *Directory Facilitator* (DF) providing service registration and look-up functionality. A major issue is that usually only a single DF agent exists in the platform, like it is for instance in the popular JADE agent platform [2]. In this way the social knowledge is concentrated on a single place, which implies potential problems with insufficient fault tolerance of the system. To overcome this issue we have designed a structure of DF agents called *dynamic hierarchical teams* that has user-defined level of fault-tolerance and is fixed scalable. Groups of DF agents form teams, in which functionality of any of the team member can be in case of failure surrogated by another one. There are communication links between teams to enable the social knowledge propagation in order to increase the overall robustness of the system [23].

3 Tools

This section gives the overview of Rockwell Automation tools developed to support configuration, run-time, debugging and simulation of agent-based industrial control systems.

3.1 Agent Development Environment

The *Agent Development Environment* (ADE) is an integrated tool easing the development and deployment of industrial agent applications. It includes the Agent Template Editor where the user designs the templates of holonic agents (see Sect. 2.2), including both the HLC and LLC part. In case of HLC the user is provided with the generic skeleton of agent to which the user plugs-in component specific behaviours and services. For LLC module there is either textual or graphical Adobe Flash-based editor of ladder logic with the object-oriented enhancements mentioned earlier such as virtuality, inheritance, support for iterating over subcomponents, etc. In the Facility Editor the user creates specific application by defining instances of agent templates and customizing their parameters.

The Control System Editor is for defining the target hardware infrastructure for control system execution including PLCs, I/O cards, communication networks, etc. In the Assignment Editor the user assigns instances of holonic agents to particular execution units. Finally, the system automatically generates the program code of holonic agents and deploys them onto the hardware infrastructure [26].

The tools was formerly implemented in SmallTalk but later on reprogrammed as an Eclipse plug-in (Fig. 3). To our best knowledge such a tool is the only one of its kind.

3.2 ACS – Agent Platform for Industrial Controllers

Autonomous Cooperative System (ACS) is a platform for Rockwell Automation Logix controllers enabling the control system developers to run the holonic agents directly on PLCs. Using a modified firmware the C++ agents can run together with the IEC61131 control programs on the same processor of the standard Logix PLC. The platform provides standard features such as communication infrastructure and distributed fault tolerant agent services directory and look-up (see Sect. 2.3).

The platform is fully compliant with FIPA specifications. We successfully tested communication between the C++ agents running in ACS on the PLC and Jade agents running on PC.

3.3 JavaSniffer

Another useful tool is the JavaSniffer, which simplifies debugging and monitoring of the inter-agent communication. It captures and displays message flow using the kind of UML's sequential diagram (Fig. 4). It understands various communication protocols in order to cluster the corresponding messages and thus to show the workflow.

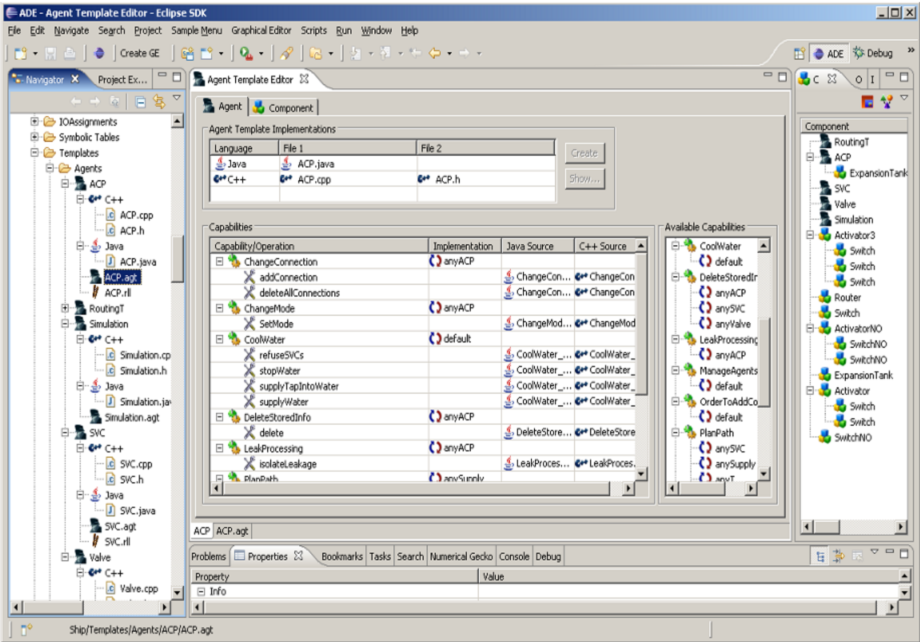

Fig. 3. Agent Development Environment for industrial control applications

It also offers advanced identification and visualization techniques for observing and analysing agent clusters in order to minimize communication among execution units [21].

3.4 Simulation Support

Testing and validating of the control system behaviour prior to its deployment in real factory have to be done in a simulated environment to avoid ramp-up delays and possible damage to the expensive equipment. For this purpose there is a need for advanced interface between the simulation tool such as Matlab or Arena and the control system running on the PLC. The issue is that the simulation can run slower or faster than the control system. So it is necessary to ensure the synchronization of both parties so that I/O values are exchanged at the same time. We have developed a Synchronizer tool, which ensures such synchronization by running the controller's programs and simulations alternately and stopping both after passing a single time period to exchange the I/O values [26].

4 Applications

There are multiple deployments of holonic and multi-agent systems in real industrial environments. The majority of them are proof-of-concepts and trials established in laboratorial conditions, and only few of them were executing in real factories.

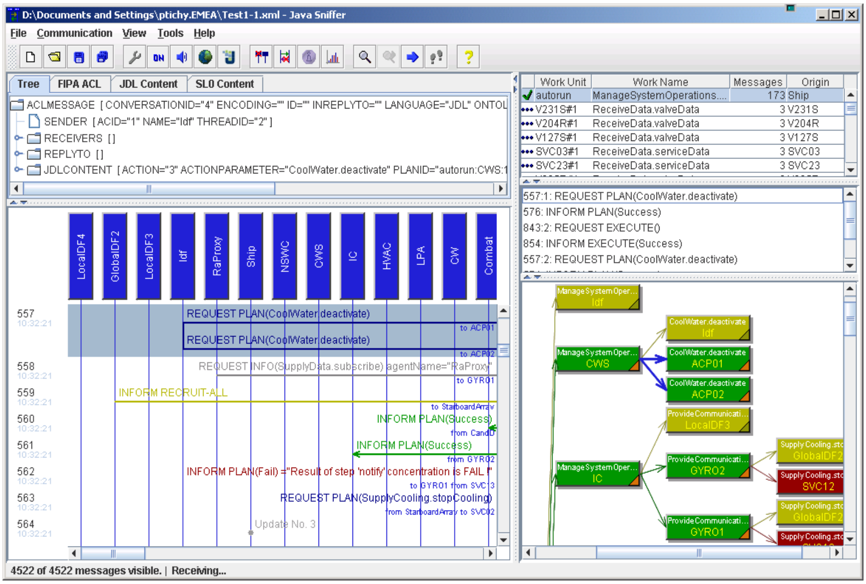

Fig. 4. JavaSniffer – tool for visualization of agent communication

Production 2000+ is the agent-based control for flexible cylinder heads production system. It was installed in DaimlerChrysler factory as a bypass of an existing line, providing greater flexibility and robustness against failures. It has been in day-to-day operation for five years with reported increase of 20% in productivity [4]. FABMAS is the PROSA-based holonic control system designed for production control of semiconductor wafers. It features dynamic routing of lots between groups of parallel machines represented by agents [18]. NovaFlex environment at UNINOVA institute in Portugal has been used for verification of agent-based control principles. Shop floor components including two assembly robots, automatic warehouse, and a transport system are controlled by agents organized according to CoBASA architecture [5]. Another example is the deployment of multi-agent manufacturing control system, following ADACOR holonic architecture, in a real laboratorial flexible manufacturing test bed at the Polytechnic Institute of Bragança, Portugal [13]. The Holonic Packing Cell developed by the University of Cambridge's DIAL laboratory established a large-scale industrial test bed for prototyping and testing distributed intelligent control systems [6]. The dynamic resource allocation is ensured by BDI (Belief-Desire-Intention) agents implemented in JACK platform. Several research projects aimed at holonic- and agent-based production control has been done by the ACIN institute at the Technical University in Vienna. The main focus is on adaptive material routing and control of assembly processes [17].

One of the earliest industrial agent projects of Rockwell Automation was the control of steel milling process. The requirement of a customer – BHP Billiton in Melbourne – was to dynamically assign jobs to available rolling stands and cooling boxes instead of using pre-defined subsets of equipment for particular recipes as before. The new agent-based control system achieved better performance in terms of increasing the equipment utilization [26].

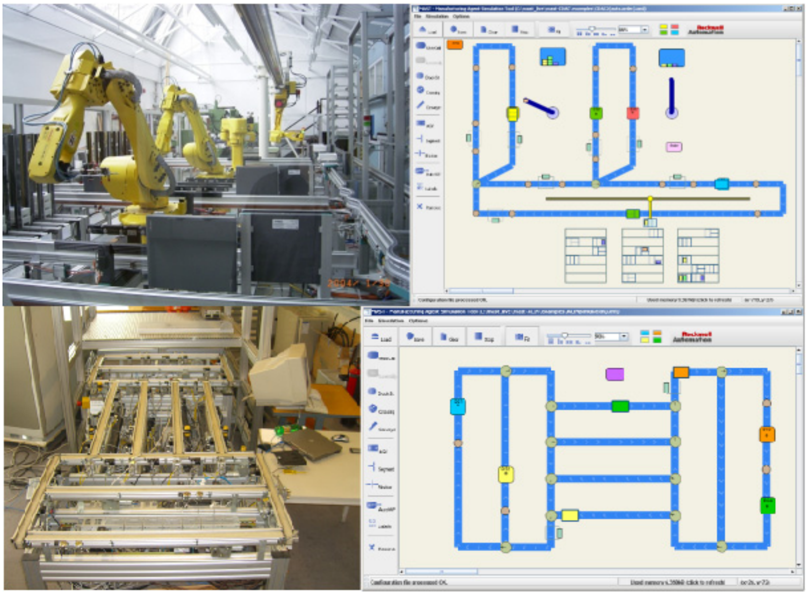

Fig. 5. Application of MAST for simulation of a packing cell at University of Cambridge's DIAL laboratory (upper two pictures); deployment of MAST for real-life control of transportation system at Vienna University of Technology's ACIN institute (lower two pictures)

4.1 Manufacturing Agent Simulation Tool

MAST – *Manufacturing Agent Simulation Tool* – has been developed by Rockwell Automation as a demonstrator of key benefits of agent-based approaches to manufacturing control. The primary aim was at the simulation of dynamic product routing but over the years it matured into generic-purpose manufacturing simulation and control tool featuring real-time connectivity to legacy PLCs, ontology-based dynamic scheduling, advanced diagnostics, etc. It models the transportation system as a set of independent agents that apply dynamic path searching algorithm to discover optimal routing paths through the conveyor network.

The first real application of MAST was established during collaboration with the DIAL laboratory at the University of Cambridge [6]. MAST was modified to provide full-fledged agent-based simulation of the packing line producing customized gift boxes (upper part of Fig. 5) [26]. A second deployment of MAST was aimed at simulating and also controlling the palette transfer system at the ACIN institute of Vienna University of Technology. MAST agents were provided with the access to real I/O values held in PLC in order to sense and actuate in the real environment (lower part of Fig. 5). Under real conditions we have verified the ability of agents to dynamically reconfigure the control system in terms of finding and applying alternative routing in case of a failure conveyor [26].

4.2 Chilled Water System

Another large agent-based application of Rockwell Automation was the robust control system for HVAC (Heating, Ventilation and Cooling) of the US Navy ship (Fig. 6). The key requirement was to ensure survivability of ships in case of equipment failures and damages caused by a hit of a missile. The on-board equipment such as chiller units, valves, and cooled services were represented by agents. The main feature was the ability to detect and isolate leakage and subsequently to reconfigure the control system in terms of finding the alternative routing of water in order to continue in cooling of the critical ship systems [26].

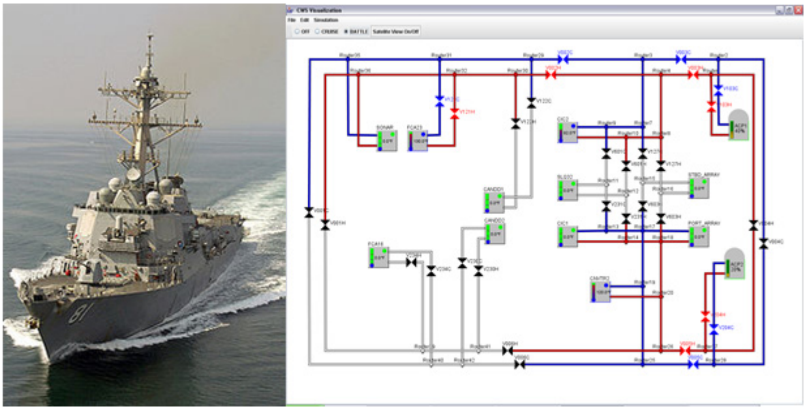

Fig. 6. Agent-based chilled water system for US Navy ship

5 Conclusions

Although the agent technology proved its benefits over classical centralized and hierarchical approaches we cannot still see massive deployments in real factories. The barriers for wider exploitation of agents in industry can be divided into two categories – the technology related and human factor related. The former one signifies the inability of the new technology to achieve the contemporary industrial requirements for real-time capabilities, robustness, safety, mature engineering tools, and standards. The latter one is related to a need for a significant paradigm change from controller-centric view to modularization and service orientation. The issue is that the control engineers, operators, and maintenance workers are not skilled to master the new technology.

The latest trends seem to be the application of two technologies coming from the IT world – the *Semantic web* and *Service-Oriented Architectures* (SOA). The agent-based control solutions tend to use ontologies for advanced representation, exchange and interpretation of knowledge [26]. Also SOA starts playing an important role. The SOCRADES project for instance has demonstrated the exploitation of SOA principles at both the business application level (MES and ERP systems) and the device level (smart I/Os, PLCs, etc.) [22]. The convergence of agent-based technologies and SOA is clearly visible and represents a future trend. SOA focuses mainly on developing standards for interfaces, protocols and workflows. Very little attention is given to the mechanisms that help the service to perform its task in an intelligent way. This is the field where SOA can benefit from leveraging some of the agent systems attributes. First experiments to combine SOA and agents have already been presented [16].

Acknowledgements. This research has been supported by Rockwell Automation laboratory for Distributed Intelligence and Control (RA-DIC) at the Czech Technical University in Prague, by the Grant Agency of the Czech Technical University in Prague, grant No. SGS12/188/OHK3/3T/13, and by the Ministry of Education of the Czech Republic within the Research Program no. MSM6840770038: Decision Making and Control for Manufacturing III.

References

1. Barata, J., Camarinha-Matos, L.: Coalitions of Manufacturing Components for Shop Floor Agility – the CoBASA Architecture. Int. Journal of Networking and Virtual Organisations 2(1), 50–77 (2003)
2. Bellifemine, F., Caire, G., Greenwood, D.: Developing multi-agent systems with JADE. Wiley, Chichester (2007)
3. Brussel, H.V., Wyns, J., Valckenaers, P., Bongaerts, L.: Reference Architecture for Holonic Manufacturing Systems: PROSA. Computers in Industry 37(3), 255–274 (1998)
4. Bussmann, S., Schild, K.: An Agent-based Approach to the Control of Flexible Production Systems. In: Proceedings of 8th IEEE International Conference on Emerging Technologies and Factory Automation, vol. 2, pp. 481–488 (2001)
5. Cândido, G., Barata, J.: A Multiagent Control System for Shop Floor Assembly. In: Mařík, V., Vyatkin, V., Colombo, A.W. (eds.) HoloMAS 2007. LNCS (LNAI), vol. 4659, pp. 293–302. Springer, Heidelberg (2007)
6. Fletcher, M., McFarlane, D.C., Thorne, A., Jarvis, D., Lucas, A.: Evaluating a Holonic Packing Cell. In: Mařík, V., McFarlane, D.C., Valckenaers, P. (eds.) HoloMAS 2003. LNCS (LNAI), vol. 2744, pp. 246–257. Springer, Heidelberg (2003)
7. Foundation for Intelligent Physical Agents (2002), `http://www.fipa.org`
8. Hegny, I., Hummer-Koppendorfer, O., Zoitl, A., Koppensteiner, G., Merdan, M.: Integrating Software Agents and IEC 61499 Real-time Control for Reconfigurable Distributed Manufacturing Systems. In: Proc. Int. Symposium on Industrial Embedded Systems, France, pp. 249–252 (2008)
9. Chirn, J.-L., McFarlane, D.: A Holonic Component-Based Approach to Reconfigurable Manufacturing Control Architecture. In: Proceedings of the International Workshop on Industrial Applications of Holonic and Multi-Agent Systems, pp. 219–223 (2000)
10. Chirn, J.-L., McFarlane, D.: Building Holonic Systems in Today's Factories: A Migration Strategy. Journal of Applied System Studies 2(1) (2001)
11. Kadera, P., Tichy, P.: Plan, Commit, Execute Protocol in Multi-agent Systems. In: Mařík, V., Strasser, T., Zoitl, A. (eds.) HoloMAS 2009. LNCS (LNAI), vol. 5696, pp. 155–164. Springer, Heidelberg (2009)
12. Leitão, P., Colombo, A.W., Restivo, F.: ADACOR: A Collaborative Production Automation and Control Architecture. IEEE Intelligent Systems 20(1), 58–66 (2005)

13. Leitão, P., Restivo, F.: Implementation of a Holonic Control System in a Flexible Manufacturing System. IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews 38(5), 699–709 (2008)
14. Lopez, O., Martinez Lastra, J.L.: A Real-time Interface for Agent-based Control. In: Int. Symposium on Industrial Embedded Systems, pp. 49–54 (2007)
15. Lüder, A., Peschke, J., Sauter, T., Deter, S., Diep, D.: Distributed intelligence for plant automation based on multi-agent systems: the PABADIS approach. Production Planning and Control 15(2), 201–212 (2004)
16. Mendes, J.M., Leitão, P., Restivo, F., Colombo, A.W.: Service-Oriented Agents for Collaborative Industrial Automation and Production Systems. In: Mařík, V., Strasser, T., Zoitl, A. (eds.) HoloMAS 2009. LNCS (LNAI), vol. 5696, pp. 13–24. Springer, Heidelberg (2009)
17. Merdan, M., Koppensteiner, G., Zoitl, A., Hegny, I.: Intelligent Agent-based Approach for Assembly Automation. In: Proc. IEEE Conf. Soft Computing in Industrial Applications, pp. 13–19 (2008)
18. Mönch, L., Stehli, M., Zimmermann, J.: FABMAS: An Agent-Based System for Production Control of Semiconductor Manufacturing Processes. In: Mařík, V., McFarlane, D.C., Valckenaers, P. (eds.) HoloMAS 2003. LNCS (LNAI), vol. 2744, pp. 258–267. Springer, Heidelberg (2003)
19. Sallez, Y., Berger, T., Tretensaux, D.: A Stigmergic Approach for Dynamic Routing of Active Products in FMS. Computers in Industry 30(3), 204–216 (2009)
20. Shen, W., Maturana, F.P., Norrie, D.: MetaMorph II: An Agent-based Architecture for Distributed Intelligent Design and Manufacturing. Journal of Intelligent Manufacturing 11(3), 237–251 (2000)
21. Staron, R.J., Tichý, P., Šindelář, R., Maturana, F.P.: Methods to Observe the Clustering of Agents Within a Multi-Agent System. In: Mařík, V., Vyatkin, V., Colombo, A.W. (eds.) HoloMAS 2007. LNCS (LNAI), vol. 4659, pp. 127–136. Springer, Heidelberg (2007)
22. Taisch, M., Colombo, A.W., Karnouskos, S., Cannata, A.: SOCRADES Roadmap: The Future of SOA-based Factory Automation (2010), http://www.socrades.eu/
23. Tichý, P.: Middle-Agents Organized in Fault Tolerant and Fixed Scalable Structure. Computing and Informatics 22, 597–622 (2003) ISSN 1335–9150
24. Van Leeuwen, E.H., Norrie, D.: Intelligent Manufacturing: Holons and Holarchies. Manufacturing Engineer 76(2), 86–88 (1997)
25. Vrba, P., Radakovič, M., Obitko, M., Mařík, V.: Semantic Technologies: Latest Advances in Agent-based Manufacturing Control Systems. International Journal of Production Research 49(5), 1483–1496 (2011)
26. Vrba, P., Tichý, P., Mařík, V., Hall, K., Staron, R., Maturana, F., Kadera, P.: Rockwell Automation Holonic and Multi-agent Control Systems Compendium. IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews 41(1) (2011)
27. Vyatkin, V.: IEC 61499 as Enabler of Distributed and Intelligent Automation: State-of-the-Art Review. IEEE Transactions on Industrial Informatics 7, 768–781 (2011)