

# Analyzing Standardization Needs for Applying Agent Technology in Industrial Environments

Paulo Leitão  
Polytechnic Institute of Bragança  
Department of Electrical Engineering  
Braganca, Portugal  
Email: pleitao@ipb.pt

Thomas Strasser  
AIT Austrian Institute of Technology  
Energy Department  
Vienna, Austria  
Email: thomas.i.strasser@ieee.org

**Abstract**—Multi-agent systems can play an important role in the realization of cyber-physical systems by providing intelligence, flexibility, robustness, self-adaptation, and self-organization to them. In spite of the promising perspective offered and concretized by the deployment of multi-agent systems in a few industrial systems and in several laboratory applications, its industrial adoption is far from the expected widely usage. Beside other issues, standardization is being identified as a critical aspect for this status and requires the need to be compliant with existing industrial practices but also a special effort to influence the specifications of existing standards and/or the introduction of new ones. The main objective of this paper is to analyze necessary standardization needs for applying and deploying agent-based technology in industrial environments, addressing industrial requirements imposed by different application fields.

## I. INTRODUCTION

A rise in the automation level in different industrial domains like manufacturing, power and energy systems, or logistics is observable during the last decades. This trend results typically in a higher system complexity but offering new and extended possibilities. Actuators, sensors, and control devices from different vendors have to work together with supervisory control and management systems in heterogeneous communication environments in a cyber-physical manner [1].

As a result, automation software used in the aforementioned areas has to deal more and more with a complex structure of distributed and autonomously acting devices. Proper approaches like agent-based control—derived from the field of distributed artificial intelligence—can play an important role in the realization of such Cyber-Physical Systems (CPS) by providing intelligence, flexibility, robustness, self-adaptation, and self-organization to these systems [2]. In spite of the promising perspective offered and concretized by the deployment of Multi-Agent Systems (MAS) in some industrial systems, and particularly in several laboratory applications, its industrial adoption is far from the expected widely usage [3]. Standardization is being identified as a critical aspect for this state and requires the need to be compliant with existing industrial practices but also a special effort to influence the specifications of existing standards and/or the introduction of new ones [4].

The objective of this work is to identify necessary standardization needs in applying and deploying agent-based technol-

ogy in industrial environments, addressing industrial-specific requirements imposed by different application fields like manufacturing, power and energy, logistics and building automation. For this purpose, the activities of the IEEE P2660.1 Working Group regarding the recommendation practices on industrial agents focusing the integration of software agents and low level automation functions will be detailed.

This paper is organized as follows. Section II introduces the concepts of software agents, multi-agent systems and industrial agents, whereas Section III analyzes existing standards in the field and identifies missing issues according to important industrial requirements. Section IV details two exemplary areas where standardization efforts should benefit the adoption of industrial agents and Section V describes the efforts being elaborated by the IEEE P2660.1 Working Group in developing recommendation practices on industrial agents focusing on the integration of software agents with low level automation functions. At last, Section VI rounds up the paper with the conclusions and future work.

## II. SOFTWARE AGENTS VS. INDUSTRIAL AGENTS

The paradigm of MAS has been developed to design large-scale distributed control systems in a more natural way. Originally derived from the field of distributed artificial intelligence, it has been adopted to control also industrial systems. MAS are constructed using autonomous and cooperative software entities (i.e., software agents) which are working together in a networked environment to solve a common (control) goal or task and adding modularity, flexibility and robustness to such kind of systems. Each software agent usually has its own skills and knowledge resulting in an autonomous proactive behavior. Agents are working together, interacting and sharing their knowledge and skills in order to achieve global objectives in a MAS environment. The resulting structure is of distributed nature (exemplary shown in Fig. 1) compared to the traditionally widely used centralized solutions [2], [5].

The main advantages of agent-based approaches can be summarized as follows [2], [5], [6]:

- **Modularity:** Possibility of adding new elements to the system under control in a plug and play manner without the need to stop, re-configure/re-parametrize, or reprogram and re-start it.

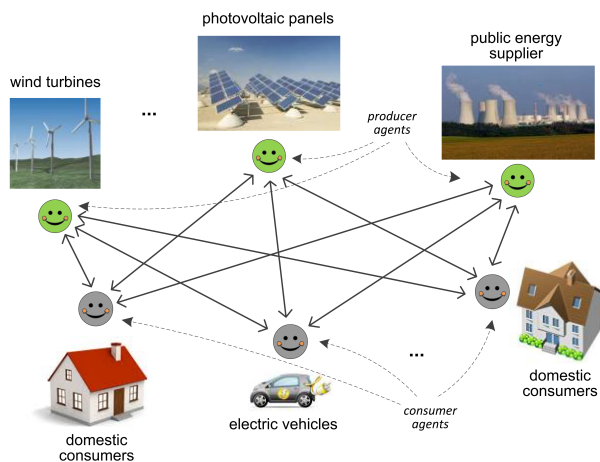


Fig. 1. Example for a MAS-based application: distributed control of an energy system [6].

- *Scalability*: Extension of the system under control can be carried out in a simple way without the need to stop it.
- *Reconfigurability*: Changes to the software structure/application can be performed on the fly (e.g., agents can stop, modify its behavior or strategy, and start again without affecting the other components of the system).
- *Robustness*: Losing one agent does not imply a system failure (e.g., if an agent associated with a customer fails, the system continues to run).

Agent technology play an important role for the implementation of the concept of CPS, which refers to the combination of mechatronics and Information and Communication Technologies (ICT) to control physical processes and systems. It provides proper means to achieve intelligent, adaptive and autonomic behavior of CPS components and applications. At this stage, and since CPS impose different requirements, it is important to define the concept of industrial agents, clarifying the difference to the traditional software agents.

Industrial agents inherit the software agent principles, such as intelligence, autonomy and cooperation, being applicable to industrial applications as well as their corresponding hardware components (cf. Fig. 1) and then facing industrial requirements [4]. This means that agents usually have associated a physical hardware counterpart, which increases the deployment complexity. Due to the aforementioned advantages, agent-based technology has been successfully applied to various industrial environments mainly in research projects and laboratory experiments. Promising industrial application fields are in the domain of smart production systems [3], [5], intelligent power and energy systems (i.e., Smart Grids) [6], [7], smart logistics [8], but also in the area of smart healthcare [9].

### III. EXISTING STANDARDS AND INDUSTRIAL REQUIREMENTS

This section aims to overview existing standards in the field and identify important industrial requirements imposed to the deployment of agent technology in industrial applications.

#### A. Existing Standards

As already stated above, standardization is seen as a critical factor for the implementation of agent technology in order to realize interoperable and scalable solutions. Up to now there are not many agent standards available but several supporting technologies which are briefly characterized below.

Useful specifications for the realization of industrial agents have been developed by the Foundation for Intelligent Physical Agents (FIPA) [10]. The two most important FIPA definitions are the Abstract Architecture and Agent Management as well as the Agent Communication Language (ACL). The first one provides entities which are required to build agent services and agent environments and the latter one specifies a communication language and the necessary corresponding interaction protocols. FIPA is currently the only standard for the development of MAS systems. Due to the high popularity of the FIPA-compliant Java Agent DEvelopment (JADE) framework, FIPA is usually adopted by the agent developers community.

With the development of Service-oriented Architectures (SOA) as well as web services and their adoption to industrial environments, a new technology becomes of interest to realize MAS [11]. Standardized approaches in this domain are the Devices Profile for Web Services (DPWS) and the Representational State Transfer (REST) [12]. DPWS addresses ubiquitous device integration using web services embedded in distributed devices whereas REST can be seen as an alternative integration approach emerged out of the world wide web initiative and focuses on Machine-to-Machine (M2M) communication. Another helpful integration approach is provided by the so-called Enterprise Service Bus (ESB) idea, a SOA-based framework, which can be used as a backbone for supporting the interoperability among distributed agent systems.

An important issue of industrial agent-based approaches is the communication between different devices [11]. In this area several standardized approaches like the Extensible Messaging and Presence Protocol (XMPP), the Message Queue Telemetry Transport (MQTT) or the Simple Object Access Protocol (SOAP) can be used to specify the communication between agents.

Since industrial agents interact with real hardware components, also (real-time) control and automation tasks have to be addressed. Several standards from the industrial automation domain exists which can be used for such activities. The most important ones are the well-known and widely used IEC 61131 for the realization of Programmable Logic Controllers (PLC) [13] as well as the IEC 61499 reference architecture for the realization of distributed control systems [14]. Moreover, IEC 62541, called OPC Unified Architecture (OPC UA) [15], is an interesting communication approach to harmonize the information exchange in industrial systems.

Summarizing, there exists a view amount of agent-based standards but a lot of supporting technologies which need to be harmonized and integrated. Standardization is still a clear necessity for the realization and implementation of agent-based approaches in industrial systems.

## B. Industrial Requirements and Missing Issues

Industrial applications impose different and strong requirements that may affect the adequacy of the existing standards for the adoption of the agent technology. Examples of these requirements are the integration with hardware devices, industrial standard compliance, reliability, fault-tolerance, scalability, quality assurance, resilience, manageability and maintainability [16], [17].

In terms of standardization, these industrial requirements, which may have different degrees of importance depending on the scenarios where industrial agents are applied, leads to some gaps in deploying agent technology, and especially in the FIPA specifications. Some examples are:

- Need for real-time interaction protocols for industrial and large-scale systems, ensuring scalability and latency.
- Need for a light-weight ACL protocol to support scalability in large-scale systems (usually the format protocol of ACL is too heavy mainly due to the headers; new protocols with a reduced size of the headers are required).
- Need for distributed yellow pages service and discovery mechanism to improve the system robustness.
- Need for special functionalities required at industrial level, namely event notification at low control level (e.g., sensors level) and service unsubscription.
- Need to combine agents with complementary paradigms/technologies to overcome some agents' limitations, e.g., with SOA to achieve interoperability and IEC 61131/IEC 61499 to fulfill real-time constraints.
- Need for the parallelization of multi-agent solutions to run on distributed computing resources, e.g., in high performance computing or cloud platforms, complemented with cyber-security and privacy aspects.

Additionally, other issues are missing in existing standards and constitute an opportunity for the development of efforts to standardize. Examples are the integration with hardware and legacy systems, and the process for debugging and validation of agent-based solutions. In the context of CPS, the integration of agents with physical hardware and software automation systems assumes a critical importance and can be identified as a current lack in industrial standardization. In fact, standardization efforts need to focus the way to interface in a easy, transparent, and reusable manner software agents with automation devices (PLCs, robots, machinery, etc.). Also important is the standard integration with legacy systems, e.g., databases, ERP (Enterprise Resource Planning), MES (Manufacturing Execution System), SCADA (Supervisory Control And Data Acquisition) and DCS (Distributed Control System). A special attention should also be devoted to the integration with humans since it may impose particular requirements, and is a crucial issue in the emergent "Industrie 4.0" strategic initiative [18].

Another pertinent standardization issue is related to the establishment of guidelines and procedures for the debugging, verification and validation of industrial agent-based solutions, ensuring that the system accomplish the application specifica-

tions. For this purpose, amongst others, the establishment of benchmark frameworks to test and compare different solutions are required, defining testing scenarios, performance indicators and measurement metrics.

## IV. STANDARDIZATION ISSUES FOR INDUSTRIAL AGENTS

After detecting missing gaps in applying agent technology in industrial environments, this section details two exemplary areas where standardization efforts provide benefits in adopting industrial agents, namely combining agents and services, and integrating agents with low-level automation functions.

### A. Combining Agents and Services

SOA can be combined with MAS to enhance functionality and to overcome some MAS limitations, namely in terms of interoperability and vertical integration of Information Technology (IT). This idea is already slightly considered in the specification of both approaches (e.g., see [19] for SOA and [20] for agents). In spite of being based on the same concept of providing a distributed approach to the system, MAS and SOA present some important differences, namely in terms of computational requirements, autonomy, and interoperability (see [21] for a deeply study). These differences highlight the complementary aspects of the two paradigms, suggesting the benefits of combining them.

Traditionally, the combination of MAS and SOA paradigms can be performed in different ways, as showed by [22]. The first option considers a gateway to translate the semantics from the agents world to the services world. This involves to translate entries on the Directory Facilitator (DF) to/from Universal Description, Discovery, and Integration (UDDI) registry entries, and to translate the agents skills to/from Web Service Description Language (WSDL) instances. Also, the message translation is performed between ACL and SOAP messages [23]. An example is the Web Services Integration Gateway (WSIG) plug-in provided by the JADE framework to offer an implementation of the gateway concept [24]. This plug-in, in form of gateway agent, was implemented by Whitestein Technologies and allows a transparent and bidirectional transformation between FIPA compliant services and Web services employing the WSDL/SOAP/UDDI stack (i.e., publishing agents capabilities as Web services used in a SOA environment).

The use of an ESB constitutes an alternative way to implement the integration of MAS and SOA, where software applications are MAS-based systems that are interacting through the use of the ESB, by exposing and consuming services. ESB provides a layer on top of an implementation of an enterprise messaging system [25], acting as backbone for supporting the interoperability among the connected software applications. Usually, ESBs provide a set of functionalities, namely process orchestration (typically via Web Services Business Process Execution Language (WS-BPEL)), protocol translation, versioning, lifecycle management, and security. An example of this approach is the solution developed by the European Union funded ARUM (Adaptive Production Management) project

where an ESB platform provides a common infrastructure for the integration of heterogeneous agent-based planning and scheduling tools and legacy systems using the service-orientation principles [26].

The challenge in this area is to identify and standardize the best practices that allow to design truly service-oriented MAS, according to the type of application and set of requirements, taking advantages of the real expected potential and benefits. An important note is that these service-oriented agents do not only share services as their main form of communication, but also complement their own goals with external provided services [22].

### B. Integrating Agents and Low-level Automation Functions

MAS in industrial automation systems often need to interact with real hardware (e.g., automation equipment like robots, machines, and conveyor belts; or power system components like generators and transformers). Depending on the agent tasks, this interaction ranges from simple gathering of data to direct control of the corresponding equipment. Especially, the latter approach requires real-time constrained execution of control tasks (see also Section III).

Agent-based systems in general are not designed to directly perform real-time constrained control of machinery, partly because of their collaborating negotiation behavior. A two-layered architecture with a higher-level fulfilling strategic, planning, and supervisory tasks and a reactive lower-level real-time layer handling real-time constrained control task, as illustrated in Fig. 2, seems to be a promising approach to overcome this limitation [5], [27], [28].

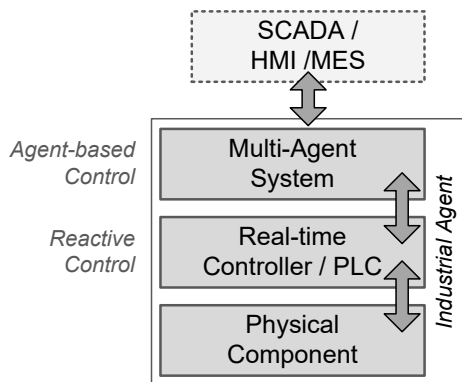


Fig. 2. Two-layer structure for industrial agents (adapted from [5], [27], [28]).

In this two layered architecture, both layers can focus on the tasks they are suited best. The MAS is in charge of providing intelligence and adaptation to the higher control level of the associated device/system, consisting of execution planning, cooperating, and negotiating with other sub-systems. The lower-level is responsible to provide real-time control operation for the automation device, which is typically executed in small industrial embedded control devices (usually compliant to IEC 61131 or IEC 61499). The core tasks of this real-time control layer are the reading of sensor values

from input signals, pre-processing of these values according to the execution of the control application(s), and generating appropriate values for actuators connected to the outputs of the control devices.

A key aspect in such two-layer architecture is the interface between both layers, and consequently between agents and the low-level control devices. In fact, agents need to be able to request services from the low-level automation functions (e.g., perform certain machining operation) and get feedback about the execution status of the services (e.g., remaining time, finished). Furthermore, agents need to be able to change parameters of the low-level control devices (e.g., adjust or parameterize the control programs to execution modes or products being produced). Finally, the agents require status feedback from the low control level, which should initiate and report the general status of the controlled automation device (e.g., depletion of supplies) and especially on critical conditions (e.g., stuck palette).

Having this in mind, a challenge in deploying industrial agents aiming to achieve interoperability is to define a standardized way of information exchange between the agent and the physical automation devices and controllers performing several automation functions. Moreover, also a standardized way of invoking these services and functions, as well as their implementation, might be necessary. Note that the MAS and the low-level control should be loosely coupled, such that the last one can also operate without an agent system.

## V. IEEE WORKING GROUP P2660.1

Recognizing the importance of standardization for the industrial adoption of agent technology, the IEEE Standards Association (IEEE-SA) has launched in October 2015 the corresponding IEEE P2660.1 Working Group aiming to develop recommendation practices on industrial agents focusing on the integration of software agents and low-level automation functions. This Working Group is sponsored by the IEEE Industrial Electronics Society (IES) and the IEEE Systems, Man, and Cybernetics Society (SMCS), as well as technically sponsored by the IEEE IES Technical Committee on Industrial Agents and the IEEE SMCS Technical Committee on Intelligent Industrial Systems. The proposed recommendation aiming to standardize the interface process to achieve re-usability and transparency, will support and help engineers to leverage the best practices of developing industrial agents for automation control problems and given application fields, in the emergent context of CPS.

The objective of this IEEE-SA Working Group is to provide the best practices rules, guidelines, and design patterns to address the integration of software agents with low-level real-time control systems, mainly IEC 61131-3 control programs running on PLCs, IEC 61499 distributed control applications, or embedded control applications. As described in [29], and illustrated in Fig. 3, the idea is to establish standardized interfaces based on SOA principles, where the physical automation device is abstracted in form of standard services. This generalized approach should consider the two different

possibilities to integrate cyber and physical components: *i*) embedding the agent within the physical control device (e.g., the agent is running directly on a PLC or an embedded controller and accessing directly the I/O signals to execute the control laws), or *ii*) connecting the agent with the existing control device in a coupled manner (e.g., the agent is running on a PC or tablet device and interacts with the control program running in a PLC/embedded controller). Note that hosting agents in the PLC to interact with the control applications can also be included in the coupled approach.

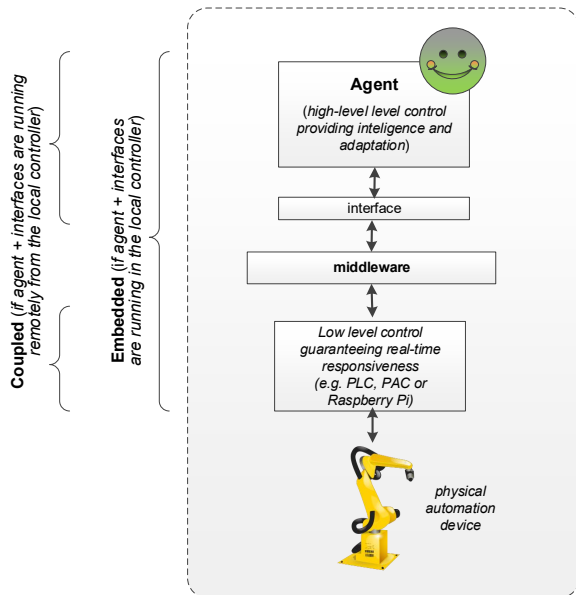


Fig. 3. Integration of agents with low level automation functions (adapted from [29]).

For this purpose, and after defining the scope and boundaries of the proposed standard, an execution plan comprising three phases was established: *i*) analysis of existing practices for the different situations related to the integration of agents with lower-level control functions, namely studying several application domains like factory automation, energy and power systems, and building automation, *ii*) generalization to derive recommended practices based on results from the analysis phase, defining the semantics of the interface and the technologies for the middleware, and *iii*) testing the identified and recommended practices. This will be realized by creating a structure of sub-groups (see Fig. 4).

The testing sub-group is transversal to all phases and aims to initially design test cases and validation/testing procedures that should be considered by other sub-groups. In the *testing phase*, this sub-group will be responsible to perform the testing and validation of the identified recommended practices.

During the *analysis phase*, three sub-groups are considered to collect and analyze the existing practices for the different situations related to the integration of agents with lower-level control functions, according to the type of application, i.e. one for factory automation, another for power and energy systems

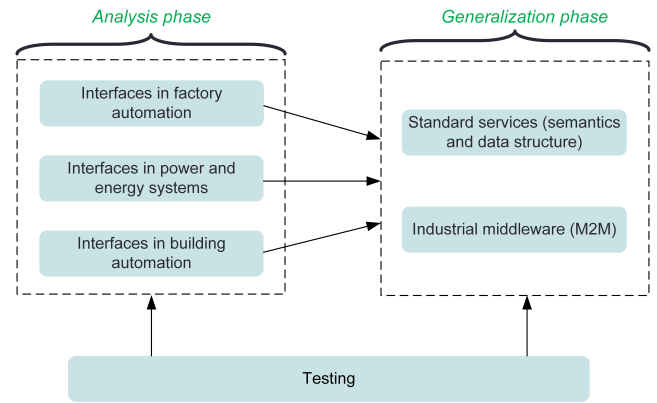


Fig. 4. Sub-groups structure of the IEEE P2660.1 Working Group.

and another one for building automation. The expected results from each sub-group are mainly to compile :

- The type of devices/controllers usually used (e.g., PLCs, robot controllers, Raspberry Pis, etc.).
- The coupled and/or embedded practices usually used.
- The type of services (i.e. functions) and data usually used.

The output of these sub-groups will feed the sub-groups covering the *generalization phase* aiming to derive the recommended practices. One sub-group will be responsible to define the semantics and data structure for the standard services, considering the typical data structures and the existing standards for communication in automation domains, namely ISO 9506 (which defines how control information is transferred among intelligent devices and systems such as computers and robots) [30] and IEC 61850 (which is a communication standard for power utility equipment) [31]. Another sub-group will define recommended industrial technologies to create an industrial-like middleware based on M2M protocols, supporting the interface between agents and physical hardware (in a coupled manner).

At this moment, IEEE P2660.1 is running its on-going activities, seeking contributions coming from the automation and control engineers, software engineers, system integrators, and automation system vendors.

## VI. CONCLUSIONS AND FUTURE WORK

During the last years, MAS has been pointed out as a promising solution to design control systems addressing the emergent visions of "Industrie 4.0" and Industrial Internet, demanding more flexible, robust, intelligent, and self-adaptive systems. This promising perspective has not yet a desired correspondence in a wider industrial adoption of this technology. Several studies and surveys point out a standardization need as a critical factor for this situation, especially the lack of standards in the field and the missing compliance of existing standards by the agent-based solutions.

The objective of this paper is to describe the work that is being done by IEEE IES Technical Committee on Industrial Agents, that recognized the importance of this problem and is being working on identifying the standardization needs in

applying and deploying agent-based technology in industrial environments, addressing industrial-specific requirements imposed by different application fields. For this purpose, the IEEE-SA has launched the IEEE P2660.1 Working Group aiming to develop recommendation practices on industrial agents focusing the integration of software agents and low level automation functions. This issue assumes an important role in the implementation of CPS, since the integration of agents (i.e., cyber parts) with the physical automation counterparts (i.e., physical parts) is mandatory for such kind of industrial systems.

The ongoing work in this field, by a structure of several sub-groups, each one analyzing different situations that require the interface between agents and physical devices and functions, may lead to the establishment of a set of recommendation practices that may mitigate the current lacks and boost the adoption of MAS and CPS-based approaches by industry.

#### ACKNOWLEDGMENT

The authors would like to thank the members of the IEEE IES Technical Committee on Industrial Agents as well as the IEEE SMCS Technical Committee on Intelligent Industrial Systems for their fruitful discussions and participation in the IEEE P2660.1 Working Group.

#### REFERENCES

- [1] ACATECH, "Cyber-Physical Systems: Driving force for innovation in mobility, health, energy and production," German National Academy of Science and Engineering (ACATECH), Tech. Rep., Dec. 2011.
- [2] M. Wooldridge, *An Introduction to Multi-Agent Systems*. John Wiley and Sons, 2002.
- [3] P. Leitão and S. Karnouskos, Eds., *Industrial Agents: Emerging Applications of Software Agents in Industry*. Elsevier, Mar. 2015.
- [4] R. Unland, "Chapter 2: Industrial agents," in *Industrial Agents: Emerging Applications of Software Agents in Industry*, P. Leitão and S. Karnouskos, Eds. Elsevier, Mar. 2015, pp. 23–44.
- [5] P. Leitão, "Agent-based distributed manufacturing control: A state-of-the-art survey," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 7, pp. 979–991, Oct. 2009.
- [6] P. Leitão, P. Vrba, and T. Strasser, "Multi-agent systems as automation platform for intelligent energy systems," in *Industrial Electronics Society, IECON 2013 - 39th Annual Conference of the IEEE*, Nov. 2013, pp. 66–71.
- [7] P. Vrba, V. Mařík, P. Siano, P. Leitão, G. Zhabelova, V. Vyatkin, and T. Strasser, "A review of agent and service-oriented concepts applied to intelligent energy systems," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 3, pp. 1890–1903, Aug. 2014.
- [8] P. Leitão, V. Mařík, and P. Vrba, "Past, present, and future of industrial agent applications," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2360–2372, Nov. 2013.
- [9] M. Furmankiewicz, A. Sołtysik-Piorunkiewicz, and P. Ziuziaski, "Artificial intelligence and multi-agent software for e-health knowledge management systems," *Business Informatics*, vol. 2, no. 32, pp. 51–62, 2014.
- [10] FIPA – The Foundation for Intelligent Physical Agents. [Online]. Available: <http://www.fipa.org/>
- [11] A. W. Colombo, S. Karnouskos, J. M. Mendes, and P. Leitão, "Chapter 4: Industrial agents in the era of service oriented architectures and cloud based industrial infrastructures," in *Industrial Agents: Emerging Applications of Software Agents in Industry*, P. Leitão and S. Karnouskos, Eds. Elsevier, Mar. 2015, pp. 67–87.
- [12] S. Karnouskos, V. Havlena, E. Jerhotova, P. Kodet, M. Sikora, P. Stluka, P. Trnka, and M. Tilly, "Chapter 8: Plant energy management," in *Industrial Cloud-based Cyber-Physical Systems: The IMC-AESOP Approach*, A. Colombo, S. Bangemann, Th. and Karnouskos, J. Delsing, P. Stluka, R. Harrison, F. Jammes, and J. Lastra, Eds. Springer, May 2014, pp. 203–218.
- [13] *IEC 61131: Programmable controllers Part 3: Programming languages*, IEC - International Electrotechnical Commission Std., 2012.
- [14] *IEC 61499: Function blocks Part 1-4*, IEC - International Electrotechnical Commission Std., 2012.
- [15] W. Mahnke, S.-H. Leitner, and M. Damm, *OPC Unified Architecture*. Springer, 2009.
- [16] A. Pereira, N. Rodrigues, and P. Leitão, "Deployment of multi-agent systems for industrial applications," in *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)*, Sep. 2012.
- [17] P. Göhner, Ed., *Agentensysteme in der Automatisierungstechnik*. Springer, 2013.
- [18] H. Kagermann, W. Wahlster, and J. Helbig, "Securing the future of German manufacturing industry: Recommendations for implementing the strategic initiative INDUSTRIE 4.0," German National Academy of Science and Engineering (ACATECH), Tech. Rep., 2013.
- [19] OASIS, "Reference Model for Service Oriented Architecture 1.0," Tech. Rep., 2006. [Online]. Available: <http://docs.oasis-open.org/soa-rm/v1.0/>
- [20] FIPA - The Foundation for Intelligent Physical Agents, "FIPA Abstract Architecture Specification," Tech. Rep., 2002. [Online]. Available: <http://www.fipa.org/specs/fipa00001/>
- [21] L. Ribeiro, J. Barata, and P. Mendes, "MAS and SOA: Complementary Automation Paradigms," in *IFIP International Federation for Information Processing*. Springer Boston, 2008, vol. 266, pp. 259–268.
- [22] J. Mendes, P. Leitão, F. Restivo, and A. W. Colombo, "Service-oriented agents for collaborative industrial automation and production systems," in *Holonic and Multi-Agent Systems for Manufacturing, Lecture Notes in Computer Science*, V. Mařík, T. Strasser, and A. Zoitl, Eds. Springer Berlin Heidelberg, 2009, vol. 5696, pp. 13–24.
- [23] D. Greenwood, M. Lyell, A. Mallya, and H. Suguri, "The IEEE FIPA approach to integrating software agents and web services," in *Proceedings of 6th International Joint Conference on Autonomous agents and multiagent systems (AAMAS '07)*, 2007, pp. 1412–1418.
- [24] F. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE*. Wiley, 2007.
- [25] G. Ziyeva, E. Choi, and D. Min, "Content-Based Intelligent Routing and Message Processing in Enterprise Service Bus," in *Proceedings of the International Conference on Convergence and Hybrid Information Technology (ICHIT'08)*, 2008, pp. 245–249.
- [26] C. Marín, L. Mönch, P. Leitão, P. Vrba, D. Kazanskaia, V. Chepegin, L. Liu, and N. Mehandjiev, "A conceptual architecture based on intelligent services for manufacturing support systems," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC'13)*, Oct. 2013, pp. 4749–4754.
- [27] J. Christensen, "HMS/FB architecture and its implementation," in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. Deen, Ed. Springer Berlin Heidelberg, 2003, pp. 53–88.
- [28] T. Strasser and A. Zoitl, "Chapter 5: Distributed real-time automation and control – reactive control layer for industrial agents," in *Industrial Agents: Emerging Applications of Software Agents in Industry*, P. Leitão and S. Karnouskos, Eds. Elsevier, Mar. 2015, pp. 89–107.
- [29] P. Leitão and J. Barbosa, "Building a robotic cyber-physical production component," in *Proceedings of 5th International Workshop on Service Orientation in Holonic and Multi-Agent (SOHOMA15)*, Nov. 2015.
- [30] *ISO/IEC 9506-1: Industrial Automation Systems - Manufacturing Message Specification, Part 1 - Service Definition*, ISO - International Organization for Standardization / IEC - International Electrotechnical Commission Std., 1992.
- [31] *IEC 61850: Communication networks and systems for power utility automation*, International Electrotechnical Commission (IEC) Std., 2010.