Guest Editorial
Special Section on Software Engineering in Industrial Automation

ACCORDING to many undisputable evidences, the complexity and importance of software in industrial automation systems is growing with a tremendous rate, making software engineering one of the major activities of automation systems developers. In general, software engineering is an established discipline with its methods, traditions and curriculum. Most of its methods and tools are applicable across different domains, from general purpose computing to real-time control of manufacturing automation systems.

On the other hand, there is colloquial opinion of software design in automation mainly consisting of programmable logic controllers (PLC) programming in primitive languages, such as ladder logic, done in front of the machine. In the past years, researchers all over the world dedicated much effort to develop and apply advanced software engineering methods and concepts in the industrial automation domain. The research on this subject has followed many directions, for example, component-based design architectures, object-oriented and aspect-oriented design methods and agile development. These methods aim at more efficient construction of software. The emergent ones include the technologies aiming at automatic composition of software functionalities (online or offline) achieved on account of multi-agent collaboration, the use of web-services, or automatic reasoning. These technologies provide for flexibility and plug-and-play composition of software which is enabled by such mechanisms as multi-agent collaboration and service discoveries.

It is our pleasure to present this Special Section on Software Engineering in Industrial Automation of the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, which reports the state-of-the-art on the subject, including review papers and new contributions to this field. This Special Section, covers a wide range of subjects, namely, formal foundations of software engineering, automatic synthesis of programs, the use of knowledge intensive Semantic Web models in manufacturing automation, multi-agent technologies, novel synergies between model-driven engineering and aspect-oriented design and design patterns. Also, the issues related to dependability and correct by design methods are discussed.

This Special Section is opened by paper [1] that proposes a novel approach for controller synthesis on shop-floor level for discrete-event systems. Despite the paper's own merits, another reason for including the paper into the first position in this Special Section was the desire to emphasize the important role and potential of rigorous formal methods in this research area. Formal models of software systems create the firm foundation of software engineering. This is especially true for the industrial automation branch of software engineering.

The purpose of design is to create dependable software with properties that can be guaranteed by design. The specifics of automation systems is that in most cases they are control systems where the dynamics of the plant must be taken into account when control software is designed. Moreover, it can be used as an advantage helping in automatic software design. The ultimate goal is synthesizing formally controllers of industrial systems with subsequent code generation from the formal model.

The approach of [1] is based on a modular modeling formalism and structural properties of models of uncontrolled plant behavior, designed using this modeling formalism. The synthesis procedure takes models of the uncontrolled plant behavior and formal specifications of the desired cyclic plant behavior under control and forbidden states as well. From these ingredients, the Transition Invariant Graph is computed from which the admissible behavior trajectories are extracted. This is performed by partial reachability analysis. An example taken from a lab-scaled manufacturing system illustrates the methodology and shows the application. The proposed approach significantly reduces complexity of the synthesis procedure, making it feasible to be used for application in larger systems of real industrial scale.

Another trend in software engineering is attention to Semantic Web technologies reported in many automation-related works. These technologies are aiming at knowledge representation and automatic manipulation and can be useful for enhancing the performance of software development processes.

The work in [2] presents an approach to using Semantic Web services in managing production processes. In particular, the devices in a production system are considered to expose Web service interfaces through which they can then be controlled. Semantic Web service descriptions formulated in OWL-S make it possible to determine the conditions and effects of invoking the Web services. The approach, demonstrated in the paper, involves three Web services that cooperate to achieve production goals using the domain Web services, one of which maintains a semantic model of the current state of the system, while another uses the model to compose the domain Web services so that they jointly achieve the desired goals. The semantic model of the system is automatically updated based on event notifications sent by the domain services.

The next paper [3] monitors the chronology of research and development of the industrial applications of multi-agent and holonic systems. According to Woodbridge [4], multi-agent

Digital Object Identifier 10.1109/TII.2013.2281278

system (MAS) approach constitutes a novel software engineering paradigm that offers a new alternative to design decision-making systems based on the decentralization of functions over a set of distributed entities. It provides the comprehensive overview of methodologies, architectures and applications of agents in industrial domain from early 1990s up to the present. It also gives an outlook of the current trends as well as challenges and possible future application domains of industrial agents. Industrial agents technology leverages the benefits of multi-agent systems, distributed computing, artificial intelligence techniques and semantics in the field of production, services and infrastructure sectors, providing a new way to design and engineer control solutions based on the decentralization of control over distributed structures. The key drivers for this application are the benefits of agent-based industrial systems, namely, in terms of robustness, scalability, reconfigurability and productivity, all of which translate to a greater competitive advantage of manufacturing enterprises.

Another very important concept of modern software engineering is model-driven engineering (MDE) that is a software development methodology which exploits domain models rather than pure computing or algorithmic concepts. MDE has been promoted as a solution to handle the complexity of software development by raising the abstraction level and automating labor-intensive and error-prone tasks. The next paper [5] presents a Model- Driven Engineering approach, which combines the Unified Modeling Language (UML) and Aspect-Oriented Software Development (AOSD) to design real-time and embedded automation systems.

In computing, aspect-oriented programming (AOP) is a programming paradigm which aims at increasing modularity by allowing the separation of crosscutting concerns. AOP forms a basis for aspect-oriented software development.

The approach of [5] aims at proper handling of nonfunctional system requirements that is a key factor during the design of industrial automation systems. In many application domains they at least are as important as functional requirements. The proposed approach allows a smooth transition from the initial design phases to implementation by using software tools, comprising the system specification and the automatic generation of source code. By combining UML with model-level aspects and a script-base code generation tool, it enables the use of AOSD during system design and implementation, even though the target platform does not natively support such concepts.

The IEC 61499 reference architecture [6] has been conceived to facilitate the development of distributed automation systems with decentralized logic. According to [7], the IEC 61499 standard has substantially contributed to the knowledge of distributed systems design in the industrial automation domain by providing the adequate notation and architecture that is complementary to the traditional PLC programming architecture. As such, the standard has opened new frontiers to explore, one of which is taken in paper [8], that outlines structuring principles and design guidelines for automation programs with purely hierarchical control architectures. The IEC 61499 modeling language pursues a component-based approach, however, it does not define per se how functional hierarchies of upper- and lower-level functional entities. The paper outlines design patterns for implementing hierarchical control solutions in terms of the IEC 61499 and illustrates the approach with a case study.

Along with the design efficiency, dependability of the resulting software systems is another crucial concern. The paper [8] presents a method to develop and implement real-time capable industrial automation software that increases the dependability of production automation systems by means of soft sensors. An application example with continuous behavior used to illustrate the initial requirements. The modeling concept is presented which supports application development and which is supplemented by an implementation approach for standard automation devices, such as PLC.

The Special Section is concluded with paper [10] that represents a practical application of several advanced software engineering techniques in the exciting application area of thermonuclear physics. In this paper, a modeling approach is proposed for the design of real-time applications. The approach is based on SysML modeling language and the MARTe framework that has been recently adopted for the development of real-time systems in several European experimental fusion reactors. The proposed approach achieves better standardization of the development cycle and a better documentation of the developed systems. Furthermore, by using tools it is possible to automatically generate a part of the real-time executable code and the application configuration file. The proposed approach has been applied for the modeling of the control system for the Frascati Tokamak Upgrade.

Although the papers, included in this Special Section are representative of the current progress in the field, they by no means could cover this area anyhow completely. The readers, interested in a more comprehensive overview, are referred to the state-of-the-art survey [11]. The survey's methodology is based on the classic SWEBOK reference document that comprehensively defines the taxonomy of software engineering domain. The survey is bridging it to the classic automation artifacts, such as the set of most influential international standards and dominating industrial practices.

Finally, the Guest Editor wants to give a brief overview to some interesting research trends reflected in the papers submitted to this Special Section but not included due to limited space and high competition. Earlier conference publications by the same authors are taken as reference points.

The enormous popularity of social networks in the last decade has created a new model for communication and collaboration between people that has also impacted on the way how technical systems are engineered. Paper [12] presents the Engineering Cockpit, a social-network-style collaboration platform for automation system engineering project managers and engineers, which provides a role-specific single entry point for project monitoring, collaboration, and management. The Engineering Cockpit increases the team-awareness of engineers and provides project-specific information across the engineering discipline boundaries.

Another clearly seen trend in this research domain is pursuit of methods for automatic generation of control programs for robots and automated machines. Thus, paper [13] investigates the use of a software tool chain to close the gap between the mechanical design and the electrical design of automation systems, showing the possibility of automatic generation of PLC and Human-Machine Interface (HMI) projects. This approach can be seen as a step towards automatic generation of entire automation software from the multitude of related design documentation. The AutomationML [14] that has been suggested as a vendor independent standard for exchange of manufacturing industry engineering data, can enable uniform representation of the knowledge required for such automatic generation.

Paper [15] is representative of the works aiming at automatic generation of control software through using a combination of model-driven engineering methods, UML, and popular industrial standards, such as Computer-Aided Engineering eXchange (CAEX) and PLCopen XML format. The domain models are defined using engineering tools as the design progresses and they can be used to achieve tool integration through model collaboration and model transformations.

Acknowledgment

The Guest Editor would like to thank all the authors who answered the call for this Special Section and all the reviewers who spent their time in reviewing numerous submitted papers, even up to three revisions. In particular, the Guest Editor would like to apologize to those authors whose excellent papers were not included due to limited space and competitive selection process.

Last but not least, special thanks have to be addressed to the managing team of the IEEE Transactions on Industrial Informatics: first, to the Editor in-Chief, Prof. Bogdan Wilamowski, who accepted the special section proposal.

Moreover, warm thanks are extended to Laura Patillo, the Administrator of the journal, for her excellent support during all the phases leading to the actual publication.

Valeriy Vyatkin, *Guest Editor*
Department of Computer Science, Electrical and Space Engineering
Luleå University of Technology
Luleå, 97187 Sweden
valeriy.vyatkin@ltu.se

References

[1] H.-C. Lapp and H.-M. Hanisch, "A new DES control synthesis approach based on structural model properties," *IEEE Trans. Ind. Informat. (Special Section on Software Engineering in Industrial Automation)*, vol. 9, no. 4, pp. 2340–2348, Nov. 2013.

[2] J. Puttonen, A. Lobov, and J. M. Lastra, "Semantics-based composition of factory automation processes encapsulated by web services," *IEEE Trans. Ind. Informat. (Special Section on Software Engineering in Industrial Automation)*, vol. 9, no. 4, pp. 2349–2359, Nov. 2013.

[3] P. Leitão, V. Mařik, and P. Vrba, "Past, present, and future of industrial agent applications," *IEEE Trans. Ind. Informat. (Special Section on Software Engineering in Industrial Automation)*, vol. 9, no. 4, pp. 2360–2372, Nov. 2013.

[4] M. Wooldridge*, An Introduction to Multi-Agent Systems*. New York, NY, USA: Wiley, 2002.

[5] M. A. Wehrmeister, C. E. Pereira, and F. J. Rammig, "Aspect-oriented model-driven engineering for embedded systems applied to automation systems," *IEEE Trans. Ind. Informat. (Special Section on Software Engineering in Industrial Automation)*, vol. 9, no. 4, pp. 2373–2386, Nov. 2013.

[6] *Function blocks—Part 1 Architecture*, IEC61499-1, International Electrotechnical Commission, Geneva, 2005, International Standard.

[7] V. Vyatkin, "IEC 61499 as enabler of distributed and intelligent automation: state-of-the-art review," *IEEE Trans. Ind. Informat.*, vol. 7, no. 4, pp. 768–781, Nov. 2011.

[8] A. Zoitl and H. Prähofer, "Guidelines and patterns for building hierarchical automation solutions in the IEC 61499 modeling language," *IEEE Trans. Ind. Informat. (Special Section on Software Engineering in Industrial Automation)*, vol. 9, no. 4, pp. 2387–2396, Nov. 2013.

[9] D. Schütz, A. Wannagat, C. Legat, and B. Vogel-Heuser, "Development of PLC-based software for increasing the dependability of production automation systems," *IEEE Trans. Ind. Informat. (Special Section on Software Engineering in Industrial Automation)*, vol. 9, no. 4, pp. 2397–2406, Nov. 2013.

[10] G. De Tommasi, R. Vitelli, L. Boncagni, and A. C. Neto, "Modeling of MARTe-based real-time application with SysML," *IEEE Trans. Ind. Informat. (Special Section on Software Engineering in Industrial Automation)*, vol. 9, no. 4, pp. 2407–2415, Nov. 2013.

[11] V. Vyatkin, "Software engineering in industrial automation: State of the art review," *IEEE Trans. Ind. Informat.*, vol. 9, no. 3, pp. 1234, 1249, Aug. 2013.

[12] T. Moser, R. Mordinyi, D. Winkler, and S. Biffl, "Engineering project management using the engineering cockpit: A collaboration platform for project managers and engineers," in *Proc. IEEE 9th Int. Conf. Ind. Informat. (INDIN)*, Jul. 26–29, 2011, pp. 579–584.

[13] P. Falkman, E. Helander, and M. Andersson, "Automatic generation: A way of ensuring PLC and HMI standards," in *Proc. IEEE 16th Conf. Emerging Technol. Factory Autom. (ETFA)*, Sep. 5–9, 2011, pp. 1–4.

[14] AutomationML, Date of access: 25-07-2013. [Online]. Available: https://www.automationml.org/o.red.c/home.html

[15] E. Estevez and M. Marcos, "Model-based validation of industrial control systems," *IEEE Trans. Ind. Informat.*, vol. 8, no. 2, pp. 302–310, May 2012.

Valeriy Vyatkin graduated with the Engineer degree in applied mathematics and the Ph.D. and Dr. Sci. degrees from Taganrog State University of Radio Engineering (TSURE), Taganrog, Russia, in 1988, 1992, and 1998, respectively, and the Dr. Eng. degree from Nagoya Institute of Technology, Nagoya, Japan, in 1999.

He is a Chaired Professor of Dependable Computation and Communication Systems at the Luleå University of Technology, Luleå, Sweden. He is also part-time with The University of Auckland, Auckland, New Zealand, where he has been an Associate Professor and Director of the InfoMechatronics and Industrial Automation Lab (MITRA) at the Department of Electrical and Computer Engineering. His previous faculty positions were with Martin Luther University of Halle-Wittenberg, Germany (Senior Researcher and Lecturer, 1999–2004), and with TSURE (Associate Professor, Professor, 1991–2002). Research interests are in the area of dependable distributed automation and industrial informatics, including software engineering for industrial automation systems, distributed architectures and multi-agent systems applied in various industry sectors: smart grid, material handling, building management systems, reconfigurable manufacturing, etc. He is also active in research on dependability provisions for industrial automation systems, such as methods of formal verification and validation, and theoretical algorithms for improving their performance.

Dr. Vyatkin has been awarded the Andrew P. Sage Award for the Best IEEE Transactions Paper in 2012.