Guest Editorial
Special Section on Formal Methods in Manufacturing

DESIGN and operation of manufacturing systems is a large and complex domain of application. The complexity of this domain stems from the large dimension and from the degrees of flexibility of such systems that are highly parallel and distributed.

Formal methods are mathematical techniques, often supported by tools, for developing man-made systems. They are extensively used in the software engineering domain, where several specialist journals and conferences with very competitive submissions processes exist, showing the vitality of this research area.

In the last years, we have seen advances and developments in the newly emerging areas of technology, as well as actual and potential applications to flexible manufacturing systems. However, the development of suitable methodologies has not kept up with the fast technological changes. There exists a gap between the level of complexity inherent to manufacturing systems and the very large potentiality of formal methods which, nevertheless, are not presently completely mastered.

As in the case of software engineers, mathematical rigor enables manufacturing engineers to handle fundamental design principles such as abstraction or modular and hierarchical development, and to deal with typical engineering problems and quality goals, like reliability, flexibility, and maintainability. Formal methods are a fault avoidance technique that yields both a deeper understanding of the system, thus helping to cover holes in the specification, and an improved system reliability, through verification and validation of the desired properties. Automata, statecharts, queueing networks, Petri nets, process algebras, and temporal logic-based models are becoming more and more used for an integrated view of the specification, validation, performance evaluation, control and scheduling of manufacturing systems.

Even if international conferences and research meetings in the manufacturing domain sometimes include specific sessions on formal methods, there do not exist, as it is the case in the software engineering field, specialized journals or conferences on the topic. Therefore, special sections like this one, both in journals and international meetings, are important to present some of the most significant research works representing the state-of-the-art in the area of formal methodologies for manufacturing systems.

In response to the call for papers for this Special Section on *Formal Methods in Manufacturing*, we received in total 27 submissions. From them, and with the help of 104 Reviewers, five papers were finally accepted.

Digital Object Identifier 10.1109/TII.2010.2042529

Formally, a system is said to be observable if, for any possible sequence of state and control vectors, the current state can be determined in finite time using only the outputs. The paper "Observability of Switched Linear Systems," by David Gómez-Gutiérrez, Guillermo Ramírez-Prado, Antonio Ramírez-Treviño, and José Ruiz-León deals with the problem of observability in a class of hybrid systems. In the systems under consideration, the discrete evolution is modeled using interpreted Petri nets while the continuous part of the state space evolves according to a family of linear systems. The classical concept of distinguishable systems for the observability of hybrid systems is interpreted for the considered model by extending previous results in the literature for the case of controlled systems. Then, detectability of the commutation time is introduced and used to study the observability of the switched linear systems.

IEC 61499 is an open standard architecture for developing distributed control and measurement applications. The basic building block of IEC 61499 is the function block, from which entire applications may be built. Each function block has event inputs and outputs as well as data inputs and outputs. However, the standard has no formal semantics and different implementations of the standard have emerged making different assumptions to cope with the ambiguities. As a consequence, the same application may behave differently when it is executed on different implementations, thus resulting in nonportable applications. The paper "On Formal Analysis of IEC 61499 Applications, Part A: Modeling," by Goran Čengić and Knut Åkesson, proposes a formal framework for both the IEC 61499 application and the runtime execution semantics separately. The application model can capture basic function blocks, service interface function blocks and their networks. The abstract mathematical model of the execution can help in capturing of different runtime execution semantics. Together, the models can be used to formally analyze and compare how an application would behave when executed using different execution semantics.

The companion paper to the previous one, "On Formal Analysis of IEC 61499 Applications, Part B: Execution Semantics," by Goran Čengić and Knut Åkesson, shows how different execution models can be compared with the formalism introduced in the previous paper. Three different execution semantics, namely, buffered sequential execution model (BSEM), non-preempted multithreaded IEC 61499 resource (NPMTR), and cyclic buffered execution model (CBEM), are mathematically defined. The mathematical definitions can be used to analyze an application's behavior when executed using those execution semantics. The mathematical definitions have been used as a basis for implementation of a runtime environment

and a software tool that generates formal models suitable for formal verification. Formal verification can be used to help discover execution errors before the application is executed on the factory floor.

In the software engineering domain, refactoring is the process of changing the internal structure of a program without modifying its functionality, in order to improve internal quality attributes of the software. The paper "Refactoring of Execution Control Charts in Basic Function Blocks of the IEC 61499 Standard," by Valeriy Vyatkin and Victor Dubinin, presents a model transformation method for refactoring of execution control charts (ECCs) of function blocks of IEC 61499. This leads to a technique for the ECCs improvement allowing the removal of conditionally dead states, thus possibly avoiding potential deadlocks. In order to achieve that goal, a formal model of ECCs is introduced, and refactoring rules are defined on the model by means of graph transformations. A software implementation based on the AGG tool for algebraic graph transformation is also presented and used to evaluate the proposed technique for a battery of ECCs representing typical structures.

The paper "Modeling Distributed Transportation Systems Composed of Flexible Automated Guided Vehicles in Flexible Manufacturing Systems," by David Herrero-Pérez and Humberto Martínez-Barberá, presents a methodology to design and implement flexible automated guided vehicles (AGVs) using an interesting and novel combination of techniques that have been properly chosen and integrated. The AGVs incorporate artificial intelligent techniques to facilitate the configuration and adaptation when there are layout modifications and to simplify the interaction between them using simple coordination models. In order to achieve higher flexibility, the AGVs team coordination problem is divided into three subproblems (task allocation, task execution, and collision free motion) for the solution of which a decentralized navigation control and a distributed Petri net approach are incorporated. The methodology has been tested in a real factory where the system is currently in operational use.

The preparation of a Special Section like this relies on an important amount of work performed by many people. We would like to thank to all of them for their active support: The Authors, for their original contributions and their prompt reply considering the received suggestions. The Reviewers, who did an excellent job with careful readings and comprehensive and useful comments that contributed to enhance the quality of papers. The Editor-in-Chief and the rest of Editorial Team of the TRANSACTIONS, for the help and guidance during all the preparation process of this Special Section.

JAVIER CAMPOS, *Guest Editor*
Universidad de Zaragoza
Zaragoza, 50009 Spain
jcampos@unizar.es

REFERENCES

[1] C. A. Vissers, M. van Sinderen, and L. F. Pires, "What makes industries believe in formal methods," in *Protocol Specification, Testing and Verification, XIII*, ser. IFIP Transactions 16, A. Danthine, G. Leduc, and P. Wolper, Eds. Amsterdam, The Netherlands: North-Holland, 1993, vol. C-16, pp. 3–26.

[2] *Proceedings International Workshop on Formal Methods and Manufacturing*, J. C. Gentina, A. Giua, and M. Silva, Eds. Zaragoza, Spain: Prensas Universitarias de Zaragoza, Sep. 1999.

[3] I. Castillo and J. S. Smith, "Formal modelling methodologies for control of manufacturing cells: Survey and comparison," *J. Manuf. Syst.*, vol. 21, no. 1, pp. 40–57, 2002.

[4] G. Morel, P. Valckenaers, J.-M. Faure, C. E. Pereira, and C. Diedrich, "Manufacturing plant control challenges and issues," *Control Engineering Practice*, vol. 15, pp. 1321–1331, 2007.

[5] T. L. Johnson, "Improving automation software dependability: A role for formal methods?," *Control Engineering Practice*, vol. 15, pp. 1403–1415, 2007.

[6] A. Matta, M. Rossi, P. Spoletini, D. Mandrioli, Q. Semeraro, and T. Tolio, "FM for FMS: Lessons learned while applying formal methods to the study of flexible manufacturing systems," *Lecture Notes in Computer Science*, vol. 4711, pp. 366–380, 2007.

[7] J. Woodcock, P. G. Larsen, J. Bicarregui, and J. Fitzgerald, "Formal methods: Practice and experience," *ACM Computing Surveys*, vol. 41, no. 4, pp. 1–36, Oct. 2009.

Javier Campos was born in Jaca, Spain, in 1963. He received the M.Sc. degree in applied mathematics and the Ph.D. degree in systems engineering and computer science from the University of Zaragoza, Zaragoza, Spain, in 1986 and 1990, respectively.

In 1986, he joined the Department of Computer Science and Systems Engineering, University of Zaragoza, where he was the Director from 2001 to 2003. In 2005, he was named Full Professor of Languages and Computing Systems at the Department of Computer Science and Systems Engineering. He has supervised the completion of more than 90 Master Thesis students and two Ph.D. students. Since 1989, he coauthored about 80 papers published in refereed journals and conferences. He has been plenary session invited speaker and tutorial invited speaker in several important meetings. He has served in the Program Committee, sometimes as a Chair, of several international conferences. He is a Founding Member of the Aragón Institute for Engineering Research. His research interests include modeling and performance evaluation of distributed and concurrent systems, Petri nets, software performance engineering, and DESs in automation.

Dr. Campos is a member of the IEEE IES Technical Committee on Factory Automation and Associate Editor of the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS.