Discrete Event and Hybrid Systems in Robotics and Automation

Tarek M. Sobh and Beno Benhabib

Department of Computer Science and Engineering
Robotics, Intelligent Sensing and Control (RISC) Laboratory
169 University Avenue
University of Bridgeport
Bridgeport, CT 06601
and
Computer-Integrated-Manufacturing Laboratory (CIMLab)
Department of Mechanical and Industrial Engineering
University of Toronto
5 King's College Rd.
Toronto, Ontario
M5S 3G8 Canada

1 Introduction

Welcome to the Special Issue on *Applications of Discrete Event and Hybrid Systems in Robotics and Automation* of the *IEEE Robotics and Automation Magazine*. This issue is intended for engineers, scientists, and practitioners who are interested in modeling, analysing, observing, controlling, simulating and monitoring complex robotic and manufacturing computer-controlled systems.

2 Discrete Event and Hybrid Systems in Robotics and Automation

The underlying mathematical representation of complex robotic and manufacturing computer-controlled systems is still insufficient to create a set

of models which accurately captures the dynamics of the system over the entire range of system operation. We remain in a situation where we must tradeoff the accuracy of our models with the manageability of the models. Closed-form solutions of mathematical models are almost exclusively limited to linear system models. Computer simulation of nonlinear, hybrid and discrete-event models provide a means for off-line design of robotic control systems. Guarantees of system performance are limited to those regions where the robustness conditions apply. These conditions may not apply during startup and shutdown or during periods of anomalous operation.

Attempts have been made to model low and high-level system changes in automated and robotic systems as discrete event dynamic systems (DEDS) and hybrid systems. Several attempts to improve modeling capabilities are focused on mapping the continuous world into a discrete one. However, repeated results are available which indicate that large interactive systems evolve into states where minor events can lead to a catastrophe. Discrete event and hybrid systems have been used in the manufacturing and automation domains to model system state changes within a process. Timed and untimed petri nets and state automata, in addition to markovian, stochastic, perturbation and other models have been used extensively to model and control automated manufacturing systems. High level DEDS controllers have also been to guide the behavior of robots based on sensory outputs.

As industries move closer to implementing agile-manufacturing concepts the need for automatic and re-programmable controllers will increase rapidly [1,2]. The productivity of flexible manufacturing systems (FMSs) in such industries will be measured in terms of: (i) device flexibility - use of reconfigurable and re-programmable machines for part production, and robotic manipulators for part transfer; and, (ii) system flexibility - use of a supervisory controller to re-program the operation of the FMS, in order to accommodate alternate production routes when needed [3].

Agile manufacturing is primarily characterized by "the ability to rapidly respond to continuously changing customer requirements." Therefore, it is assumed that in a FMS, parts could be re-routed in an on-line manner in response to such changes, as well as in response to un-expected device failures or deadlocks, without intervention from an external agent.

A centralized supervisory controller for a FMS must perform the follow-

ing three tasks: (1) monitor the behaviour of the system using sensory feedback, (2) evaluate phenomena in accordance with the governing supervisory-control strategy, and (3) enforce the comman the strategy through the execution of the device programs [4]. The design of a supervisory controller entails the formulation of control laws, and the synthesis of supervisors. The laws specify how the supervisor is to react to the behavior of the FMS, the goal being to have some production specifications satisfied within the standing control-enforcement constraints.

From a planning and control perspective, a FMS for discrete production can be seen as a dynamic system whose states evolve according to the occurrence of abrupt physical events, thus exhibiting the characteristics of a discrete-event system (DES). Such systems are event driven, discrete in time and space, usually asynchronous, and typically non-deterministic.

In the past, DESs have usually been sufficiently simple that intuitive or ad-hoc control solutions have been adequate [5]. However, the increasing complexity of these systems has created a need for formal approaches for their analysis and control. The essential distinction between an ad-hoc approach and a formal approach is that the latter one provides a mathematical framework (e.g., algebraic set theory, formal language theory, etc.) for the formulation and synthesis of the supervisory-control laws. With the use of mathematical tools developed within the formal approach, the synthesized supervisory-control law is (mathematically) proven to be free of conflict and deadlock. Petri-net theory [6,7], real-time temporal logic [8,9] and controlled automata [10] are formal approaches that have been commonly applied to the analysis and control of DESs.

Because of the non-deterministic nature of behaviour of a manufacturing system, its supervisory control must be carried out in a closed loop. The above mentioned traits complicate and greatly increase the complexity of the supervisory-control implementation. Thus, the control of even a moderately-complex system can easily require an immensely large DES strategy. In [11], it has been shown that, when solving basic control-synthesis problems, although they have been noted to be of polynomial complexity in the number of states, the number of states in a practical system can be exponential in the number of constituent processes.

To some extent, this problem of excessive states can be mitigated through

modular synthesis, use of aggregation, decentralization, and hierarchies [12]. However, these techniques are of limited use, since they draw on special characteristics of the numerous control objectives. In a complex environment with many interdependencies between objectives, these techniques will not be sufficient for reducing the number of states to manageable numbers.

If it is not possible to construct, for a given set of control objectives, a DES supervisory controller having a manageable number of states, it might be possible to create a split approach that uses some alternative mechanism in addition to a DES supervisory controller. This second mechanism would relieve the DES supervisory controller of the need for so many states by either (i) taking on the responsibility for some of the control objectives, or (ii) asserting control whenever events diverge from the (reduced number of) states of the DES supervisory controller [4].

A second problem to be addressed is the selection of hardware that can be easily re-programmed on-line. The utilization of personal computers (PC), augmented with data-acquisition and interface devices, as well as programmable-logic-controller (PLC) technology have been proposed in the literature for the execution of control strategies generated on-line. A primary reason for the selection of PLCs over PCs is that they are standard, rugged manufacturing hardware widely used in factory automation.

2.1 Applications of Ad-hoc Approaches to FMS Control

In [5], a workcell-management concept is introduced for the integration of workcell-programming, workcell-coordination and error- recovery issues. A manufacturing-workcell programming language is also proposed. Its basic features permit the evaluation of mathematical, relational and logical expressions, the assignment of variables, the conditional and unconditional branching of program flow, and looping. The management system was tested using multiple PCs in a token-ring environment to simulate a manufacturing workcell.

In [13], the conceptual design and partial implementation of an on- line supervisor for a robotic assembly workcell is described. The proposed supervisor is defined as an on-line system responsible for the real-time monitoring of the assembly process. The proposed system was implemented on a computer workstation connected directly to an assembly workcell. In

[14], a knowledge-based on-line system is proposed for scheduling, execution monitoring, and failure diagnosis and recovery for a flexible-assembly-cell environment. It is a hierarchical system with three levels: a task level, a functional level and an action level. For the implementation, a PC was used to host the supervisory controller, which was in turn connected to dedicated microprocessors utilized to control the workcell devices.

A common shortcoming to the above supervisory-controllers is their lack of formalism to verify and ensure correctness of the control strategies (i.e., free of conflict and of deadlock).

2.2 Application of Formal Approaches to FMS Control

Both Petri-net and controlled-automata DES-modeling techniques have been utilized for PC-based implementations of supervisory control [15-18]. In [7], an extended Petri-net notation is introduced for the modeling and control of a manufacturing system. In [15, 16], a scheduling and control system for manufacturing workcells utilizes Petri nets for the local control of the workcell devices at the machine level in a hierarchical system. The modeling and performance evaluation of Petri- net models applied to manufacturing workcells are discussed in [17]. In [18], an augmented timed Petri-net system is introduced for handling failures within a robotic flexible-assembly workcell.

A Petri-net operating system has also been developed as the basis for a controller, [19, 20]. A Petri-net description language is proposed to convert a graphical (Petri-net) model of a manufacturing system into a textual representation. The output of the conversion process is a set of (English) statements, where for each state and transition the corresponding preceding and following states and transitions are listed as a single statement. To supervise the workcell, the Petri-net operating system resided in a PC, that was in turn interfaced to other PCs which acted as local controllers of the machines in the workcell.

In a rare PLC-based controller example, Petri-net modeling is used as an intermediate step in moving from a high-level description of a control-strategy to the Boolean format of the corresponding ladder-logic description [21]. The conversion from Petri-net to ladder-logic code is performed by a set of transformation rules defined in the work.

One of the two controlled-automata-based implementations reported so far has been for the supervisory control of an integrated- circuit wafer-fabrication system [22]. The supervisory controller consists of two parts: a supervisor and a controller. The role of the supervisor is to ensure that safety constraints are enforced. The role of the controller is to direct the system toward the desired goal, that is, to accomplish a specific set of tasks. In the implementation of the controller, a dedicated computer workstation was utilized and directly connected to the device.

As another real implementation example, controlled automata is utilized in [3] for the supervisory control of a robotic manufacturing workcell. The control strategy is developed based on the framework presented in [10], automatically translated into a ladder-logic code, and subsequently downloaded into the PLC. The PLC program was then executed to control the workcell devices.

Some DES-type controllers have also been targeted for the control of power plants using learning automata [23], satellite stabilization through inductive learning [24], and flexible-wing aircraft control using fuzzy logic [25].

3 Conclusions

This special issue is devoted to discrete event and hybrid systems applications that are relevant in the robotics and automation area. The focus of the issue is to present some problems, modeling strategies and tools related to robotics and automation in which discrete event and hybrid systems frameworks play a significant role. We hope you will enjoy the papers in this issue.

References

[1] L. O'Connor, "Agile Manufacturing in a Responsive Factory," ASME, Mechanical Engineering Magazine, 1994, Vol. 116, No. 7, p. 54.

[2] "Manufacturing a la Carte: Agile Assembly Lines, Faster Development Cycles," IEEE, Spectrum, Vol. 30, No. 9, 1993, pp. 24-85.

[3] S.C. Lauzon, J.K. Mills, and B. Benhabib, "An Implementation Methodology for the Supervisory Control of Flexible Manufacturing Workcells," SME, Journal of Manufacturing Systems, 1997, Vol. 16, No. 2.

[4] R.A. Williams, B. Benhabib, and K.C. Smith, "A DES-Based Supervisory Control System for Manufacturing Systems," SME, Journal of Manufacturing Systems, 1996, Vol. 15, No. 2, pp. 71-83.

[5] B. Benhabib, C.Y. Chen and W.R. Johnson, "An Integrated Manufacturing Workcell Management System," Journal of Manufacturing Review, 1989, Vol. 2, No. 4, pp. 266-276.

[6] T. Murata, "Petri Nets: Properties, Analysis and Applications," Proceedings of the IEEE, 1989, Vol. 77, No. 4, pp. 541-580.

[7] H. Van Brussel, Y. Peng, and P. Valckenaers, "Modeling Flexible Manufacturing Systems Based on Petri Nets," CIRP Annals on Manufacturing Technology, 1993, Vol. 42, No. 1, pp. 479-484.

[8] F. Lin, "Analysis and Synthesis of Discrete Event Systems Using Temporal Logic," Journal of Control-Theory and Advanced Technology, 1993, Vol. 9, No. 1, pp. 341-350.

[9] J.S. Ostroff and W.M. Wonham, "A Framework for Real-Time Discrete-Event Control," IEEE, Transactions on Automatic Control, 1990, Vol. 35, No. 4, pp. 386-397.

[10] P. Ramadge and W.M. Wonham, "The Control of Discrete Event Systems," Proceedings of the IEEE, 1989, Vol. 77, No. 1, pp. 81-89.

[11] Y-C. Ho, "Performance Evaluation and Perturbation Analysis of Discrete Event Dynamic Systems," IEEE, Transactions on Automatic Control, 1987, Vol. 32, No. 7, pp. 563-572.

[12] W.M. Wonham and P.J. Ramadge, "Modular Supervisory Control of Discrete-Event Systems," Math. Control Signals Systems, 1988, Vol. 1, pp. 13-30.

[13] M.C. Moed and R.B. Kelley, "An On-line Supervisor for Robotic Tasks," IEEE, International Conference on Computer Integrated Manufacturing, Troy (NY), 1988, pp. 274-281.

[14] R. Alami and H. Chochon, "NNS, A Knowledge-Based On-line System for an Assembly Workcell," IEEE, International Conference on Robotics and Automation, New York, 1986, pp. 603-609.

[15] J. Long, B. Descotes-Genon, and P. Ladet, "Hierarchical and Intelligent Control of Flexible Manufacturing Systems," 7th IFAC Symposium on Information Control Problems in Manufacturing Technology, 1992, Vol. 1, pp. 243-248.

[16] J. Long, B. Descotes-Genon, and P. Ladet, "Distributed Intelligent Control and Scheduling of Flexible Manufacturing Systems," 8th International Conference on CAD/CAM, Robotics and Factories of the Future, 1992, Vol. 2, pp. 1760-1767.

[17] F. DiCesare, G. Harlakis, J.M. Proth, M. Silva and F.B. Vernadat, Practice of Petri Nets in Manufacturing, Chapman & Hall, New York, 1993.

[18] K. Venkatesh, M. Kaighobadi, M. Zhou and R.L. Caudill, "Augmented Timed Petri Nets for Modeling, Simulation, and Analysis of Robotic Systems with Breakdowns," SME, Journal of Manufacturing Systems, 1994, Vol. 13, No. 4, pp. 289-301.

[19] A.A. Desrochers and R.Y. Al-Jaar, Application of Petri Nets in Manufacturing Systems, IEEE Press, Piscataway (NJ), 1995.

[20] M. Zhou and F. DiCesare, Petri Net Synthesis for Discrete Event Control of Manufacturing Systems, Kluwer Academic Publishers, Boston, 1993.

[21] M.A. Jafari and T.O. Boucher, "A Rule-Based System for Generating a Ladder-Logic Control Program from a High-Level Systems Model," Journal of Intelligent Manufacturing, 1994, Vol. 5, pp. 103-120.

[22] S. Balemi, "Discrete Event Systems Control of a Rapid Multiprocessor," 7th IFAC Symposium on Information Control Problems in Manufacturing Technology, Toronto, Canada, 1992, pp. 53-58.

[23] H.E. Garcia, A. Ray and R.M. Edwards, "A Reconfigurable Hybrid System and its Application to Power-Plant Control," IEEE, Transactions on Control Systems and Technology, Vol. 3, No. 2, 1995, pp. 157-170.

[24] M. Lemmon, P. Anysaklis, X. Yang and C. Lucisano, "Control System Synthesis Through Inductive Learning of Boolean Concepts," IEEE, Control Systems Magazine, June 1995, pp. 25-36.

[25] S. Chiu, S. Chand, D. Moore and A. Chaudhary, "Fuzzy Logic for Control of Roll and Moment for a Flexible-Wing Aircraft," IEEE, Control Systems Magazine, June 1991, pp. 42-48.