Challenges and Opportunities in Logic Control for Manufacturing Systems

Report of the NSF Workshop held at the University of Michigan Ann Arbor, MI June 26–27, 2000

Dawn Tilbury and Pramod Khargonekar
Engineering Research Center for Reconfigurable Machining Systems
University of Michigan
Ann Arbor, MI 48109-2125
{tilbury,pramod}@umich.edu

DRAFT
August 9, 2000

Discrete part manufacturing systems typically consist of numerous machines working together in a coordinated and sequential fashion. Programmable logic controllers are widely used to implement the control algorithms for these machines. Systems with hundreds or thousands of inputs and outputs, many of them simple on/off switches, are not uncommon. The logic controller must handle not only the normal operation sequence and synchronization of the machines, but also the operator interface and error handling and recovery routines.

In current industrial practice, PLC's are programmed using a low-level language such as ladder logic, resulting in large and unwieldy programs. The overall design is experience-based, and verification is typically done only through experimentation or simulation. Because of the complexity of the control programs and the manufacturing systems, either verification option is time-consuming and expensive.

Theoretical foundations have been developed to model and control dynamical systems whose evolution is determined by the occurrence of discrete events rather than the values of continuous variables. The two most common frameworks are Petri nets and finite state machines; both allow for formal verification of the correctness of a control system. However, despite significant research advances in recent years, these formal techniques have not been widely employed in industry.

To create an understanding of the gaps which exist between the theory of discrete-event systems and control and their implementation in industrial manufacturing systems, 50 participants from industry and academia were brought together in a workshop held at the University of Michigan in Ann Arbor in June, 2000. The workshop presentations focused on coordination and sequencing problems, complexity management, error handling and recovery, automatic generation of executable code, and control verification.

1 Industrial Challenges in Logic Control

Industrial involvement in the workshop consisted of machine builders (who write the logic control programs), end users (manufacturing companies), and control suppliers (of both hardware and software). To begin the workshop presentations, Bryan Graham from Lamb Technicon (a machine builder) outlined the complexities and challenges faced by logic programmers. Sushil Birla and Jerry Yen from General Motors described control specifications that end-users will require in the future. Presentations from two control suppliers rounded out the industry perspectives: James Christensen from Rockwell Automation discussed the emerging 61499 international standard for distributed logic control, and Bertil Brandin from Siemens described a new product for validating logic controllers.

1.1 Industry Needs

The first breakout session focused on industry needs and state-of-the-art. Participants were broken into four groups focused on the needs of end-users, machine builders, control vendors, and collaboration. Fault detection and diagnostics that are more effective and more integrated with the logic control programs were indicated by the end-users as key future needs. In addition, open architecture was seen as important for flexibility. Machine builders expressed frustration with the lack of clear and precise specifications for writing logic control programs, along with the ever-expanding number of programming languages that they must use and master. The machine builders' wish list includes integrated tools for mechanical and control design and better user-friendly software. The robustness of Windows-based PC controls was debated. Although some participants were in strong support, and currently run Windows-based controls in their factory, it was noted that strict controls are needed to avoid operating system crashes and hence no consensus emerged. The control suppliers group was excited about the recent advances in distributed control with networked communications, but also recognized the challenges that a distributed architecture brings to implementation and verification of control systems. Although standards were seen as important, and as having the potential to eliminate the "glueware" between different control methods, the proliferation of multiple standards has made true standardization difficult in practice. The fourth breakout group discussed collaboration avenues between academia and industry. The most fruitful collaborations were described as specific projects with parallel teams working in both the university and industry. Industry participants expressed a desire for academics to carry out a critical comparison of the capabilities of different logic control approaches. Retraining the existing workforce, both logic programmers and machine operators, was seen as another area for collaboration, especially using new and innovative teaching methods.

1.2 Education

A unifying theme emerging from all four industry breakout sessions was the need for better education in logic control. Since the topic is interdisciplinary, requiring skills from electrical, mechanical, and control engineering as well as computer science, it was acknowledged to be difficult to integrate into existing disciplinary-specific programs. However, an interdisciplinary course on system integration at either the graduate or undergraduate level would be valuable. Due to the existing shortage of logic programmers, the job market is excellent and the pay is substantial.

2 Logic Control Formalisms

The academic presentations were divided into two sessions, one focusing on finite state machine methods for logic control and the other on Petri net methods. Christos Cassandras from Boston University gave an overview of how formal discrete-event system (finite state automata) models can be used to develop logic controllers. Beno Benhabib of the University of Toronto described an implementation of a finite-state-machine-based supervisory logic control program in a PC/PLC-based architecture. A three-level hierarchy (machine, system, and scheduling) for manufacturing logic control was proposed by Placid Ferreira from the University of Illinois, using finite state machine models for the system and control.

Alan Desrochers of Rensselaer Polytechnic Institute showed how Petri nets can be used to model, control, and analyze the performance of manufacturing systems. When modeling a system, places can be used to model machines, and tokens indicate machine availability. MengChu Zhou of New Jersey Institute of Technology showed how Petri nets used as controllers can be more flexible and understandable than standard relay ladder logic implementations. Dawn Tilbury of the University of Michigan also presented a Petri-net based controller, with places used to model operations and tokens indicating active operations. Petri net based controllers can be implemented on industrial-standard PLC's using an IEC 1131-3 standard programming language called Sequential Function Charts, or SFC, which is based on Petri nets.

2.1 Academic State-of-the-Art

The second breakout session focused on academic state-of-the-art and future research directions. The most exciting new developments in academic work in logic control were acknowledged to be improved verification and validation algorithms. The move from hardwired safety controls to software implementations of safety circuits was seen as a significant advance, made possible by improved formal verification. In addition, it was noted that the inclusion of Stateflow with Matlab can integrate finite-state machine control and simulation with continuous-time algorithms, which can be particularly useful for teaching. Petri nets and SFC/Grafcet are more commonly used in Europe than in North America; this can make transferring new Petri net-based control algorithms straightforward.

2.2 Barriers to Implementing Research Results

It was acknowledged, however, that there are significant barriers to implementing research results into in industry. One of the most significant barriers is the development of commercial products which employ the new results. Most logic control programmers learn about new technologies from their vendors, not from reading technical journals. In addition, there is often information lost between the developer of a new tool and the customer—it is often impossible to encode all of the knowledge base into the software product. Industry can also be resistant to change. End users are often cautious about incorporating unproven technology into expensive manufacturing systems with projected lifetimes of ten or more years. Since control vendors traditionally have depended upon proprietary solutions for maintaining and increasing market share, they are hesitant to adopt a new technology for which they don't own the intellectual property rights.

Other barriers to transferring research to practice include the fact that much of the academic work on logic control has only considered the automatic or normal operation cycle of a manufacturing system, whereas up to 90% of the control code may deal with alternate control modes and fault handling/diagnostics. Academics are not always knowledgeable about the latest available technologies used in industry. Industry already perceives that there are too many different logic control programming languages, thus new languages are met with significant resistance. The lack of an agreed-upon standard for logic control programs is also seen as a significant barrier to implementing new research results. The relatively small market for manufacturing control software reduces the cost-effectiveness of introducing new solutions or new products.

The culture clash between industry and academia was discussed often. Industry participants felt that they didn't have enough time to spend with academics, and professors felt that they had a lack of time to spend with industry. Both sides acknowledged the problems associated with a transient workforce: students graduate, and employees switch jobs frequently.

2.3 Overcoming the Barriers

Partnerships between academia and industry can help overcome some of these barriers, and students with internships in industry can create a bridge between the two groups. Joint research projects are a useful mode of cooperation, although it was noted that a company will only see the benefit of the joint project if it has a parallel internal project.

To ensure that the most important problems are studied, industry should assist in research problem formulations. Industry can also propose benchmarking tests which can be used to compare competing approaches to logic control problems. A third-party neutral evaluation of new approaches compared with existing techniques, or comparing competing new approaches, would help ease the reluctance of end-users to adopt new technologies. In addition, there may be much to learn from how other industries address the problems of logic control and coordination. In particular, it was noted that the aerospace and rail transportation industries also deal with the control of large, distributed, complex systems. Standard interfaces can also lower the barrier of entry for new technologies to be adopted—with interoperable systems, a complete solution would not need to be provided by each vendor.

Cultural changes in both industry and academia can also help to overcome the barriers. Many manufacturing companies focus their limited research resources on product development rather than on manufacturing processes. In order to stay competitive, it is recognized that the product design must be superior, but many firms are satisfied with "adequate" manufacturing. In addition, in when a time or budget crunch arises, research projects often take a back seat to getting a new line up and running. Stronger management support of research in manufacturing processes would be helpful in overcoming barriers.

There was a sense among many participants that academia does not reward industrial project-specific research solutions to the same extent that theoretical research is rewarded. On the other hand, it was noted that these attitudes are changing—many universities consider industrial collaborations to be important and valuable. Government funding, such as through NIST projects, can also help overcome the barriers to collaboration.

2.4 Future Research Directions

Topics for future research include improving logic control software design and implementation through integrated diagnostics, better human-machine interfaces, validation, and automatic code generation. As distributed control becomes more popular in implementations, design and analysis techniques for distributed logic control systems will be needed. Effective compositional methods for distributed control systems are lacking—distribution of control functionality often increases the overall complexity. For example, a system with 25,000 I/O points, when implemented in a distributed fashion, often ends up consisting of four systems with 5,000 I/O points each and one system with 22,000 points. Modular control design is also increasing in importance. The most important research issues are at the interfaces or interlocks between different control modules. Specification, modeling, and analysis of these interlocks remains an open area of research.

3 Where do we go from here?

Participants felt that the workshop succeeded in creating a better understanding of the gaps which exist between the theory of discrete-event systems and control and their implementation in industrial manufacturing systems. In order to bridge these gaps, several different avenues of collaboration were suggested. The first goal is to increase the awareness of the issues and challenges in logic control for manufacturing systems, both in industrial practice and in academia, using the web, magazine publications, and meetings/conferences. A list of the "top 10 challenges in logic control for manufacturing systems" was created at the workshop. Each table at dinner came up with a list of challenges, and an informal vote was taken on the second day. The workshop website was expanded to include the workshop recommendations, summaries of the breakout session discussions, along with electronic copies of the talks and posters that were presented at the workshop. A summary of the workshop findings will appear in publications such as *IEEE Robotics and Automation Magazine, IEEE Control Systems Magazine, ASME-DSCD Newsletter, InTech, I&CS, Managing Automation*, and *Control Engineering*. Special sessions will be organized at conferences such as ACC, ASME, SME, IMTS, and National Manufacturing Week, and a similar workshop may be held in 2-3 years to discuss the progress and new challenges.

Jim Mooney from Comaupico has contributed an industrial application that can be used as a benchmarking problem for logic control design approaches. The application, called OP 790, is a stand-alone automatic station. It consists of the conveyor devices for trafficking and locating a pallet, a bolt feeder, the tooling for selecting a separator plate from one of four stacks and placing it on the pump casting, and the tooling for the powerhead spindles. Complete specifications for the logic controller for this station have been provided by Comaupico and are linked from the workshop web page.

The workshop provided an excellent forum for industry and academia to meet together and discuss the challenges in logic control for manufacturing systems. Participants came away from the workshop with a new appreciation for both the challenges and opportunities in this important area. The follow-on activities—articles, conference sessions, future workshops—will expand the reach of the discussions, and continue to build industry-university collaborative relationships.

Top 10 Challenges in Logic Control for Manufacturing Systems

1. Diagnostics
2. Education
3. Theory/Practice Gap
4. Software
5. Verification
6. Standards
7. Automatic Synthesis
8. Programming Languages
9. Discrete-Event Theory
10. Distributed Control