Blackboard Agents for Mixed-Initiative Management of Integrated Process-Planning/Production-Scheduling Solutions Across the Supply Chain

David W. Hildum,[†] Norman M. Sadeh,[†] Thomas J. Laliberty,[‡] John McA'Nulty,[‡] Stephen F. Smith[†] and Dag Kjenstad[†]

† Intelligent Coordination and Logistics Laboratory
Center for Integrated Manufacturing Decision Systems
The Robotics Institute
Carnegie Mellon University
Pittsburgh PA 15213-3890
412.268.{7598,8827,8811,2357} · *fax:* 412.268.5569
{HILDUM,SADEH,SFS,DAG}@CS.CMU.EDU

‡ Software Engineering Laboratory
Raytheon Electronic Systems
Raytheon Company
Tewksbury MA 01876-0901
508.858.{5756,1458} · *fax:* 508.858.5976
LALIBERTY_THOMAS@CAEMAC.MSD.RAY.COM
MCANULTY@CAESUN.MSD.RAY.COM

Abstract

As companies increasingly customize their products, move towards smaller lot production and experiment with more flexible customer/supplier arrangements, they increasingly require the ability to respond quickly, accurately and competitively to customer requests for bids on new products and efficiently work out supplier/subcontractor arrangements for these new products. This in turn requires the ability to rapidly convert standard-based product specifications into process plans and quickly integrate new orders with their process plans into existing production schedules across the supply chain. This paper describes IP3S, a blackboard-based agent for supporting integrated process planning/production scheduling across the supply chain. IP3S agents support concurrent development and dynamic revision of integrated process-planning/production-scheduling solutions across the supply chain, maintenance of multiple problem instances and solutions across the supply chain, flexible user-oriented decision making, declarative representation of control information, the use of a common representation for exchanging information, coordination with other planning/scheduling agents and information sources, and ease of integration with legacy systems. The IP3S agent has been customized for and validated in the context of a large and highly dynamic machine shop at Raytheon's Andover manufacturing facility. Empirical results show an average performance improvement of 23% in solution quality over a decoupled approach to building process-planning/production-scheduling solutions.

Introduction

As companies increasingly customize their products, move towards smaller lot production and experiment with more flexible customer/supplier arrangements such as those made possible by Electronic Data Interchange (EDI) (Swaminathan, Sadeh, & Smith 1995; Goldman, Nagel, & Preiss 1995), they increasingly require the ability to respond quickly, accurately and competitively to customer requests for bids on new products and efficiently work out supplier/subcontractor arrangements for these new products. This in turn requires the ability to rapidly convert standard-based product specifications into process plans and quickly integrate new orders with their process plans into existing production schedules across the supply chain to best accommodate the current load of the facility, the status of machines, fixtures and tools, and the availability of raw materials. To effectively support such capabilities requires bridging the gap between CAD/CAM and production scheduling through the development of integrated process-planning/production-scheduling functionalities. Such functionalities are key to supporting agile manufacturing (Goldman, Nagel, & Preiss 1995) techniques across the supply chain, while maintaining short leadtimes, low inventories and a high level of due date performance.

The concurrent development and dynamic revision of integrated process-planning/production-scheduling solutions in large-scale environments is a complex and often ill-defined problem requiring human intervention to accommodate a range of conflicting considerations. These include engineering considerations (e.g., machine accuracy, tooling requirements), scheduling considerations (e.g., due date and leadtime performance, inventories, resource capacities) and supplier considerations (e.g., pricing, delivery dates). A number of decision variables are also involved, relating to process-planning decisions (e.g., machine and tool selection), scheduling decisions (e.g., release and sequencing decisions), make-or-buy decisions and supply decisions. Effective decision support in such an environment requires user-oriented interactive functionalities that support the rapid development, revision and evaluation of alternative solutions in response to new events (e.g., new order arrivals, requests for bids, delays in supply delivery, machine breakdowns).

The work described in this paper was sponsored by the Advanced Research Projects Agency under contract F33615-95-C-5523. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Advanced Research Projects Agency or the United States Government.

This paper presents IP3S (Sadeh *et al.* 1996), a *blackboard*-based agent for supporting mixed-initiative decision-making and integration functionalities (Smith, Lassila, & Becker 1996), namely:

1. *concurrent development and dynamic revision of integrated process-planning/production-scheduling solutions,* using new analysis and diagnosis tools that enable efficient process-plan development through the early consideration of resource-capacity and production constraints (e.g., the current load of the facility) and greater optimization of production activities through direct visibility of process alternatives and tradeoffs *across the supply chain*
2. *maintenance of multiple problem instances and solutions across the supply chain,* allowing the user to control the development of the problem and explore alternative tradeoffs ("what-if" scenarios) by interactively addressing external events (e.g., new order arrivals, requests for bids, resource breakdowns) and imposing and retracting various assumptions (e.g., different delivery dates, work shifts, resource assignments and requirements), and evaluating the impact of these decisions through the incremental modification of process plans and production schedules
3. *flexible, user-oriented decision making,* allowing the user to take over and guide the construction and revision of solutions at multiple levels
4. *representation of declarative, domain-specific control information* to support extensible automated problem solving
5. *a common blackboard representation* for exchanging process-planning and production-scheduling information
6. *coordination* with other planning/scheduling agents (e.g., suppliers, tool shops) and information sources (e.g., enterprise requirement planning systems, manufacturing execution systems)
7. *portability and ease of integration with legacy systems,* making it possible to quickly customize the system to support the integration of a range of problem-solving tasks (e.g., engineering, design, enterprise-level planning) in a number of environments

The IP3S agent has been customized for and validated in the context of the Raytheon Andover machine shop facility, which consists of roughly 150 CNC machine tools and more than 100 people working over three shifts. Customization of IP3S for this environment involved the use of Raytheon's IPPI process-planning module (Raytheon Company 1993a; 1993b) as the IP3S agent's process-planning knowledge source and Carnegie Mellon's MICRO-BOSS scheduling tool (Sadeh 1994) as its production-scheduling knowledge source.

Integrating Process Planning and Production Scheduling

The technical challenges in effectively integrating process-planning and production-scheduling decisions in a complex and dynamic environment such as Raytheon's machine shop are many. From a pure process-planning perspective, the number of orders that require the generation of new process plans and production of new tools, and the sheer variety of parts and machines (and their various characteristics) present a significant challenge. As in other large machine shops, production scheduling in this environment is also no easy task. Major scheduling challenges include (1) the presence of multiple sources of uncertainty, both internal (e.g., machine breakdowns) and external (e.g., new order arrivals, delays in tool production and raw material delivery), (2) the difficulty in accurately accounting for the finite capacity of a large number of resources operating according to complex constraints, and (3) the need to take into account the multiple resource requirements of various operations (e.g., tools, NC programs, raw materials, human operators).

While considerable progress has been made with respect to software technologies for process planning and finite-capacity production scheduling, very little attention has been given to issues of integration. Except for a few attempts (summarized in (Huang, Zhang, & Smith 1995)), often in the context of small manufacturing environments, process-planning and production-scheduling activities are typically handled independently, and are carried out in a rigid, sequential manner with very little communication. Process alternatives are traded off strictly from the standpoint of engineering considerations, and plans are developed without consideration of the current ability of the shop to implement them in a cost-effective manner. Likewise, production scheduling is performed under fixed process assumptions and without regard to the opportunities that process alternatives can provide for acceleration of production flows. Only under extreme and ad hoc circumstances (e.g., under pressure from shop floor expediters of late orders) are process-planning alternatives revisited. This lack of coordination leads to unnecessarily long order leadtimes and increased production costs and inefficiencies, and severely restricts the ability to effectively coordinate local operations with those at supplier/customer sites, whether internal (e.g., a tool shop) or external (e.g., raw material suppliers).

Even with the support of sophisticated state-of-the-art computer-aided process-planning and scheduling techniques, process planning and production scheduling remain highly interactive processes, where the user has to be able to evaluate alternative decisions based on experience and knowledge that is not easily amenable to computer modeling. Rather than committing to a prespecified decision flow, as in earlier approaches (see (Huang, Zhang, & Smith 1995)), the IP3S blackboard architecture emphasizes a more versatile integration framework where the user can dynamically select between alternative decision flows and control regimes. The resulting shell provides a customizable framework capable of supporting a wide range of integrated process-planning/production-scheduling decision flows, including all three of the approaches identified in (Huang, Zhang, & Smith 1995) as well as a number of more complex hybrids.

IP3S: A Blackboard Agent for Integrated Process Planning/Production Scheduling

The use of blackboard architectures (Erman *et al.* 1980) as a vehicle for integrating multiple sources of knowledge to solve complex problems has been demonstrated in a wide range of application domains. Blackboard architectures emphasize modular encapsulation of problem-solving knowledge within independent knowledge sources. These knowledge source modules work collectively to develop solutions to problems by communicating through a shared data structure, namely, the blackboard.

By explicitly separating domain knowledge—in the case of IP3S, process-planning, production-scheduling and *coordination* knowledge—and control knowledge, blackboard architectures offer several key advantages:

- *flexibility of the control mechanism,* making it possible for the user to select from among a dynamic set of control regimes (e.g., highly interactive control regimes where most decisions are made by the user versus more autonomous regimes where the user specifies high-level tasks or "goals" and lets the system decide how to accomplish them)
- *extensibility of the architecture,* making it particularly easy to add and enhance knowledge sources (e.g., new analysis and diagnosis knowledge sources)
- *ease of integration with legacy systems* through the encapsulation of existing problem-solving systems as knowledge sources
- *reusability of knowledge sources* across multiple domains (e.g., utilizing existing analysis and diagnosis knowledge sources in different scheduling applications)

Figure 1 provides an overview of the IP3S agent architecture. The system consists of a *blackboard,* a *controller,* a collection of *knowledge sources* (KSs)—including a process-planning KS, a production-scheduling KS, a communication KS and several analysis/diagnosis KSs (e.g., a KS to generate resource utilization statistics to help evaluate resource contention in different situations)—and a Motif-based graphical user interface (GUI). The IP3S blackboard, controller, KSs and GUI are implemented in C++. The blackboard (operating as a server), the controller and GUI (operating together as a single client), and the KSs (each operating separately and alternating between the roles of server and client) run as independent processes that communicate with each other using Expersoft's CORBA-based XShell™ environment.

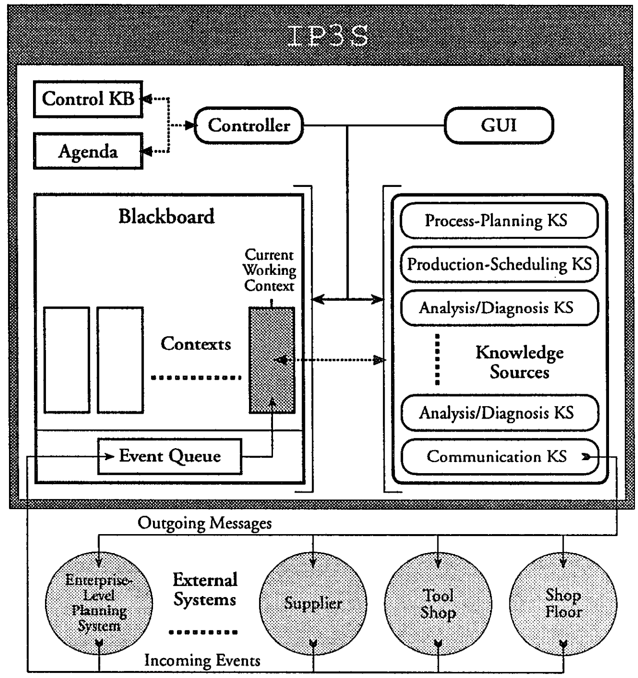

Figure 1: Overview of the IP3S agent architecture

The IP3S Blackboard

The blackboard is the shared data structure on which KSs post solution components (e.g., new process plans and production schedules) and analysis results (e.g., resource utilization statistics). It is partitioned into an arbitrary number of *contexts* that correspond to different sets of working assumptions (e.g., the set of orders that need to be planned and scheduled, available resource capacities, and supplier constraints) and different solutions (e.g., process plans and production schedules). Within each context, a summary of the current state of the solution is maintained in the form of a set of *unresolved issues.* An unresolved issue is an indication that a particular aspect of the current context solution is incomplete, inconsistent or unsatisfactory (e.g., an order lacks a process plan, a resource breakdown conflicts with a reservation, a promised delivery date is violated). Problem solving in IP3S progresses through cycles during which one or more unresolved issue instances are selected to be resolved, a particular method of resolution is selected from among the set of methods applicable to the instance(s), and the method is executed by invoking the appropriate KS. Unresolved issues are created and deleted as a result of (1) KS invocations, (2) the incorporation of external events into a context, and (3) the modification of assumptions within a context to perform "what-if" analysis.

In the remainder of this section we describe the major architectural features of the IP3S blackboard, with an emphasis on the mixed-initiative problem-solving and integration capabilities they support.

Contexts The mixed-initiative decision-support capabilities of IP3S rely heavily on the use of contexts to support the representation of multiple problem instances. A context consists of a collection of resources (including human operators), tools, raw material supplies, a collection of orders (and possibly requests for bids) and their corresponding process plans/production schedules. In addition, the set of unresolved issues represents inconsistencies within a partial solution that must be removed to produce a complete and satisfactory solution. As assumptions are modified and solutions are constructed within a context, the set of unresolved issues is updated to help the system and the user keep track of aspects of the current solution (within that context) that require further problem-solving attention.

The mixed-initiative power of the context mechanism comes from the capability it provides for the user to define a problem progressively and alternately. This can be done through either the incorporation of events into a context (e.g., from other agents or information sources like an enterprise-level planning system, raw material suppliers, a tool shop, the shop floor) or the modification of problem assumptions within a context (e.g., by changing various order and resource attributes such as due dates, work shifts, and supply-availability dates).

Contexts may be created either by the user or automatically by the system. It is through the creation of multiple contexts that "what-if" analysis is supported by IP3S. By creating multiple copies of a context, changing various assumptions within the copies and producing solutions for each, alternate solution paths can be explored. The user or the system can leave a particular context at any point in time and explore other potentially more promising alternatives in other contexts. Changes to order and resource attributes within one context remain local to that context and do not affect other contexts that may include the same entities. When a KS is invoked, its results are visible only to the context in which the user is currently working (called the "current working context").

Events Events received from other planning/scheduling agents and information sources (e.g., suppliers, manufacturing execution systems) are posted on the blackboard *event queue* in preparation for being *incorporated* within one or several contexts by the user or the system. These events include the notification of incoming orders, requests for bids, resource breakdowns and various shop floor updates. When an event is incorporated into a context, the blackboard translates the initial result (or implication) of the action described by the event into an appropriate unresolved issue. The objective for the user or the system is to resolve each such issue, through the activation and execution of one or more KSs, until all events have been incorporated into a context and no more unresolved issues remain.[1]

The event processing mechanism in IP3S supports two important mixed-initiative capabilities:

1. It allows both the user and the system to ignore events that are unlikely to affect the part of the solution upon which work is currently being done. For example, when revising a plan for a part that needs to be processed within the week, incoming-order events for new orders due three months downstream can be ignored.
2. It allows both the user and the system to process conditional events, such as requests for bids. For example, upon receipt of a request for bid on a possible order, a copy of the current context can be created, within which the order can be planned and scheduled. The resulting solution showing the impact of the possible order can then be evaluated to determine a realistic completion date and decide whether or not to submit a bid.

Unresolved Issues As the assumptions within a particular context are modified or as new events are incorporated into a context, the set of unresolved issues within the context is updated automatically by the IP3S blackboard. The set of unresolved issues within a context defines areas in the current partial solution where further problem-solving effort remains to be done to produce a complete, consistent and satisfactory solution. It provides a powerful *workflow management mechanism* that helps IP3S users keep track of the work that remains to be done in a given context.

The IP3S architecture distinguishes between three types of unresolved issues, relating to (1) the *completeness* of the solution, such as an order lacking a process plan or production schedule, (2) *inconsistencies* within the solution, such as outdated resource utilization statistics, and (3) potential areas for solution *improvement*, such as an order with an excessively late completion date or long leadtime. Table 1 provides a sampling of IP3S unresolved issues. To refine a previous statement, note that a complete problem solution for a particular context exists so long as all *completeness*- and *inconsistency*-related unresolved issues have been resolved.

Table 1: A sampling of IP3S unresolved issues

Order related:	Context related:
Completeness:	*Completeness:*
Order-w/o-Process-Plan	Query-Awaiting-Response
Order-w/o-Production-Schedule	*Inconsistency:*
Tool-Completion-Date-Required	Outdated-Resource-Utilization-Statistics
Improvement:	Unprocessed-Shop-Floor-Update
Tardiness	*Improvement:*
	Tardiness

The IP3S Controller

The IP3S Controller is responsible for directing solution construction, revision and analysis, either through close interaction with the user, or on its own with the help of a knowledge base of control heuristics. The primary control-related mixed-initiative capabilities of IP3S manifest themselves in two key Controller functionalities:

1. *support for multiple control regimes*, ranging from a highly interactive mode where the user specifies each problem-solving action to an autonomous mode where the Controller takes responsibility for the selection of which events to incorporate into the current context, the determination of which unresolved issues to resolve, and the selection of the specific methods for their resolution
2. *support for multi-level customizable problem-solving tasks* to provide a range of low- to high-level modes of user interaction (e.g., the activation of a specific low-level KS service, the posting of high-level objectives (or "goals"), the activation of a sequence of services and goals)

IP3S allows the user to select from among different levels of interaction and different control regimes at any time. In addition, the set of high-level problem-solving tasks provided to the user can easily be augmented to accommodate changing user-interaction patterns. Specifically, a hierarchy of high-level goals and scripts can be defined in terms of the basic set

[1] A qualification on this condition will be introduced later.

of services provided by the particular problem-solving systems encapsulated as KSs and incorporated within IP3S.

To support these mixed-initiative capabilities, the IP3S Controller follows an *execution profile* that records the assignment of various problem-solving tasks (e.g., the incorporation of events, the selection of unresolved issues to resolve and the methods for their resolution) to either the system (i.e., the Controller) or the user. The assignment of tasks can be changed at any point by modifying the execution profile. To provide multiple levels of interaction with the system through the definition and activation of aggregate and goal-oriented problem-solving tasks, the IP3S Controller maintains its own declarative *control knowledge base* that links each unresolved issue to the set of problem-solving services applicable for its resolution. The control knowledge base also contains the collection of generic and domain-specific control heuristics that are used by the Controller to perform the tasks assigned to it, as recorded in the execution profile.

The IP3S Controller uses an *agenda* mechanism to keep track of the problem-solving tasks remaining to be executed. When a particular course of action is selected, either manually by the user or automatically through consultation with the appropriate control heuristics, one or several problem-solving task items are placed on the agenda, describing an action or sequence of actions to be performed by the system. The IP3S control architecture supports three types of agenda items:

1. *service* activations, which correspond directly to specific problem-solving services provided by the IP3S KSs
2. *goal* activations, which are used to specify high-level, objective-oriented problem-solving tasks that can be satisfied by the execution of either a service or (more likely) a *sequence* (or "script") of services and subgoals
3. *scripts*, which specify a predefined sequence of KS services and goals generally known to accomplish a particular problem-solving task

The IP3S Problem-Solving Cycle

All problem-solving activity in IP3S is triggered by either the incorporation of a new event (such as an incoming order or a shop floor status update) into the current working context, or the modification of an assumption within the current working context (e.g., "what-if" analysis to evaluate the benefits of adding work shifts or purchasing new machines), both of which can be performed by either the user or the Controller (as specified by the execution profile). The flow of problem solving in IP3S is summarized in Figure 2. It proceeds from the modification of the current working context in a clockwise direction through the following steps: (1) updating the set of unresolved issues within the current working context to reflect the initial problem-solving action, (2) selecting an unresolved issue to resolve, (3) selecting a resolution method for the selected unresolved issue, (4) activating the selected resolution method and (5) executing the problem-solving service that corresponds to the activated resolution method.

The IP3S Controller is invoked whenever there are problem-solving tasks on the agenda remaining to be executed, or, when running automatically (and depending on the execution profile), there are events to incorporate or unresolved issues to resolve.

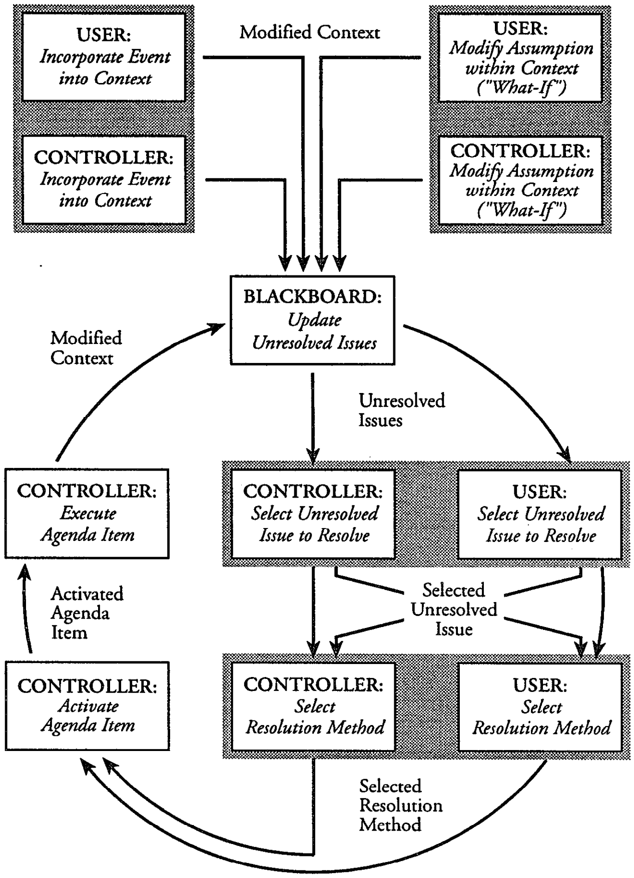

Figure 2: Problem-Solving Flow in IP3S

IP3S Knowledge Sources

Knowledge sources serve as the primary problem solvers in a blackboard system. They communicate their results by posting new information to the blackboard (e.g., new process plans and production schedules) and modifying existing information (e.g., updated process plans and reoptimized production schedules). In IP3S, each domain-level KS acts primarily as a server that supports a variety of problem-solving services. A KS service may require a set of parameters which are defined by the unresolved issue(s) for which the service is applicable.

The IP3S shell relies on two independent problem-solving systems as the KSs responsible for performing the various process-planning, production-scheduling and analysis services. In addition, IP3S is equipped with an internal Communication KS for managing interaction with various external systems. The KSs implemented for the Raytheon customization effort are described below:

- The **Process-Planning KS** is implemented by Raytheon's IPPI process-planning module, which considers both existing and projected resource demand (summarized and posted on the blackboard by the Resource-Utilization KS) to construct process plans consisting of machining-operation sequences that avoid the use of bottleneck resources.

- The **Production-Scheduling KS** is implemented by the MICRO-BOSS system, a dynamic finite capacity scheduling tool that has been shown to support efficient just-in-time operation in complex and dynamic manufacturing environments.
- The **Resource-Utilization Analysis KS** estimates resource contention by accounting for both current reservations within the existing schedule and projected demand from unscheduled orders. Its results are posted on the blackboard for use by the Process-Planning KS.
- The **Communication KS** facilitates coordination between the IP3S agent and other planning/scheduling agents and information sources.

Results

The IP3S agent has been validated in the context of the Raytheon Andover machine shop. Below, we present results of four sets of experiments, each representative of different shop load conditions. For each experiment, solutions were generated using two approaches: a traditional *decoupled* approach where process plans were built independently of load considerations and an integrated approach where process plans were optimized by taking into account the presence of bottlenecks (as indicated by the statistics produced by the Resource-Utilization KS). A threshold parameter was used to determine bottleneck conditions, with values of 30%, 50%, 70% and 90% being tested in each experiment. Table 2 summarizes the results of these experiments.

Table 2: Summary of IP3S Experimental Results

Experiment	Average *Decoupled* Cost	Average *Integrated* Cost	Average Improvement	Average Best Improvement	Average No. of Orders	Average Total Part Qty.
S1	840900	836310	1%	10%	81	7817
S2	3728124	3251212	10%	22%	174	5611
S3	6877796	3730082	37%	52%	66	3798
S4	1328717	945226	44%	69%	76	4727
Avg.	3193884	2190707	23%	38%	99	5489

To facilitate evaluation, process-planning options were restricted to equally satisfactory choices, and solution quality was measured strictly in terms of due date, inventory and lead-time performance. Specifically, a cost was associated with each solution, computed as the weighted sum of the tardiness and inventory costs of all shop orders, with each order weighted by its part quantity. Tardiness penalties were adjusted to be substantially larger than inventory costs to reflect the importance of due date performance in this environment (as is the case in most just-in-time environments).

The results show an average 23% improvement in schedule cost obtained using the integrated planning and scheduling approach facilitated by IP3S, with an average improvement of 38% when taking the best bottleneck-threshold value. Closer analysis indicates that this mainly reflects significant improvements in due date performance, the most important objective in these experiments.

Summary

The IP3S agent is designed around an innovative blackboard architecture that supports flexible mixed-initiative user-oriented management of integrated process-planning/production-scheduling solutions. The system has been customized for a large and highly dynamic machine shop where 50% of incoming orders require the generation or revision of process plans. Empirical evaluation of the system in this environment shows that it can yield substantial performance improvement across a range of load conditions.

References

Erman, L. D.; Hayes-Roth, F.; Lesser, V. R.; and Reddy, D. R. 1980. The Hearsay-II speech understanding system. *Computing Surveys* 12(2):213–253.

Goldman, S. L.; Nagel, R. N.; and Preiss, K. 1995. *Agile Competitors and Virtual Organizations.* New York: Van Nostrand Reinhold.

Huang, S. H.; Zhang, H.-C.; and Smith, M. L. 1995. A progressive approach for the integration of process planning and scheduling. *IIE Transactions* 27(4):456–464.

Raytheon Company. 1993a. *Metal Fabrication Computer Aided Process Planning - Technical Specifications.* RAYCAM document #6685092A.

Raytheon Company. 1993b. *Metal Fabrication Computer Aided Process Planning - User Manual.* RAYCAM document #6685092B.

Sadeh, N. M.; Laliberty, T. J.; Hildum, D. W.; McA'Nulty, J.; Bryant, R. V.; Smith, S. F.; Flood, D.; and Gardner, A. 1996. Development of an integrated process planning/production scheduling shell for agile manufacturing. In *Proceedings, Fifth National Agility Conference.*

Sadeh, N. M. 1994. Micro-opportunistic scheduling: The Micro-Boss factory scheduler. In Zweben, M., and Fox, M. S., eds., *Intelligent Scheduling.* San Francisco CA: Morgan Kaufmann. chapter 4, 99–135.

Smith, S. F.; Lassila, O.; and Becker, M. 1996. Configurable, mixed-initiative systems for planning and scheduling. In Tate, A., ed., *Advanced Planning Technology: Technological Achievements of the ARPA/Rome Laboratory Planning Initiative.* Menlo Park CA: AAAI Press. 235–241.

Swaminathan, J. M.; Sadeh, N. M.; and Smith, S. F. 1995. Impact of supplier information on supply chain performance. Technical report, The Robotics Institute, Carnegie Mellon University, Pittsburgh PA.