Proceedings of SPIE, “Architectures, Networks, and Intelligent Systems for Manufacturing Integration”, Pittsburgh, Pennsylvania, 15-16 October 1997, pp. 108-115.

Contract-Net Based Scheduling for Holonic Manufacturing Systems

P. Kanchanasevee[1], G. Biswas[1], K. Kawamura[1], and S. Tamura[2]
[1]Center for Intelligent Systems, Vanderbilt University
Nashville, Tennessee 37235, USA.
[2]System & Software Engineering Laboratory
Toshiba Corporation, Kawasaki, Japan.

Abstract

Manufacturing is currently undergoing a revolutionary transition with focus shifting from mass production to mass customization. This trend motivates a new generation of advanced manufacturing systems that can dynamically respond to customer orders and changing production environments. It is becoming increasingly important to develop control architectures that are modifiable, extensible, reconfigurable, adaptable, and fault tolerant. Heterarchical control structures, made up of multiple, distributed, locally autonomous entities, provide this kind of control. Our research focus is on efficient and effective scheduling and routing methodologies that can applied to heterarchically controlled manufacturing processes. The Contract-Net based scheduling approach, developed in Distributed Artificial Intelligence (DAI), adopts a multi-agent cooperative problem-solving paradigm based on bidding and negotiation mechanisms to implement production plans as distributed and localized schedules for individual workstations. This paper discusses a Contract-Net based scheduling algorithm in a realistic manufacturing testbed, a model induction motor assembly plant. This testbed, developed as part of the HMS project, is a typical example of low-volume, high-variety production facility, and it highlights many of the problems that arise from the inflexibility of centralized management system architectures.

Keywords: Contract-Net Protocol(CNP), Heterarchical Scheduling, Holonic Manufacturing Systems (HMS), Distributed Artificial Intelligence (DAI).

1. Introduction

Future manufacturing systems need to be dynamically reconfigurable to produce customized products in small batches with fast turn-around times in a cost-efficient manner. This is in contrast to current mass production systems which can produce large batches of standardized products.

The next generation of advanced manufacturing systems need to incorporate:

1. more distributed control to achieve fault tolerance and robustness,
2. machine flexibility, i.e., the ability to change and reconfigure machine functionality to handle a greater variety of product mixes effectively, and
3. product flexibility, i.e., adapt parts/products selection and assembly sequencing to best suit current system configuration.

Other factors that need to be addressed are dynamic reconfiguration to accommodate unusual events, such as machine breakdown and the introduction of new technology and processes as they become available. The goal is to maintain effective and efficient manufacturing operations with minimum downtime in reconfiguring, replanning, and rescheduling manufacturing operations. Holonic Manufacturing Systems (HMS) is focused on the holonic concept as a framework for allowing adaptive reconfiguration.

2. Holonic Manufacturing Systems

The Holonic Manufacturing Systems (HMS) Consortium is an international industry driven project addressing systematization and standardization, research, pre-competitive development, deployment and support of architectures and technologies for open, distributed, intelligent, autonomous and cooperating systems on a global basis (E. H. Van

Leeuwen, and D. Norrie). It is one of ten projects endorsed by the Intelligent Manufacturing Systems (IMS) Steering Committee in May 1995. The basic concepts of holonic systems are developed from Arthur Koestler’s book “The Ghost in the Machine”. Koestler postulated a set of underlying principles to explain the self-organizing tendencies of social and biological systems. He proposed the term “holon” to describe the building blocks of these systems. This term is combination of the Greek word “holos”, meaning “whole”, with the suffix “-on” meaning “part”, as in “proton” or “neuron”. The term reflects the tendencies of real-world agents that act autonomously but can cooperate to form apparent self-organizing hierarchies of sub-systems. A good example is the cell/tissue/organ/system hierarchy in living systems. Koestler used the term “holarchy” to describe these holonic hierarchies. In the manufacturing framework, the HMS Consortium has defined the following key concepts:

Holon: An autonomous and cooperative building block of a manufacturing system for transforming, transporting, storing and/or validating information, and physical objects. The holon consists of an information processing part and often a physical processing part. A holon can form part of another holon.

Autonomy: The capability of an entity to create and control the execution of its own plans and/or strategies.

Cooperation: A process whereby a set of entities develops mutually acceptable plans and executes them.

Holarchy: A system of holons that can cooperate to achieve a goal or objective. The holarchy defines the basic rules for cooperation of he holons and thereby limits their autonomy.

Holonic Manufacturing Systems (HMS): A holarchy that integrates the entire range of manufacturing activities from order booking through design, production and marketing to realize the agile manufacturing enterprise.

3. Motivation

Vanderbilt University is as a member of the Holonic Manufacturing Systems (HMS) project. We are participating in Holonic Resource Management Systems (HoRMS) which is one of the seven workpackages in the HMS project. HoRMS will address problems and decision making associated with complex management and control of resources used for achieving business targets. The designs will allow for resource-based reactions to changes in requirements or unforeseen events in order to maintain and improve system-wide performance. Holons make independent decisions on the use of resources, e.g. people, machines, material, energy, and time by negotiating with other holons. This workpackage focuses on discrete manufacturing. We use the Induction Motor Production Plant to illustrate the performance of the concepts and will be used in our pilot implementation of an intelligent distributed scheduler.

4. Induction Motor Production System

Induction motors exemplify a low-volume, high-variety production system and highlight many problems that arise from the inflexibility of centralized management system architectures. The system covers the parts machining and assembly processes. A schematic of the production process is shown in Figure 1. The input raw materials include the bearing bracket (BB), the main bearing (BRG), the frame, copper wire, steel sheets, and steel rods. The production system is made up of the following machine types: casting machines, boring machines, machining centers, fraise machines, clank Presses, lathe machines, and grinding machines. The machining processes are need to create the following parts: bearing bracket (BB), bearing (BRG), stator frame (SS), stator core, slot insulator, shaft (SFT), rotor core, rotor conductor, fan cover, and fan, which are then assembled to generate the final product.

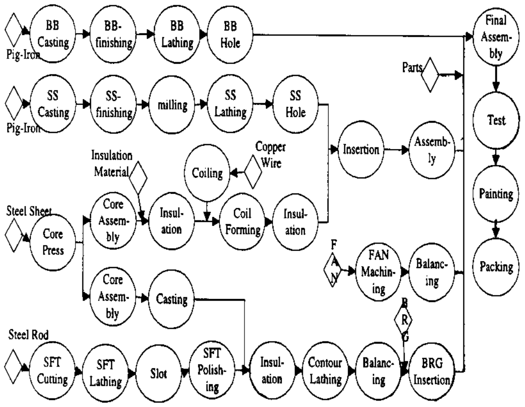

Figure 1. Motor Production Processes.

Proceedings of SPIE, "Architectures, Networks, and Intelligent Systems for Manufacturing Integration", Pittsburgh, Pennsylvania, 15-16 October 1997, pp. 108-115.

In general, a motor plant generates 800 different types of motor, with an order mix of 200 different types of motor per day and 1,000 motor components per day. Lot sizes vary from 1 to 100, however, the average lot size is 5.

5. The Testbed

As a first step, we are building an initiate testbed that focuses on the shaft manufacturing section. There are three products that are generated from this testbed:

- P-D24 corresponds to an induction motor shaft with diameter 24 mm and length 327 mm. This shaft is for 1.5 kw induction motors.
- P-SD42 corresponds to an induction motor shaft with diameter 42 mm and length 603 mm. This shaft is for 11 kw induction motors.
- P-SD48 corresponds to an induction motor shaft with diameter 48 mm and length 676 mm. This shaft is for 22 kw induction motors.

Five machine-types:

- M-1 is a Turning center machine.
- M-2 is a Grinding machine.
- M-3 is a NC lathe machine.
- M-4 is a NC fraise machine.
- M-5 is a Machining center machine.

are needed for the manufacturing process, and the number of each machine type in the testbed can be specified by the user. There are three alternate production methods shown in Figure 2. The first production method, the raw material, steel rod, goes to machine type M-1, and is then transferred to machine type M-2. The second production method starts with machine type M-3, then transfers to machine type M-4, and finished at machine type M-2. The third method also starts at machine type M-3, then goes to machine type M-5, and finishes at machine type M-2.

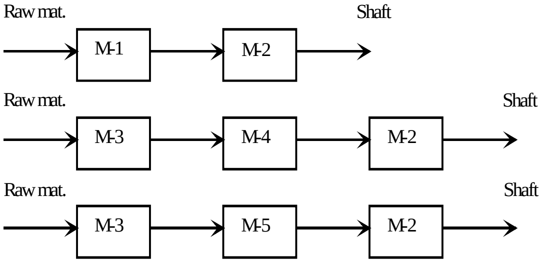

Figure 2. Production Methods

5.1 System Architecture

The heterarchical architecture was selected for our testbed. Completely eliminates the master/slave relationship among the entities comprising the control system. The result is an flat architecture where modules cooperate as equals, rather than being assigned subordinate and supervisory relationships. Communication and control among modules, to resolve production tasks, is achieved via message-passing schemes. This provides the necessary flexibility and robustness. When a module fails, another module can take over its tasks. In spite of this flexibility, careless implementation of the heterarchical control may produce some sort of cooperative anarchy (Duffie & Prabhu 1994). Systematic implementation provides several advantages over traditional hierarchical control (Baker & Merchant 1993; Duffie et al. 1988; Duffie 1990; Duffie & Prabhu 1994):

- higher degree of distribution, modularity, and maintainability,
- higher robustness and fault tolerance, and
- implicit modifiability and reconfigurability.

5.2 Holons

There are five basic holons defined for this testbed:

- **Product Holon**, which corresponds to a product to be manufactured. For this testbed, they are P-SD24, P-SD42, and P-SD48.
- **Machine Holon**, which corresponds to a machine in the testbed. The five machine types are M-1, M-2, M-3, M-4, and M-5.
- **Scheduler Holon**, which performs the decision making, product reservation, and resource allocation functions.
- **Computing Holon**, which calculates the costs and time for a machine to perform a particular machining or assembly tasks.
- **Negotiation Holon**, which handles negotiation processes between part and machine holons.

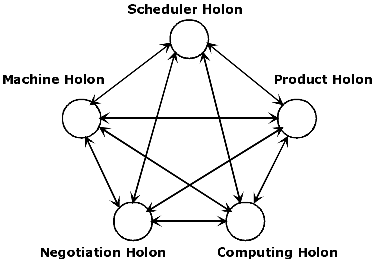

Figure 3. Basic Holons

6. The Computational Components

There are three primary computational components in the testbed:

1. Order generation scheme, which is a stochastic process.
2. The Bidding Production Reservation Scheduling (BPRS) methodology which uses a Production Reservation (PR) scheme with a Contract-Net, and
3. Data collection for performance studies.

6.1 Order generation

The testbed uses a stochastic algorithm to determine which of P-SD24, P-SD42, and P-SD48 is the next product to be manufactured. The interarrival time between orders can be modeled by one of the three probability distributions: Normal, Exponential, and Uniform. Processing time also have been modeled using the above three distributions. This conforms with experiments reported in related research (Huq 1994, Blackstone, Phillips, & Hogg 1982; Pedgen, Shannon, & Sadowski 1990; Wein & Chevalier 1992), and also to situations observed in practice.

6.2 Bidding Production Reservation Scheduling (BPRS)

There are two major parts in this component: Production Reservation Scheduling (PRS) and Contract-Net Protocol (CNP). The links between Production Reservation Scheduling and Contract-Net Protocol are presented in the interaction among holons section.

6.2.1 Production Reservation Scheduling (PRS)

This approach for manufacturing scheduling has been motivated by airline reservation systems (Conway & Maxwell 1998). A Production Reservation Scheduler (PRS) tackles a scheduling problem from a different perspective to avoid its computational intractability. In this approach, each job is scheduled, by a centralized scheduler, as it arrives into the system. In addition to breaking down the complexity of the scheduling problem, such an approach is a better match for realistic situations where jobs usually come one at a time to the system. Since the schedule of a new job is determined based on current work loads, times-in-progress, and work-in-progress inventories, it should be possible to reduce both lead-times and tardiness of individual jobs while determining their machine sequences. The benefits of the original proposal for this approach has been recently substantiated by an industrial case-study (Baker 1991;1992). In this case-study, using a PRS resulted in 47% work-in-progress inventory reductions, 59% average time-in-progress reductions, 3% total-work-content reductions, as well as substantial tardiness reductions. Moreover, since the developed PRS was market-driven, i.e., the costs of the various manufacturing activities were incorporated into the planning and scheduling activities, production cost was cut by 3%. The PRS implemented for this case-study utilized the Contract-Net concept, originally proposed in the Distributed Artificial Intelligence (DAI) community. Hence, this concept and its relevance to the decentralized scheduling problem are presented next.

6.2.2 The Contract-Net Protocol (CNP)

Researchers and practitioners in the field of Distributed Artificial Intelligence (DAI) have been studying problems where a group of agents interact to solve a problem (Demazeau & Muller 1990; Werner & Demazeau 1992). Distributed planning is an example of such a problem where a group of agents try to develop a plan collectively in the absence of a central coordination mechanism.

The Contract-Net (Smith 1978) is one of the techniques that resulted from research in DAI to solve the problem of allocation of the tasks in a decentralized system. It defines a bidding approach that enables task allocation among multiple agents (Parunak 1987; Smith 1978). According to the Contract-Net paradigm, software agents negotiate task allocation via a bidding mechanism. Using such a scheme, available agents compete against each other for the right to perform a given task. The bid submitted by each agent reflects the agent’s capability for performing the task. The better the bid, the more capable is the agent for performing the task. The bid depends on the agent’s local criteria and assessment of its own capabilities for achieving the task. Once bids are compared, the task is offered to the agent with the best bid, and this process continues for the next task, and so on. This concept was originally put into practice for an air traffic control application (Smith 1978). In this case, the application was a distributed sensor system whose goal was to track airplanes. The Contract-Net was used to determine which radar station, in a collection of radar stations, had the most accurate knowledge about an airplane’s location.

In a manufacturing domain, the Contract-Net protocol has first been used for task distribution among a hierarchically organized set of manufacturing entities in the YAMS (Yet Another Manufacturing System) architecture (Parunak 1988; 1987). It has also been employed to test rescheduling effectiveness in a decentralized job shop environment (Tsukada & Shin 1994) after the original schedule was generated by a centralized scheduler (Bean 1991). Moreover, the Contract-Net Protocol has proved to be very beneficial and applicable to heterarchically-controlled manufacturing systems (Baker 1988; Lin & Solberg 1993). To realize decentralized control in a manufacturing environment, the machines comprising the manufacturing system and the parts to manufactured need to be modeled as autonomous agents (Lin & Solberg 1993). These agents use the Contract-Net Protocol to implement a bidding scheme that allows the parts to dynamically and cooperatively determine, given the set of machines, each part’s production schedule (Baker 1992).

Sandholm developed the Transportation Cooperation Net (TRACONET) system which based on Contract-Net scheme. This system consists of a number of geographically dispersed dispatch centers of different companies. Each center is responsible for the deliveries initiated by certain factories and has a certain number of vehicles to take care of the deliveries. The local problem of each agent is a heterogeneous fleet multi-depot routing problem. The objective is to minimize the transportation costs. In the negotiations, the agents exchange sets of deliveries whenever this is profitable. The negotiations can be viewed as an iterative way of enchanting the global routing solution by traversing a sequence of task allocations among agents. With this iterative task reallocation scheme, a global solution closer to the global optimum is reached although no global optimization run is performed (Sandholm 1993).

A. Saad, K. Kawamura, and G. Biswas proposed a heterarchical scheduling approach for the flexible manufacturing systems. The approach adopts a multiagent cooperative problem solving paradigm by using a bidding mechanism based on the Contract Net Protocol to generate the production plan and schedule. Overall, scheduling is implemented using a Production Reservation approach. A typical job shop testbed demonstrates the effectiveness of this approach by comparing its performance to heuristic dispatching rules. Production Reservation augmented with the heuristic dispatching rules has also been shown to be even more effective, and should prove especially valuable in large variety discrete part and assembly manufacturing processes (A. Saad, K. Kawamura, and G. Biswas 1997).

6.2.3 The Interaction Among Holons

The interaction among holons during negotiation process can be divided into five processes: Announcing, Calculating Cost, Bidding, Decision Making, and Awarding and Resource Allocation.

When the new product enters to the system, the information about this product is passed to the Product Holon. This information consists of its due date, priority and the different production methods that can be applied to manufacture the product. As mentioned above, there are three alternate production methods for this system. The Product Holon, that has the specific due date, priority and all alternate production methods, passes all information to the Negotiation Holon. In the Announcing process, the Negotiation Holon sends request to bid messages containing the specific due date, priority, and the first production method to all Machine Holons if considers appropriate for the particular task.. The set of Machine Holons chosen then pass the received information to the Computing Holon for calculating costs. The Computing Holon calculates the cost and time of the first production method for each Machine Holon. The cost and time depend on the current environment for each Machine Holon. This cost, C, is a linear function and is based on the specific due date and priority:

$$C = W1 * (\text{due-date} - \text{now}) + W2 * (\text{priority});$$

where W1 and W2 are weighting factors (real number) and can be varied depending on product type.

Moreover, the time that each Machine Holon will require to finish the specific operations will be calculated by the Computing Holon. Both cost and time information will be sent back to each Machine Holon that intends to make a bid. During the Bidding process, the Machine Holons make bids by sending calculated cost and time back to the Negotiation Holon. Since there are three alternate production methods, the cost and time for each production method are not the same. The above processes are repeated for the other two production methods. Then all the calculated costs and times for each production method is compared so the minimum cost and time that correspond to the one of the production methods can be selected. In the Awarding and Resource Allocation stage, the Negotiation Holon will select the proper production method that gives the minimum cost and time and send selected production method and related information to the Scheduler Holon. The Scheduler Holon now starts the Production Reservation Scheduling (PRS) and resource allocation process by sending the confirmation messages to the original Product Holon and all Machine Holons that are selected. All the above processes will be repeated when the new Product Holon, that is associated with the next product, enters to the system. Figure 4 depicted the interaction among holons.

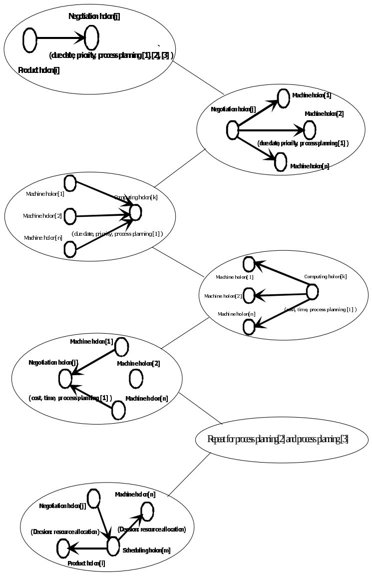

Figure 4. Interaction Among Holons

6.3 Data Collection

Performance studies of the Bidding Production Reservation Scheduling (BPRS) scheme will be conducted by developing a simulation scheme implemented in Java language. Two dispatching rules will be used for comparative. Early Due Date (EDD), where parts/products in the input buffer are sorted in the order of their due dates, and Shortest Processing Time (SPT), where parts/products with smallest processing time is picked for processing from the machine’s input buffer, will be used for comparative performance studies. The Augmented Production Reservation scheme will be developed by combining the BPRS scheme with the local dispatching rules, EDD and SPT. The difference between BPRS and Augmented BPRS is in terms of when products are released to the input buffer of the machine. In case of BPRS, parts are held back till they are ready for processing and the allocated turns for jobs are not changed. In the Augmented BPRS, order of jobs in a queue may be changed based on the EDD or SPT dispatching rules. In order to validate these algorithms and study their performance, many statistical performance parameters will be measured during simulation. These parameters are Lead time, Flow time, Production time, Maximum WIP (Work-In-Progress), and Percentage of tardy.

7. Conclusions and Future Work

The developed planning and scheduling algorithms cannot directly apply to the actual manufacturing systems without testing their performance. A simulation testbed needs to be created and will be used to measure the performance of these algorithms. To achieve this objective, the functional characteristics of the testbed have to be carefully defined. These characteristics should reflect the functional characteristics of the actual manufacturing system, the Induction Motor Production System, which are presented in this paper.

In future work, the implementation of this testbed will be developed. There are many points that have to be consolidated during the implementation stage. To achieve the reusability and modifiability, The object-oriented technique should be applied when modeling this testbed. To reflect the actual manufacturing system, the real data from the Induction Motor Production System will be incorporated into all parameters of this testbed. To be more adaptable in the sense of platform independence, the Java language is selected to use as an implementation language. Finally, data collection and statistical analysis will be performed to indicate the performance of these algorithms.

References

Baker, A. D., and Merchant, M. E., “Automatic factories: How will they be controlled”, *IEEE Potentials*, V. 12, No. 4, pp. 15-20, 1993.

Bean, “Matchup scheduling with multiple resources, release dates and disruptions”, *Operations Research*, Vol 39, No. 3, pp. 470-483, 1991.

Biswas, G., Kawamura, K. and Saad, A., “Intelligent Manufacturing Systems: State-Of-The-Art”, *Proceedings of the 10th Annual Technical and Business Exhibition Symposium (TABES)*, Huntsville Association of Technical Societies (HATS), May 1994, pp. 94-503.

Blackstone, J. H.; Phillips, D. T.; and Hogg, G. L., “A state-of-the-art survey of dispatching rules for manufacturing job shop operations”, *International Journal of Production Research*, Vol. 20, No. 1, pp. 27-5, 1982.

Cheng, S. C.; Ramamritham, K.; and Stankovic, J. A., “Scheduling algorithms for hard real-time systems - a brief survey”, *IEEE software*, 1988.

Duffie, N. A.; Piper, R. S.; Humphrey, B. J.; and Hartwick, J. P., “Fault-tolerant heterarchical control of heterogeneous manufacturing system entities”, *Journal of Manufacturing Systems*, Vol. 7, No. 4, pp. 315-327, 1988.

Duffie, N. A., “Synthesis of heterarchical manufacturing systems”, *Computer Science in Industry, Special Issue: Josef Hatvany Memorial: Total Integration - Analysis and Synthesis*, Vol. 14, pp. 167-174, 1990.

Duffie, N. A., and Prabhu, V. V., “Real-time distributed scheduling of heterarchical manufacturing systems”, *Journal of Manufacturing Systems*, V. 13, No. 2, pp. 94-107, 1994.

Huq, Z., “A process oriented manufacturing system simulation to measure the effect of shop control factors”, *Journal of Simulation*, Vol. 62, No. 4, pp. 218-228, 1994.

James H. Christensen, “Holonic Manufacturing Systems: Initial Architecture and Standards Directions”, *First European Conference on Holonic Manufacturing Systems*, Hannover, Germany, 1994.

Koestler, A., “*The Ghost in the Machine*”, Arkana Books, 1989.

Lin Guo, Peter B. Luh and Yuji Kyoya, “Holonic Manufacturing Scheduling: Architecture, Cooperation Mechanism, and Implementation”, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Tokyo, Japan, 1997.

Parunak, H. V. D., “Characterizing the manufacturing scheduling problem”, *Journal of Manufacturing Systems*, Vol. 10, No. 3, pp. 241-259, 1991.

Pedgen, C.; Shannon, R.; and Sadowski, R. P., “*Introduction to Simulation Using Siman*”, Hightstown, NJ: McGraw-Hill, Inc., 1990.

Smith, R. G., “A Framework for Problem Solving in a Distributed Processing Environment”, Standford University, CA., *Computer Science Department Report STAN-CS-78-700*, 1978.

Smith, R. G., “The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver”, *IEEE Transactions on Computer*, *C-29*, No. 12, pp. 1104-1113, 1980.

S., Tamura, “Testbed J1: Induction Motor Production Scheduling”, *HoRMS TESTBED sub-project, Work Package Plan: Holonic Management Systems*, 1996.

`Saad, A., Kawamura, K., and Biswas, G., “Performance Evaluation of Contract Net-Based Heterarchical Scheduling for Flexible Manufacturing Systems”, Special Issues on Intelligent Manufacturing Planning and Shop Floor Control, *International Journal of Intelligent Automation and Soft Computing*, 1997.

Proceedings of SPIE, "Architectures, Networks, and Intelligent Systems for Manufacturing Integration", Pittsburgh, Pennsylvania, 15-16 October 1997, pp. 108-115.

Tsukada, T., and Shin, K. G., "Polite rescheduling: Responding to local schedul disruptions in distributed manufacturing systems", *In Proceeding of the IEEE International Conference on Robotics ans Automation*, pp. 1986-1991, 1994.

Tuomas Sandholm, "An Implementation of the Contract Net Protocol Based on Marginal Cost", *Eleventh National Conference on Artificial Intelligent (AAAI-93)*, Washington DC., pp. 256-262.

Wein, L. M., and Chevalier, P. B., "A broader view of the job-shop scheduling algorithm", *Management Science*, Vol. 38, No. 7, pp. 1018-1033, 1992.