FROM MANUFACTURING SCHEDULING TO SUPPLY CHAIN COORDINATION: THE CONTROL OF COMPLEXITY AND UNCERTAINTY

Peter B. LUH **Weidong FENG**

Department of Electrical & Computer Engineering
University of Connecticut, Storrs, Connecticut, 06269-2157, USA
Luh@engr.uconn.edu *Feng@engr.uconn.edu*

Abstract

With time-based competition and rapid technology advancements, effective manufacturing scheduling and supply chain coordination are critical to quickly respond to changing market conditions. These problems, however, are difficult in view of inherent complexity and various uncertainties involved. Based on a series of results by the authors, decomposition and coordination by using Lagrangian relaxation is identified in this paper as an effective way to control complexity and uncertainty. A manufacturing scheduling problem is first formulated within the job shop context with uncertain order arrivals, processing times, due dates, and part priorities as a separable optimization problem. A solution methodology that combines Lagrangian relaxation, stochastic dynamic programming, and heuristics is developed. Method improvements to effectively solve large problems are also highlighted. To extend manufacturing scheduling within a factory to coordinate autonomic members across chains of suppliers, a decentralized supply chain model is established in the second half of this paper. By relaxing cross-member constraints, the model is decomposed into member-wise subproblems, and a nested optimization structure is developed based on the job shop scheduling results. Coordination is performed through the iterative updating of cross-member prices without accessing other members' private information or intruding their decision-making authorities, either with or without a coordinator. Two examples are presented to demonstrate the effectiveness of the method. Future prospects to overcome problem inseparability and improve computing efficiency are then discussed.

Keywords: Manufacturing scheduling, supply chain coordination, complexity and uncertainty, decomposition and coordination, Lagrangian relaxation.

1. Introduction

1.1 Motivation

With time-based competition and rapid technology advancements, manufacturing scheduling has been recognized by industrial practitioners as an invaluable tool that can significantly improve on-time delivery, reduce inventory, cut lead time, improve resource utilization, and increase system throughput.

Scheduling is especially important for multi-product and small-lot production, where many products are required to be manufactured in small quantities to meet the diverse and time-varying demand. At the same time, as companies are increasingly relying on their business partners or suppliers, manufacturing scheduling within a factory has to be extended to coordinate members across chains of suppliers. Manufacturing scheduling and supply chain coordination, however, are extremely difficult because of inherent complexity and various uncertainties involved. The problems are large and are generally NP-hard in view of the combinatorial nature of integer or mixed-integer optimization. Uncertainties include the arrivals of orders, order processing requirements, and resource capacities. The problems are further complicated for supply chains in view of member autonomy, where members have private information and decentralized decision-making authority. How to effectively schedule and coordinate activities within and across chains of suppliers without accessing other members' private information or intruding their decision-making authorities is becoming a major challenge.

1.2 Literature Review

Given the economical and logistical importance of scheduling problems, many of the early efforts centered on obtaining optimal solutions. Two prominent methods are branch and bound (Fisher, 1973) and dynamic programming (Potts and Van Wassenhove, 1987; Pinedo, 1995). Since most practical scheduling problems are NP-hard (Garey and Johnson, 1979), production schedules are also generated by experienced shop-floor personnel using dispatching rules in practice. Various heuristics have been developed based on due dates, criticality of operations, processing times, and machine utilization (e.g., Blackstone, Phillips, and Hogg, 1982). Many artificial intelligence approaches also use heuristics for scheduling (e.g., Kuziak, 1990). These heuristics are computationally efficient; however, it is difficult to evaluate the quality of solutions. Also, most heuristics do not provide effective ways to iteratively improve solutions. Attempts to bridge the gap between heuristic and optimization approaches have also been undertaken, including the combination of Lagrangian relaxation and heuristics (Luh, et al., 1990). Dynamic programming was embedded within Lagrangian relaxation in Chen, Chu, and Proth, 1998 and in Wang, et al., 1997.

Most of the above efforts focus on deterministic scheduling. For stochastic scheduling, many results are on the sequencing of parts under simplifying configurations by minimizing an expected objective function (e.g., Soroush, 1996). Not many results, however, have been obtained for complicated problems of more than two types of machines, as the problems are considerably harder (Pinedo, 1995). Scheduling problems have also been considered within the queuing framework where orders arrive randomly with random processing times. Results in this area mostly concentrate on performance analysis of simple scheduling policies, rather than generating optimal or near-optimal schedules (e.g., Kumar and Meyn, 1995). Heuristics have also been combined with probabilistic or fuzzy theory (e.g., Custodio et, al., 1994), however, they suffering from the difficulties of heuristic methods as presented in the previous paragraph.

As companies are increasingly relying on their business partners or suppliers, coordination across chains of suppliers is becoming critical. Most of scheduling results, however, cannot be easily extended to supply chain coordination in view of organization autonomy. Existing results on supply chain coordination focus at the strategic level, with limited consideration on operational complexity and uncertainty. The results include inventory control, contracting strategy, and quantity discounts. Coordination from the supplier's view was investigated in Monahan (1984), and it was shown that a supplier is able to increase its profit by offering quantity discounts to distributor. Incentive coordination was studied by Lee and Whang (1999) and Chen (1999) within the context of inventory control. The effects of contract parameters such as quantity commitments (Anupindi and Bassok, 1999) and returns policies (Lariviere, 1999) were also analyzed.

At the operational level, supply chain coordination can roughly be classified into centralized and distributed models. Among centralized models, a deterministic mixed integer model was presented based on economic order quantity (EOQ) to develop a "global resource deployment policy" without considering uncertain orders or member autonomy (Cohen and Lee, 1989). A linear integer programming model was presented to minimize production, inventory, and transportation costs in Arntzen et al, (1995). A linear programming model for supply chain planning was developed for manufacturing networks (Gaonkar and Viswanadham, 2001). These models are generally large, and special attentions are needed on computational requirements. As for distributed models, agent-based architectures have been extensively studied. Members are represented by agents, and coordination is carried out through exchanging information and imposing constraints by using rule-based methods, e.g., following the "Contract Net Protocol" originally developed in Smith (1980) (e.g., Swaminathan, Smith, and Sadeh 1998). These agent-based methods are powerful for supply chain modeling, but are difficult to obtain optimized solutions.

1.3 Outline of the Paper

The rapid growth of information technology opens up new opportunities to link manufacturing scheduling and supply chain coordination. Scheduling and coordination utilizing Internet connectivity would enable members to synchronize information and product flow to optimize supply chain performance. In this paper, based on a series of results by the authors, decomposition and coordination by using Lagrangian relaxation is identified as an effective way for manufacturing scheduling and supply chain coordination to control complexity and uncertainties. In Section 2, a manufacturing scheduling problem is formulated within the job shop context as an integer optimization problem with uncertain order arrivals, processing times, due dates, and part priorities.

In view that the problem is NP-hard and computational requirements to obtain an optimal solution grow exponentially as the problem size increases, our goal, therefore, is not to obtain an optimal solution. Rather, we want to obtain effective solutions in a computationally efficient manner. Furthermore, in view of the presence of major uncertainties, it is clear that the traditional "static" schedules typically represented by Gantt charts are not adequate. We are therefore seeking *scheduling policies* describing what to do under which circumstances. A solution methodology that combines Lagrangian relaxation, stochastic dynamic programming, and heuristics is developed in Section 3. This decomposition and coordination approach and the resulting semi-closed loop scheduling policies effectively handle complexity and uncertainties with reasonable computational requirements. Method improvements to effectively solve large problems are also highlighted.

In Section 4, the above method is extended to supply chain coordination, where a decentralized supply chain model is established. The model is decomposed into member-wise subproblems by relaxing cross-member constraints, and a nested optimization structure is developed. The subproblems are similar to job shop scheduling problems of Section 2, and can be effectively solved by individual members. Coordination is performed through the iterative updating of cross-member prices without accessing other members' private information or intruding their decision-making authorities, either with or without a coordinator. An example of coordination with a coordinator is presented in Section 5 within the context of assets overhaul and repair service chain. An example of distributed and asynchronous coordination without a coordinator is presented in Section 6. Finally, Section 7 summarizes the paper, and presents future prospects.

2. Formulation

In this section, manufacturing scheduling will be formulated within the job shop context as an integer optimization problem, where a job shop is a typical setting for manufacturing low-volume high-variety products. The time horizon is divided into K discrete time intervals indexed by k, $0 \le k \le K-1$. There are H machine types, and the number of type h machines ($1 \le h \le H$) available at time k is given and denoted as M_{kh}. There are I parts to be scheduled, and part i ($1 \le i \le I$) has its arrival time a_i, due date d_i, and priority (weight) w_i. Part i is assumed to require a series of J_i operations for completion, and operation j ($1 \le j \le J_i$) of part i is denoted as (i, j). The first operation of part i, $(i, 1)$, can only be started after the arrival of the order or appropriate materials. Operation (i, j) has to be performed on a machine of type h belonging to a given set of "eligible" machine types H_{ij} for a duration of time t_{ijh}, and the processing may start only after its immediate preceding operation has been completed. For a particular part, the arrival time a_i, processing times $\{t_{ijh}\}$, due date d_i, and priority (weight) w_i may not be known exactly in advance. Such parameters are modeled as independent random variables with given discrete distributions. For simplicity, machine availability is assumed to be deterministic. The objective is to maximize on-time delivery of parts and to reduce work-in-process (WIP) inventory. The mathematical formulation is presented below with a balance between modeling accuracy and method complexity (for details see Wang, et al, 1997; Luh, Chen, and Thakur, 1999).

1) ***Arrival Time Constraints***: The first operation of part i cannot be started until the arrival of the order for the part or appropriate materials, i.e.,

$$a_i \le b_{i1},\ i = 1, ..., I, \tag{1}$$

where b_{i1} is the beginning time of operation $(i, 1)$.

2) ***Operation Precedence Constraints***: Operation j+1 of part i cannot be started before the completion of operation j plus a "time-out" S_{ij} between the two operations, i.e.,

$$c_{ij} + S_{ij} + 1 \le b_{i,j+1},\ i = 1,...,I;\ \ j = 1,...,J_i - 1, \tag{2}$$

where c_{ij} is the completion time of (i, j), and $b_{i,j+1}$ the beginning time of $(i, j+1)$.

3) ***Processing Time Requirements***: Operation j of part i must be assigned to a machine of type $h \in H_{ij}$ with the required amount of processing time t_{ijh}, i.e.,

$$c_{ij} = b_{ij} + t_{ijh} - 1,\ i = 1,...,I;\ \ j = 1,...,J_i\ ;\ h \in H_{ij}. \tag{3}$$

The "+1" and "–1" are needed in (2) and (3), respectively, since a beginning time is assumed to be the beginning of a period and a completion time the end of a period.

Part i may have uncertainty arrival time, processing times, due date, and priority; and constraints (1), (2), and (3) are required to be satisfied for each possible realization of random events to accurately model the uncertainties. Different realizations of random parameters may lead to different beginning times, and $\{b_{ij}\}$ are thus random also.

4) ***Machine Capacity Constraints***: The number of operations assigned to machines must be less than or equal to machine capacity per type at each time period, i.e.,

$$\sum_{ij} \delta_{ijkh} \le M_{kh},\ k = 0,...,K-1; h \in H\ , \tag{4}$$

where δ_{ijkh} is a 0-1 operation variable equal to 1 if (i, j) is assigned to a machine of type h at k, and 0 otherwise.

With uncertain order arrivals and processing, it is difficult for (4) to be satisfied for all possible realizations of random events. They are thus approximated in the expected sense for reduced method complexity, i.e.,

$$E\left[\sum_{ij} \delta_{ijkh}\right] \le M_{kh},\ k = 0,...,K-1; h \in H\ . \tag{5}$$

5) ***Objective Function***: The goals of on-time deliveries of parts and low work-in-process inventory are modeled by penalties on late deliveries and on releasing orders too early:

$$J \equiv E\left[\sum_i (w_i T_i^2 + \beta_i E_i)\right], \tag{6}$$

where tardiness T_i of part i is the amount of overdue time, i.e., max (0, c_{iJ_i} $- d_i$). Let $\bar{b}_{i1}$ be the desired beginning time of the first operation, then earliness E_i is the amount of early start time, i.e., max(0, $\bar{b}_{i1} - b_{i1}$). Parameters w_i and β_i are, respectively, weights associated with tardiness and earliness penalties. The square on tardiness reflects that a part becomes more critical with each time unit passing its due date. The problem is to minimize (6) through selecting appropriate machine types and beginning times, subject to constraints (1), (2), (3), and (5). Note that in view of uncertainties, we are not looking for a "static" schedule. Rather, we are looking for a "scheduling policy" indicating what to do under various realizations of random events. A solution is "model feasible" if it satisfies (1), (2), (3) and (5); and a solution is "implementable" if it can be practically implemented for all possible realizations of random events, i.e., it satisfies (1), (2), (3) and (4).

In the above formulation, (1), (2) and (3) are related to individual parts, and (5) couples various parts together. In view that both (5) and the objective function (6) are part-wise additive, the formulation is "separable." A solution methodology that synergistically combines Lagrangian relaxation, stochastic dynamic programming, and heuristics is thus developed to manage uncertainties and to provide near-optimal solutions with quantifiable quality[1].

3. Solution Methodology

In the method, expected machine capacity constraints that couple parts together are first "relaxed" by using "soft prices" or Lagrange multipliers. In view of the separability of our formulation, the relaxed problem can be decomposed into smaller subproblems, one for each part to be solved by stochastic dynamic programming. The close-loop nature of stochastic dynamic programming is fully exploited so that (1), (2), and (3) are exactly satisfied for each possible realization of random events. The prices or multipliers are then iteratively adjusted based on the degrees of capacity constraint violation to maximize the "dual function" following the market economy concept, i.e., increasing prices for over-utilized time slots and reducing prices for under-utilized ones. At the end of such iterations, simple heuristics are used to dispatch operations on machines as machines become available. The key steps are as follows.

3.1 The Lagrangian Relaxation Framework

Expected machine capacity constraints (5) are first relaxed by using Lagrangian multipliers $\{\pi_{kh}\}$, and the relaxed problem is obtained as:

$$\min_{\{b_{ij},h_{ij}\}} L, \text{ with}$$

$$\begin{aligned} L \equiv\ & E\left[\sum_i (w_i T_i^2 + \beta_i E_i)\right] + \sum_{kh} \pi_{kh}\left(E\left[\sum_{ij} \delta_{ijkh}\right] - M_{kh}\right) \\ =\ & E\left[\sum_i (w_i T_i^2 + \beta_i E_i) + \sum_i \sum_{jkh} \pi_{kh}\delta_{ijkh}\right] - \sum_{kh} \pi_{kh} M_{kh} \end{aligned} \tag{7}$$

subject to (1), (2), and (3). This relaxed problem can be naturally decomposed into the following part-wise subproblems:

$$\min_{\{b_{ij},hij\}} L_i, \text{ with } L_i \equiv E\left[w_i T_i^2 + \beta_i E_i + \sum_{j=1}^{J_i} \sum_{k=b_{ij}}^{c_{ij}} \pi_{kh}\right] \tag{8}$$

[1] The method can also be used to solve deterministic problems without taking expectation, and by using dynamic programming instead of stochastic dynamic programming (Zhang, et al., 2001).

subject to (1), (2), and (3).

Let L_i^* denote the resulting minimal subproblem cost. The high level dual problem is then obtained as:

$$\max_{\{\pi_{kh}\}} q, \text{ with } q \equiv \sum_i L_i^* - \sum_{kh} \pi_{kh} M_{kh}. \quad (9)$$

The subproblems are effectively solved by using stochastic dynamical programming as presented next.

3.2 Stochastic Dynamic Programming for Subproblems

Backward stochastic dynamic programming (DP) is used to solve part subproblems (8) and to manage uncertainties. In this process, DP stages are operations, and at each stage, states are possible operation beginning times. To be more specific, the terminal cost is given by

$$V_{iJ_i}(b_{iJ_i}, h_{iJ_i}) = E\left[w_i T_i^2 + \sum_{k=b_{iJ_i}}^{c_{iJ_i}} \pi_{kh_{iJ_i}} \right]. \quad (10)$$

The expectation is taken with respect to possible processing times of the last operation, due dates, and weights. The recursive DP equation is

$$V_{ij}(b_{ij}, h_{ij}) = E\left[\beta_i E_i \Delta_{ij} + \sum_{k=b_{ij}}^{c_{ij}} \pi_{kh_{ij}} + \min_{\{b_{i,j+1}, h_{i,j+1}\}} V_{i,j+1}(b_{i,j+1}, h_{i,j+1}) \right],$$
$$j = 1, \ldots, J_i - 1 \quad (11)$$

subject to (1), (2), and (3). In the above, $\Delta_{ij} = 1$ if $j = 1$, and $\Delta_{ij} = 0$ otherwise; and expectation is taken with respect to all possible processing times. Finally

$$L_i^* \equiv \min_{b_{i1}, h_{i1}} E[V_{i1}(b_{i1}, h_{i1})], \quad (12)$$

This minimization is subject to (1), and expectation is taken with respect to all possible arrival times.

In view that the solution obtained from stochastic dynamic programming is a policy, i.e., a closed-loop solution describing what to do under which circumstance, the above procedure can effectively manage uncertainties. The complexity is only slightly higher than that for the corresponding deterministic DP.

3.3 Multipliers Updating and Heuristics

Multipliers are updated based on the degrees of constraint violation. In view that integer variables are involved, the dual problem in (9) is polyhedron concave and non-differentiable. It is thus maximized by using a subgradient method (Bertsekas, 1999):

$$\pi^{n+1} = \pi^n + s^n g^n, \quad (13)$$

where n is the iteration index, s^n the step size, and components of the subgradient g^n are obtained after *all* the subproblems are solved:

$$g_{hk}(\pi^n) = E\left[\sum_{ij} \delta_{ijhk}\right] - M_{hk}. \quad (14)$$

The step size s^n is determined by:

$$s^n = \gamma^n (q^* - q^n) / \| g^n \|^2, 0 < \gamma^n < 2, \quad (15)$$

where q^* is the optimal dual value, and q^n the dual value obtained at iteration n. Since q^* is generally unavailable, estimate of q^* is used instead of q^*. This iterative procedure repeats until some stopping criteria are satisfied.

Since expected machine capacity constraints (5) have been relaxed, subproblem solutions, when put together, generally do not provide a model feasible schedule, i.e., (5) might be violated at certain time slots. Furthermore, a model feasible schedule satisfying (5) is generally not implementable, i.e., (4) might be violated. To obtain an implementable schedule, a list scheduling heuristic can be used based on a selected dual solution and the realization of random events; or stochastic dynamic programming results can be directly used for online dispatching. The overall solution therefore consists of open-loop multipliers (to be updated periodically, e.g., at the beginning of a shift) and closed-loop stochastic dynamic programming policies, and is therefore "semi-closed loop."

The above method has been effectively used for scheduling to control complexity and uncertainties. For deterministic problems, numerical testing of large problems was reported in Zhang, et al., 2001. Based on sample data sets from Toshiba's gas insulted switchgear factory, a total of 2,000 products for a total of 20,000 operations were scheduled on 24 cells each with 30 units of a key resource. Results show that the average duality gap reaches 8.25% in 10 minutes on a Pentium III 500 MHz PC, implying that near-optimal schedules are obtained within a reasonable amount of computational time. For stochastic problems, 38 parts with a total of 130 operations were scheduled on 35 machines belonging to 14 machine types over a planning horizon of 90 time units. Testing results supported by simulation demonstrate that near-optimal schedules can be efficiently obtained within a reasonable amount of CPU time (Luh, Chen, and Thakur, 1999).

3.4 Method Improvements

For large problems with many parts each having multiple operations, solving all the subproblems to obtain a subgradient (14) is time consuming. A surrogate optimization framework has been developed where a proper "surrogate subgradient" to update multipliers can be obtained without solving all the subproblems (Kaskavelis and Caramanis, 1998; Zhao, Luh, and Wang, 1999). In fact, only approximate optimization of one subproblem is necessary to obtain a good updating direction, thereby saving much computation time. The step-sizing rule (15) also plays an important role on convergence. In view that q^* is unknown, the "variable target value scheme" of Kim, Ahn, and Cho (1991) was extended to the context of surrogate optimization framework, where q^* is approximated by a dynamically adjusted "variable target value" (Chen, Luh and Fang, 2001). To generate good schedules, a time window-based heuristic that combines the Shortest Processing Time rule with the Critical Ratio rule has also been developed (Chen, Luh and Fang, 2001). Testing results show that the improvement is significant for problems having a large number of parts with long chains of operations, or for problems with high levels of machine utilization.

In view that arrival time constraints, operation precedence constraints, and processing time requirements are satisfied for all possible realizations of random events by using stochastic dynamic programming, and uncertainties are effectively managed by the semi-closed loop solutions. The expected capacity constraints reduce the computational complexity without much loss of modeling accuracy and scheduling performance. In view that a dual value is a lower bound on the optimal cost, the quality of a feasible schedule can be quantitatively evaluated by comparing its cost with the maximum of the dual costs obtained. Furthermore, re-scheduling can be efficiently performed by initializing the multipliers at values obtained from the previous run, with drastically reduced computational requirements.

4. Supply Chain Coordination

As companies are increasingly relying on their business partners or suppliers, manufacturing scheduling within a factory has to be extended to coordinate members across chains of suppliers. Supply chain coordination, including scheduling and synchronization of activities within and across members, is the second focus of this paper. Two types of coordination will be investigated. For the first case, coordination among members is performed by an additional coordinator (Gou, Luh, and Y. Kyoya, 1998); while for the second case, coordination is performed in a decentralized way by individual members without a coordinator (Luh, et al., 2003a). Both face the challenge of member autonomy, and decomposition and coordination based on Lagrangian relaxation will be identified as an effective tool for both cases. In the following, a decentralized model will first be established building on the formulation of Section 2. The model will be decomposed into member-wise subproblems by relaxing cross-member constraints. These subproblems can be solved by using the method presented in Section 3. Coordination among members is then performed based on the pricing concept of market economy without accessing other members' private information or intruding their decision-making authorities.

4.1 Decentralized Modeling

Considering a supply chain with N members, and for member n ($1 \le n \le N$), there are I_n orders to process. Order i ($1 \le i \le I_n$) of member n is denoted as (n, i), and may consist of a set of operations specified by its process plan. It has its arrival time a_{ni}, due date d_{ni}, and priority (weight) w_{ni}. Two types of constraints are considered. Internal constraints are internal to each member, and include arrival time constraints, operation precedence constraints, processing time requirements, and machine capacity constraints as presented in Section 2. Cross-member constraints (n, i) to (n', i') of another member n'. Suppose that (n', i') is to be used as a component part for (n, i), then the "*cross-member precedence constraint*" is given as:

$$c_{n'i'} + S_{n'i'ni} + 1 \le b_{ni}, \qquad (16)$$

where $c_{n'i'}$ is the completion time of (n', i'), b_{ni} the beginning time of (n, i), and $S_{n'i'ni}$ the "time-out" in-between (e.g., transportation time). Such constraints couple members together, and are difficult to handle. Similar to machine capacity constraints, they are approximated by the following "*expected cross-member precedence constraint*:"

$$E[c_{n'i'} + S_{n'i'ni} + 1 - b_{ni}] \le 0. \qquad (17)$$

4.2 Objectives Function

Suppose that the goal of each member is described by (6) to minimize a weighted sum of expected tardiness and earliness penalties, i.e.,

$$J_n \equiv E\left[\sum_i (w_{ni}T_{ni}^2 + \beta_{ni}E_{ni})\right]. \qquad (18)$$

It is further assumed that the overall objective of the chain is to minimize the sum of individual members' objectives:

$$J \equiv \sum_{n=1}^{N} J_n . \qquad (19)$$

4.3 Solution Methodology

In view those cross-member constraints (17) are linear and the objective J (19) is a sum of individual members' objectives, the formulation is separable, and Lagrangian relaxation can be effectively applied. By relaxing cross-member constraints (17) using Lagrangian multipliers $\eta_{n'i'ni}$, the following relaxed problem is obtained:

$$\min_{\{c_{nT}, b_{ni}\}} L,$$

$$\text{with } L \equiv \sum_{n=1}^{N} J_n + \sum_{n'=1}^{N} \sum_{i'=1}^{I_{n'}} \mathrm{E}\left[\eta_{n'i'ni}(c_{n'i'} - b_{ni})\right] \qquad (20)$$

With cross-member multipliers $\{\eta_{n'i'ni}\}$ given, (20) can be decomposed into member-wise subproblems:

$$\min_{\{b_{nij}, h_{nij}\}} L_n,$$

$$\text{with } L_n \equiv J_n + \mathrm{E}\left[\sum_i^{I_n} \eta_{nin''i''} c_{ni} - \sum_i^{I_n} \eta_{n'i'ni} b_{ni}\right], \qquad (21)$$

where b_{nij} denotes the beginning time of the j^{th} operation of order (n, i), and h_{nij} the machine type to process it. The second term on the right of (21) is for order (n, i) feeding downstream order (n'', i''), and the third term is for order (n, i) receiving upstream order (n', i'). Comparing with (6), it is clear that the objective of (21) is (6) modified by penalty terms depending linearly on beginning times of first operations, completion times of last operations. Consequently, these subproblems can be effectively solved by using the method presented in Section 3.

There are two price updating schemes leading to two types of coordination. One is coordination with a coordinator, where cross-member precedence prices are updated by the coordinator based on subproblem solutions obtained by individual members as shown in Figure 1. This process is a straightforward extension of the subgradient or surrogate subgradient method presented in Section 3.

For the case without a coordinator, each cross-member multiplier is updated in a distributed and asynchronous manner by members directly related to it as shown as in Figure 2. To avoid two organizations updating one price at the same time, a "token" is introduced for each multiplier, and only the member holding the token can update it. It is assumed that the *total asynchronism assumption* (Bertsekas and Tsitsiklis, 1989) holds, i.e., all prices are updated infinitely often, and old price information will eventually be purged from the chain. It is shown that updating can be performed without being trapped at a local maximum (Luh, et al., 2003a). The iterative process repeats until prices are converged or some stopping criteria are satisfied, and heuristics are then used to construct a feasible solution.

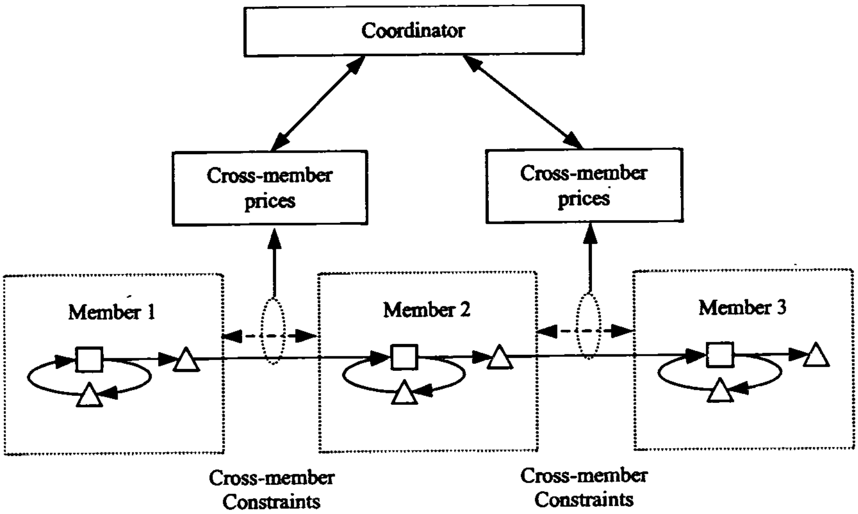

Figure 1 Coordination with a coordinator

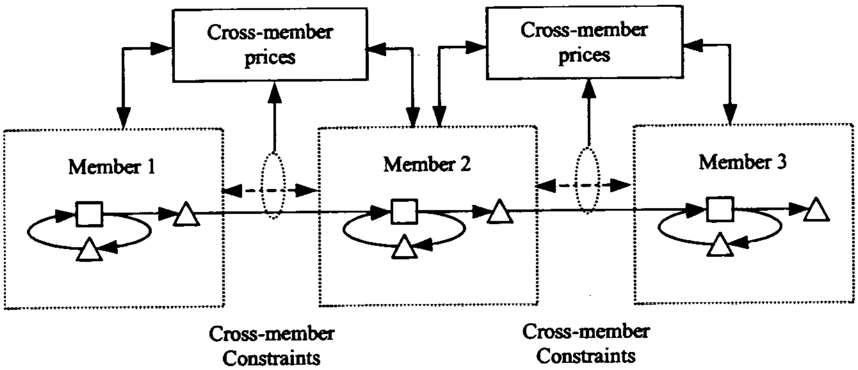

Figure 2 Coordination without a coordinator

With multipliers dynamically updated and schedules adjusted, the above decomposition and coordination framework can fulfill existing commitments while maintaining agility to take on new orders. The nested optimization structure is shown in Figure 3.

Note that cross-member multipliers $\{\eta_{n'i'ni}\}$ represent the sensitivity of the overall cost with respect to cross-member precedence violations. They are marginal values of a time unit for the early or late delivery of orders, or for the early or late receiving of component parts. These prices thus reflect the pressure on order delivery, and provide quantitative answers to the old question "time is money, but how much?" To illustrate the method, two examples are presented next.

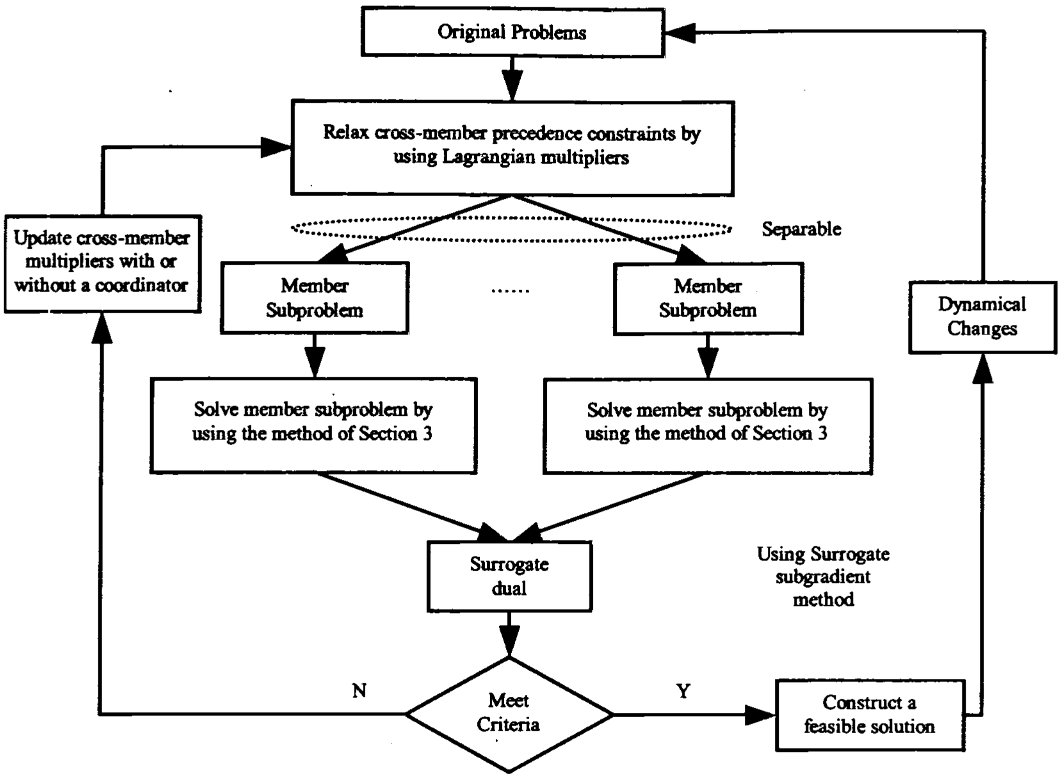

Figure 3 Nested structure for supply chain coordination

5. Assets Overhaul and Repair Services

5.1 Problem description

The first example is on the scheduling of an overhaul and repair service chain. Overhaul and repair services have been an important segment of the remanufacturing industry,[2] and are characterized by complicated processes and massive uncertainties, including (Guide, 2000):

- The uncertain quantity and timing of asset (e.g., jet engine) arrivals for overhaul and repair services;
- Time consuming and highly variable disassembly operations that disassemble an asset into modules and in turn into parts, and disassembly is often in conjunction with a "discovering" inspection process;
- The uncertain service scopes, which to a certain extent, is influenced by the inspection process;
- The existence of both serial-number-specific-parts (required to be assembled into the original assets they belonged to), and rotable parts (refurbished parts satisfying certain

[2] Remanufacturing is a process in which worn-out products are restored to like-new conditions through a series of disassembly, clean, refurbish, and assembly processes with the infusion of new parts as necessary in a factory environment.

qualifications for general use);

- Stochastic routings and highly variable processing times for repair operations.

To have short and predictable turn-around-times and low inventory, effective scheduling such services and managing inventory are imperative.

A schematic of the overhaul and repair services chain is presented in Figure 4. For simplicity, only one overhaul center and one repair shop are shown, although in reality there could be multiple overhaul centers and repair shops. Organizationally, each overhaul center is a member of the chain, and each repair shop is also a member of the chain.

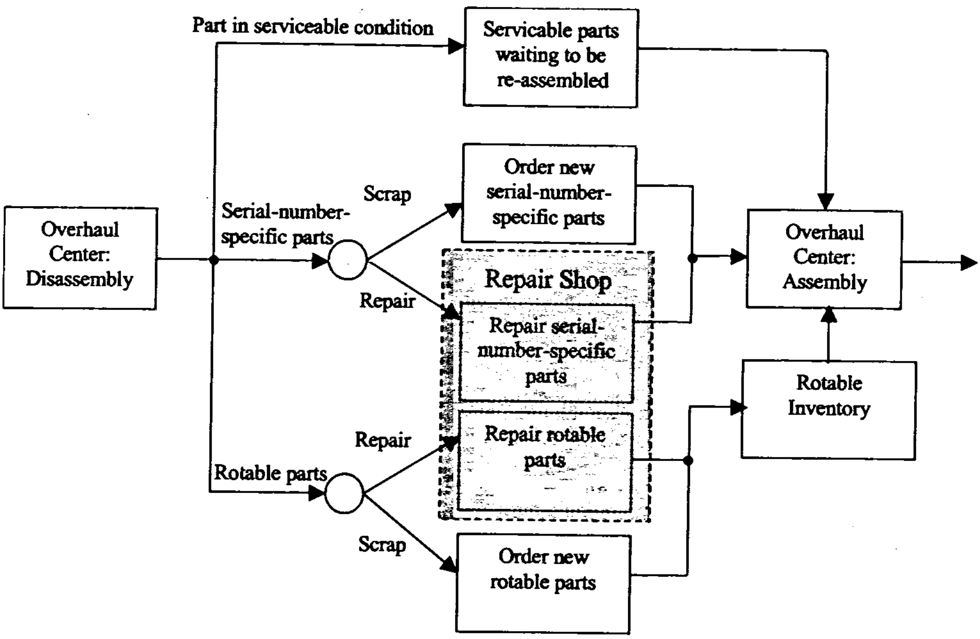

Figure 4 Schematic of the overhaul and repair services

Problem formulation is similar to what was presented in Section 4 with the addition of rotable inventory dynamics (Luh, et al., 2003b). The inventory level of rotable part type r at time k, $I_r(k)$, can be described by a flow balance equation. The level is increased by one when the last repair operation for such a part is completed. This is indicated by operation completion indicator $\beta^r_{emiJ_ik}$ for a particular part i of type r belonging to module m of asset e at time k. The level is decreased by one prior to a module assembly when such a part is needed, and this is indicated by module assembly beginning indicator $\alpha^r_{em2(k+1)}$. The above translate into the following inventory dynamics:

$$I_r(k+1) = I_r(k) + \sum_e \beta^r_{emiJ_ik} - \sum_e \alpha^r_{em2(k+1)} . \quad (22)$$

The inventory dynamics is subject to

"*Inventory Level Constraints*" requiring inventory levels to be non-negative:

$$I_r(k+1) \geq 0, \forall\, r \text{ and } k = 0,\ldots, K-1. \quad (23)$$

Similar to expected machine capacity constraints, they are approximated by the *Expected Inventory Level Constraints*:

$$\mathrm{E}[I_r(k+1)] \geq 0, \forall\, r \text{ and } k = 0,\ldots, K-1. \quad (24)$$

The objective is similar to (19) with the addition of costs for holding rotable inventory at γ_r per unit time for part type r:

$$J \equiv E\left[\sum_e \left(w_e T_e^2 + \beta_e E_e\right) + \sum_r \left(\sum_{k=0}^{K-1} \gamma_r I_r(k+1)\right)\right]. \quad (25)$$

The overall problem is to minimize (25) subject to arrival time constraints, operation precedence constraints, operation processing requirements, expected machine capacity constraints, expected cross-member precedence constraints, inventory dynamics (22), and expected inventory level constraints (24), with given machine capacities $\{M_{kh}\}$ and initial inventory levels $\{I_{r0}\}$. The decision variables are operation beginning times for disassembly, repair, and assembly operations.

5.2 Solution Methodology and Testing Results

The problem is solved by using the method presented in Section 4 with a coordinator to update the prices. Additional steps include substituting out inventory dynamics, and relaxing expected inventory level constraints by using an additional set of multipliers.

The method has been implemented by using MATLAB and tested on a Pentium III, 1.2 GHz, 256 SDRAM PC. Preliminary testing has been performed for a problem with 100 assets and each asset has one module, and that module has two parts. The surrogate subgradient (SSG) method with the variable target value step-sizing scheme was used, and the multipliers are updated after subproblems associated with two assets are solved. The average results for 3 cases are summarized in Table 1. Although the result after 1000 iterations is presented, the algorithm converged to a steady dual value after 400 or 600 iterations.

Table 1 Testing results for 100 assets using SSG

No. of Assets	Method	No. of Iteration	Dual Cost	CPU time (seconds)
100	SSG	1000	21236.47	330.82

The above results demonstrate that medium size problems can be effectively solved without excessive amount of computation time, also the scalability of the method for large scale practical problems.

6. Coordination without a Coordinator

6.1 Problem Description

Consider a supply chain without a

coordinator as depicted in Figure 2, where coordination is achieved through an iterative price updating process carried out in a distributed and asynchronous manner. The method has been implemented on a network of PCs, where each member is equipped with a PC and the method of Section 3. Communication among members is implemented in Java. Testing results are presented below (Luh, et al., 2003a).

6.2 Testing Results

For a simple example, the convergence of cross-member prices is shown in Figure 5.

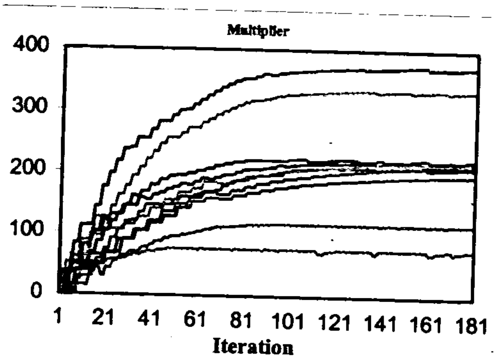

Figure 5 Convergence of prices

For another example, cross-member prices for specific orders are plotted against the number of iterations in Figure 6.

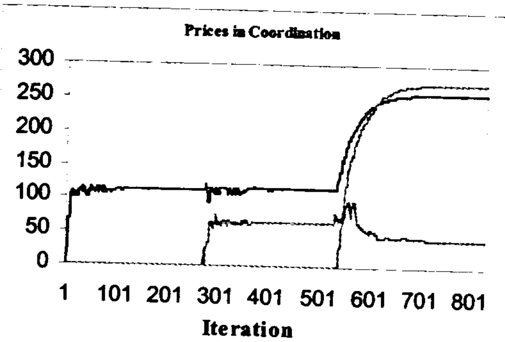

Figure 6 Changing of prices over time

It can be seen that the prices converge, although may change to different values upon the arrivals of new orders. They represent the costs for delaying delivery by one time unit, reflect the pressure on order delivery, and serve as coordination signals in a dynamic setting.

7. Summary and Prospects

7.1 Summary

In this paper, decomposition and coordination based on Lagrangian relaxation are identified as an effective way for manufacturing scheduling and supply chain coordination to control complexity and uncertainties. In the method, "hard" coupling constraints are "relaxed" by using Lagrange multipliers. For manufacturing scheduling, the relaxed problem is decomposed into part-wise subproblems that are not NP-hard. For supply chain coordination, the problem is decomposed into member-wise subproblems to be solved by individual members based on their internal conditions and cross-member prices. Coordination is achieved either with or without a coordinator through an iterative price updating process without accessing other members' private information or intruding their decision-making authorities. This decomposition and coordination approach and the resulting semi-closed loop policies effectively manage complexity and uncertainties with reasonable computational requirements. In addition, cross-member multipliers represent the sensitivity of the cost with respect to the precedence violations, thus reflect the pressure on order delivery, and provide quantitative and dynamic answers to the old question "time is money, but how much?"

7.2 Prospects

Challenges remain, including establishing models that cover a broader range of manufacturing and supply chain settings, and providing high quality solutions for real-time implementation in distributed decision networks. For example, existing methods are based mostly on model separability. If nonlinear cross-product terms or variance terms are included in the objective function as a measure of risk or for other reasons, the problem will become inseparable. Even if a problem is originally separable, non-linear cross product terms may be added to improve convergence, overcome the solution oscillation difficulty, or improve solution quality. As a result, the Lagrangian or augmented Lagrangian may become inseparable. Preliminary results to overcome such a difficulty by using the surrogate optimization framework are encouraging. From a different perspective, it was assumed in this paper that the overall objective function is the sum of individuals' objective functions. Members in a supply chain, however, could have conflicting goals. In such gaming situations, how to effectively coordinate members to obtain Pareto optimized solutions need to be further explored.

To provide high quality solutions for real-time implementation in distributed decision networks, a synergistic combination of off-line methods and on-line methods will be an important research topic. Based on Lagrangian relaxation, the offline method will optimize the expected objective over time, and address base-line solutions to large stochastic scenarios. Collaborating with the offline method, the online method could be developed based on constraint satisfaction and local search to provide robust and real-time resolution for supply chain coordination. Novel integration of the two could be explored to benefit from the features of each, e.g., use prices to guide the local method to preserve solution features that are desirable for long term quality, and use information on unsatisfied constraints as coordination signals for the off-line solver.

Another area of significant potential is to improve enterprises' profitability through integrated manufacturing scheduling, supply chain coordination, and yield management.[3] Manufacturing capacities are "perishable" as airline seats or advertising slots – if they are not used when available, their values are wasted or lost. Yield management techniques can be combined with scheduling, coordination, and real-time pricing to increase profitability.

8. Acknowledgement

This work was supported in part by the National Science Foundation under DMI-0223443, and by a contract from the United Technologies Research Center, USA.

[3] Yield management is to allocate limited capacities or to determine prices to optimize the total revenue or "yield" on investment. Airlines have used it to determine seasonal ticket prices based on demand and capacity available. Broadcasting companies have used it to determine how many advertising slots to sell now to the "upfront market" and how many to sell later at different prices.

References

[1] Anupindi, R., Bassok, Y., "Supply contacts with quantity commitments and stochastic demand," *Quantitative Models for Supply Chain Management*, S. Tayur and R. Ganeshan Eds., Boston: Kluwer Academic Publishers, pp. 198-232, 1999.

[2] Arntzen, B. C., Brown, G. G., Harrison, T. P., and Trafton, L. L., "Global supply chain management at digital equipment corporation," *Interface*, Vol. 25 (1), pp. 69-93. 1995.

[3] Bertsekas, D. P., *Nonlinear Programming*, Second Edition, Athena Scientific, Belmont, MA, 1999.

[4] Bertsekas, D. P., Tsitsiklis, J. N., *Parallel and Distributed Computation: Numerical Methods*, Prentice-Hall, 1989.

[5] 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, pp. 27-45, 1982.

[6] Chen, F., "Decentralized supply chains subject to information delays," *Management Science*, Vol. 45, No. 8, pp. 1076-1090, 1999.

[7] Chen, H., Chu, C., and Proth, J. M., "An Improvement of the lagrangian relaxation approach for job shop scheduling: a dynamic programming method," *IEEE Trans. on Robotics and Automation*, Vol.14, No.5, pp. 786-795, 1998.

[8] Chen, H. X., Luh P. B., and Fang, L., "A Time window based approach for job shop scheduling," *Proceeding of the 2001 IEEE International Conference on Robotics and Automation, Seoul Korea*, Vol.1, pp. 842-847, 2001.

[9] Cohen, M. A., Lee, H. L., "Strategic analysis of integrated production-distribution systems: models and methods," *Operations Research*, Vol. 36, No.2, pp. 216-228, 1989.

[10] Custodio, L. M. M., Sentieiro, J. J. S. and Bispo, C. F. G., "Production planning and scheduling using a fuzzy decision system," *IEEE Transaction on Robotics and Automation*, Vol. 10, No. 2, pp. 160-167, 1994.

[11] Fisher, M. L., "Optimal solution of scheduling problems using lagrange multipliers, Part I", *Operation Research*, Vol. 21, pp. 1114-1127, 1973.

[12] Gaonkar, R., Viswanadham, N., "Collaboration and information sharing in global contract manufacturing networks," *IEEE/ASME Trans. Mechatronics*, Vol. 6, pp. 366-376, 2001.

[13] Garey, M. R., Johnson, D. S., *Computers and Intractability*, San Francisco: W. H. Freeman and Co., 1979.

[14] Gou, L., Luh, P. B., and Kyoya Y., "Holonic manufacturing scheduling: architecture, cooperation mechanism, and implementation," *Computers in Industry*, Vol. 37, pp. 213-231, 1998.

[15] Guide, V. D. R., "Production planning and control for remanufacturing: Industry practice and research needs," *Journal of Operations Management*, Vol. 18, pp. 467-483, 2000.

[16] Kaskavelis, C. A., Caramanis, M. C., "Efficient lagrangian relaxation algorithms

for industry size job-shop scheduling problems," *IIE Transactions*, Vol. 30, No. 11, pp. 1085-1097, 1998.

[17]Kim, S. H. A., Cho, S., "Variable target value subgradient method," *Mathematical Programming*, Vol. 49, pp. 359-369, 1991.

[18]Kumar, P. R., Meyn, S. P., "Stability of queuing networks and scheduling policies," *IEEE Transaction on Automatic Control*, Vol. 40, No. 2, pp. 251-260, 1995.

[19]Kuziak, A., *Intelligent Manufacturing Systems*, Englewood Cliffs, Prentice-Hall, 1990.

[20]Lariviere, M. A., "Supply chain contracting and coordination with stochastic demand," *Quantitative Models for Supply Chain Management*, S. Tayur and R. Ganeshan Eds., pp. 233-268, Boston: Kluwer Academic Publishers, 1999.

[21]Lee, H. L., Whang, S., "Decentralized multi-echelon supply chains: incentive and information," *Management Science*, Vol. 45, No.5, pp. 633-640, 1999.

[22]Luh, P. B., Chen, D., and Thakur, L. S., "An Effective approach for job-shop scheduling with uncertain processing requirements," *IEEE Transactions on Robotics and Automation*, Vol. 15, No. 2, pp. 328-339, 1999.

[23]Luh, P. B., Hoitomt, D. J., Max, E., and Pattipati, K. R., "Schedule generation and reconfiguration for parallel machines," *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 6, pp. 687-696, 1990.

[24]Luh, P. B., Ni, M., Chen, H. X., and Thakur, L. S., "A price-based approach for activity coordination in a supply network," *IEEE Transactions on Robotics and Automation*, Vol. 18, No. 2, pp.335-346, 2003a.

[25]Luh, P. B., Soorapanth, S., Yu, D. Q., and Khibnik, A. I., "Scheduling asset overhaul and repair services," *Proceedings of the 2003 NSF Design, Service and Manufacturing Grantees and Research Conference*, Birmingham, AL, pp. 3109-3118, 2003b.

[26]Monahan, J. P., "A quantity discount pricing model to increase vendor profits," *Management Science*, Vol. 30, No. 6, pp. 720-726, 1984.

[27]Potts, C. N. and Van Wassenhove L. N., "Dynamic programming and decomposition approaches for the single machine total tardiness problem," *European Journal of Operational Research*, Vol. 32, pp. 405-414, 1987.

[28]Pinedo, M. L., *Scheduling – Theory, Algorithms and Systems*, New Jersey: Prentice Hall, 1995.

[29]Smith, R. G., "The contract net protocol: high-level communication and control in the distributed problem solver," *IEEE Transactions on Computers*, Vol. C-29, No. 12, pp. 1104-1113, 1980.

[30]Soroush, H. M., "Optimal sequence in stochastic single machine shops," *Computers and Operations Research*, Vol. 23, No. 7, pp. 705-721, 1996.

[31]Swaminathan, J. M., Smith, S. F., and Sadeh, N. M., "Modeling supply chain dynamics: a multi-agent approach," *Decision Sciences*, Vol. 29, No. 3, pp. 607-632, 1998.

[32]Wang, J., Luh, P. B., and Zhao, X., "An optimization-based algorithm for job shop scheduling," *SADHANA*, Vol. 22, Part 2, pp.

241-256, 1997.

[33]Zhang, Y.,. Luh, P. B., Narimatsu, K. T., Moriya, T. S., and Fang, L., "A macro-level scheduling method using lagrangian relaxation," *IEEE Transactions on Robotics and Automation*. Vol. 17, No. 1, pp. 70-79, 2001.

[34]Zhao, X., Luh, P. B. and Wang, J., "Surrogate gradient algorithms for lagrangian relaxation," *Journal of Optimization Theory and Application*, Vol. 100, No. 3, pp. 699-712, 1999.

Peter B. Luh received his B.S. degree in Electrical Engineering from National Taiwan University, Taipei, China in 1973, M.S. degree in Aeronautics and Astronautics Engineering from M.I.T., Cambridge, MA in 1977, and Ph.D. degree from Harvard University in 1980. Since 1980 he has been with the University of Connecticut, and currently is the SNET Professor of Communications & Information Technologies in the Department of Electrical and Computer Engineering, the Director of Taylor L. Booth Engineering Center for Advanced Technology at the University of Connecticut, and a Visiting Professor of Tsinghua University, Beijing, China. He is interested in planning, scheduling, and coordination of design, manufacturing, and supply chain activities; and schedule, bid, and portfolio optimization and load/price forecasting for power systems. He is a Fellow of IEEE, Editor-in-Chief of IEEE Transactions on Robotics and Automation, the founding Editor-in-Chief of the newly created IEEE Transactions on Automation Sciences and Engineering, an Associate Editor of IIE Transactions on Design and Manufacturing, and an Associate Editor of Discrete Event Dynamic Systems.

Weidong Feng received his B.S. degree in Industrial Electrical Engineering from the Light Industrial College of Zhengzhou, Zhengzhou, China in 1990, and M.S. degree in Systems Engineering in 1996 and Ph.D. degree in 1999 from Tianjin University, Tianjin, China. He had two years postdoctoral research experience in the Economics and Management School at Tsinghua University, Beijing, China, and also industrial experience in operations management and integrated product development. Since 2001 he has been a postdoctoral research fellow in the Department of Electrical and Computer Engineering, also a research fellow of Taylor L. Booth Engineering Center for Advanced Technology at the University of Connecticut. His current research interests focus on manufacturing/logistic planning and scheduling, supply chain coordination, service operations management, optimization technologies and their applications.