Price-Based Approach for Activity Coordination in a Supply Network

Peter B. Luh, *Fellow, IEEE*, Ming Ni, Haoxun Chen, and L. S. Thakur

***Abstract*—Pressed by market globalization and concomitant competition, more and more manufacturers are relying on their suppliers to provide raw materials and component parts so as to focus on their core competence. As a result, the coordination of activities across a network of suppliers becomes critical to quickly respond to dynamic market conditions. In this paper, a novel framework combining mathematical optimization and the contract net protocol is presented for make-to-order supply network coordination. Interactions among organizations are modeled by a set of interorganization precedence constraints and the objective is to achieve the organizations' individual and shared goals of fast product delivery and low inventory. These interorganization constraints are relaxed by using a set of interorganization prices that represent marginal costs per unit time for the violation of such constraints. The overall problem is thus decomposed into organizational subproblems, where individual organizations schedule their activities based on their internal situations and interorganization prices. Coordination is achieved through an iterative price-updating process carried out in a distributed and asynchronous manner. With prices dynamically updated and schedules adjusted, this approach coordinates activities to fulfill existing commitments while maintaining agility to take on new orders. Numerical testing results show that interorganization prices converge and prices may change as new orders arrive to reflect the new pressure on deliveries. The method thus provides a novel framework for activity coordination across a supply network and answers in a quantitative manner the perennial question, "Time is money, but how much?"**

***Index Terms*—Activity coordination, Lagrangian relaxation, supply chain management.**

I. Introduction

A SUPPLY NETWORK is a network of autonomous or semiautonomous organizations such as suppliers, manufacturers, warehouses, distributors, and retailers through which goods are produced and delivered to customers. Pressed by market globalization and concomitant competition, more and more manufacturers are relying on their suppliers to provide raw materials and component parts so as to focus on their core competence. As a result, the coordination of activities across a network of suppliers becomes critical to quickly respond to dynamic market conditions [1]. Activity coordination involves proper scheduling and synchronization of activities within and across organizations for fast product delivery and low inventory, assuming that quality and cost aspects are satisfied. It is particularly important for "make-to-order" supply networks, since their flow of materials is triggered by dynamic customer orders and there is little work-in-process inventory to buffer coordination inefficiencies.

Effective coordination, however, is difficult, since activities may be related in a complex way and a delay of one activity may have a domino effect on activities linked through precedence relationships or through sharing of common resources. Most coordination problems, when formulated mathematically, are NP-hard optimization problems and the computational requirements to obtain an optimal solution increase drastically with problem size. In addition, supply networks generally operate in a dynamic environment, as the arrival of an urgent order may trigger a chain of events causing existing commitments to be compromised. Furthermore, organizations generally have their own private information and decision-making authority. Effective coordination methods must be developed without accessing others' private information or intruding on their decision-making authority.

In this paper, a novel approach, combining mathematical optimization and the contract net protocol, is presented for make-to-order supply network coordination. In the formulation, each organization has its own information and decision-making authority, and interactions among organizations are modeled by a set of interorganization precedence constraints. The objective is to achieve the organizations' individual and shared goals of fast product delivery and low inventory over a specific planning horizon as presented in Section III. Motivated by the pricing mechanism for the coordination of resource allocation, these constraints are relaxed by using a set of interorganization prices. These prices are Lagrange multipliers associated with interorganization constraints (as opposed to multipliers associated with resource constraints in traditional resource allocation problems) and represent marginal costs per unit time for the violation of such constraints. The overall problem is, thus, decomposed into organizational subproblems, where individual organizations schedule their activities based on their internal situations and interorganization prices without accessing others' private information or intruding on their decision-making authority.

Manuscript received August 17, 2001; revised July 12, 2002. This paper was recommended for publication by Associate Editor C. Chu and Editor N. Viswanadham upon evaluation of the reviewers' comments. This work was supported in part by the National Science Foundation under Grant DMI-9813176 and in part by a grant from the Connecticut Yankee Ingenuity Grant Program. This paper was presented in part at the International Computer Science Convention Symposia on Intelligent Systems and Applications, Wollongong, Australia, December 11–15, 2000.

P. B. Luh and M. Ni are with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269 USA (e-mail: Peter.Luh@uconn.edu, mingni@engr.uconn.edu).

H. Chen was with the University of Connecticut, Storrs, CT. He is now with the Université de Technologie de Troyes, Laboratoire LOSI, Troyes Cedex 10010, France (e-mail: Haoxun.Chen@univ-troyes.fr).

L. S. Thakur is with the Department of Operations and Information Management, University of Connecticut, Storrs, CT 06269 USA (e-mail: Thakur@uconn.edu).

Digital Object Identifier 10.1109/TRA.2003.809589

Coordination is achieved through an iterative price-updating process carried out in a distributed and asynchronous manner as presented in Section IV. With prices dynamically updated and schedules adjusted, this approach coordinates activities to fulfill existing commitments, while maintaining agility to take on new orders. Numerical testing results presented in Section V show that interorganization prices converge and these prices may change as new orders arrive to reflect the new time pressure on order deliveries. The method provides a novel structure for activity coordination and answers in a quantitative manner the perennial question, "Time is money, but how much?"

II. Literature Review

In most multiagent approaches, coordination among agents is carried out by exchanging information and imposing constraints using rule-based methods [2], [3], [11]. A commonly used framework is the contract net protocol, which was originally developed for distributed problem solving by mimicking the human contract negotiation process [10] and has then been extended for supply chain coordination. For a two-tier supply chain with a manufacturer (contract manager) and its potential suppliers (contractors), it works as follows.

— Upon receiving an order from its customer, a manufacturer announces *requests for bids* for raw materials or component parts to its potential suppliers. Part specifications and requested delivery dates are included in the requests for bids.
— Potential suppliers check their own status and submit *bids* to the manufacturer.
— The manufacturer selects bids based on the announced criteria and awards contracts to the suppliers selected.
— The selected suppliers send acknowledgments back to the manufacturer and abide by the contracts to deliver the raw materials or component parts.

The above process mimicks the interactions among organizations in a supply chain. The resulting contracts generally are binding and there is no flip-flop of decisions once contracts are settled. For a multitier supply chain, a supplier may further announce requests for bids and award contracts to its own suppliers.

To meet various requirements in supply chain coordination, the original contract net protocol has been extended. In the mediated constraint relaxation approach of Beck and Fox [3], a mediator, an additional agent, was introduced to gather information and to form a constraint graph. This mediator is also responsible for resolving conflicts among agents via constraint relaxation. An approach that combines a contract net protocol-based bidding mechanism with a mediation method under a multiagent framework was presented in [9]. A hierarchical agent architecture was presented in [8], where agents were organized hierarchically. Low-level agents are responsible for short-term planning and scheduling within individual facilities, and high-level agents are responsible for overall strategic and tactical decisions. A time-bound negotiation framework was developed in [7], where a qualified bid should be selected within a specified amount of time.

In the above methods, coordination is generally performed in an *ad hoc* manner following some prespecified rules. The quality of solutions is problem specific and cannot be easily quantified. Furthermore, it is difficult to adjust the contracts to accommodate dynamic changes such as the arrivals of urgent new orders.

The above limitations can be overcome if appropriate architecture and methods are developed. Important lessons can be learned from the market economy, where prices play a key role in coordinating the decentralized allocation of resources. A recent example is the use of machine prices for job shop scheduling for a separable model [12], [14]. The key idea is decomposition and coordination, where decomposition is achieved by relaxing coupling machine capacity constraints using soft prices or Lagrangian multipliers, and coordination is accomplished through the iterative updating of prices to adjust operation beginning times. Numerical results show that near-optimal schedules can be obtained for problems with up to 50 000 operations within a reasonable amount of computation time on a Pentium III 500-MHz PC [14]. This price-based coordination idea has been extended to coordinate multiple cells in a factory [6]. In that problem, the relationships among cells are modeled as *intercell precedence constraints*, and these constraints are relaxed by using a set of *intercell prices*, which are similar to but distinct from prices for resource allocation. The overall problem is, thus, decomposed into cell-level subproblems, which, in turn, are solved by using the price-based scheduling methods described above. Coordination across cells is achieved through the iterative adjustment of intercell prices by an additional coordinator in a *centralized and synchronous* manner without accessing individual cells' local information or intruding on their decision-making authority.

A planning model for distributed manufacturing was presented in [13]. The model is a multicommodity network model where each commodity represents a product to be produced by a network of facilities. A Lagrangian relaxation approach was developed to decompose the problem into a number of single-product, multifacility subproblems and a resource subproblem to generate near-optimal plans.

III. Problem Formulation

The goal of this paper is to present a new framework and the corresponding approach to optimize activity coordination among organizations in a make-to-order supply network. In the following, a price-based coordination architecture is first presented in Section III-A, and the mathematical formulation for a snapshot problem is presented in Section III-B.

A. System Architecture

1) Price-Based Coordination Architecture: Organizations in a supply network may be divisions within a company or different companies, and they may be organized in a hierarchical or heterarchical manner. Since they are mostly autonomous or semiautonomous with private information and individual decision-making authority, centralized models and methods are not suitable to coordinate their activities. A decentralized optimization model integrated with the contract net protocol will, therefore, be established. In the model, organizations are represented by autonomous and cooperating decision

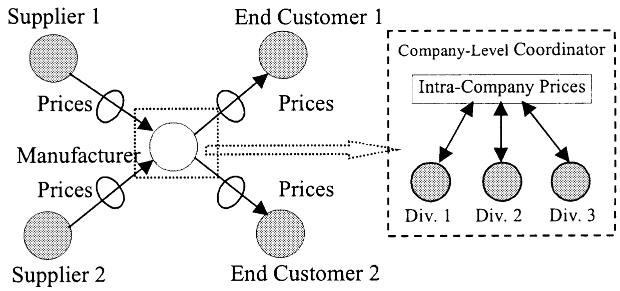

Fig. 1. Price-based coordination of a supply network.

makers, and interactions among them are modeled by a set of interorganization precedence constraints.

After an organization received an order (or a tentative order related to a request for bids), it checks its bill of materials. *Potential suppliers* of raw materials or component parts are then identified. A bidding process similar to that of the contract net protocol is carried out where bids are solicited, appropriate suppliers are identified, contracts are awarded, and deliveries are coordinated. On top of this contract net protocol, prices associated with interorganization precedence constraints are used to coordinate activities across organizations. In the process, each organization solves its scheduling subproblem by optimizing its objective function (which is its own objective modified by penalty terms depending linearly on interorganization prices, see (12) below) subject to its internal resource capacity and operation precedence constraints. To coordinate activities across organizations, these prices are iteratively updated based on the degrees of constraint violation. In the view that organizations may have a wide range of computing and communication capabilities, a centralized and synchronous price updating mechanism is not appropriate. Coordination is, therefore, carried out by a distributed and asynchronous price-updating mechanism. In this way, suppliers are selected and delivery dates are determined and coordinated for overall system performance. A simple network with two end customers, one manufacturer, and two suppliers is depicted in Fig. 1, where the manufacturer has three divisions.

2) Coordination Process: To be more specific, consider the case where an organization received a request for bids containing a *requested delivery date*, and decides to submit a bid containing a *tentative delivery date*. This date is determined through solving the organization's scheduling subproblem. In the process, two kinds of orders are considered: orders already under contracts (*contracted orders*), and orders in response to requests for bids (*tentative orders*). For contracted orders, one option is to freeze their schedules to avoid missing their contracted due dates. The lack of flexibility, however, may lead to inefficient resource utilization and the failure to submit competitive bids for new requests for bids. In our approach, existing schedules can be modified, however, with stiff penalties for missing the due dates. In this way, resources can be more efficiently utilized and bids can be submitted with competitive tentative delivery dates. An iterative process is then carried out across organizations to reconcile the requested delivery dates and tentative delivery dates while optimizing system-wide performance. If an organization's tentative delivery date cannot

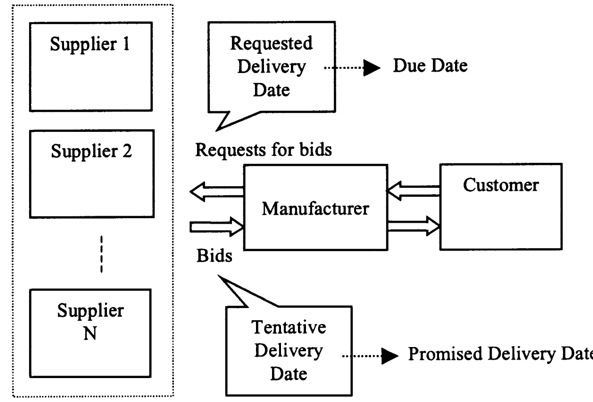

Fig. 2. Bidding process for a three-tier network.

meet its customer's requested delivery date, the associated interorganization price is increased based on the degrees of constraint violation. In this way, the increased price forces the organization to provide an earlier tentative delivery date for the next iteration, and at the same time, adds pressure to the customer to consider a later requested delivery date. This price is similarly adjusted by the customer organization. To avoid two organizations adjusting the same price at the same time, a "token" is introduced for each such price and is circulated between these two organizations. Only the one holding the token can adjust the price. Deliveries of contracted orders may also be adjusted through a similar price-updating process.

When prices are close to convergence, the customer organization selects an appropriate supplier for each request for bids. The selected supplier's tentative delivery date then becomes its promised delivery date and the organization's requested delivery date becomes the order's due date. This mechanism is depicted in Fig. 2. Upon the arrivals of new orders or upon unexpected disruptions, rescheduling is triggered and another cycle of the price-updating process is carried out. The detailed derivation of the coordination process will be presented in Section IV after a snapshot problem is formulated next.

B. Problem Formulation

Consider a decentralized formulation for a snapshot of the problem. It starts with the description of individual organizations based on [12] and [14], followed by a description of interactions across organizations, and then, the overall objective.

1) Formulation of Individual Organization Problems:

Variables: Suppose that there are I organizations in the supply network and Organization $i (i = 1, 2, \ldots, I)$ has J_i orders to be scheduled. Among J_i, let the *subset of tentative orders* be denoted as O_i^T and the *subset of contracted orders* be denoted as O_i^C. The jth $(j = 1, 2, \ldots, J_i)$ order of Organization i is denoted as (i, j) and is associated with an order or a request for bids $(i^{c(i,j)}, j^{c(i,j)})$ of a customer organization $i^{c(i,j)}$. To simplify the notation, $i^{c(i,j)}$ will be denoted as i^c and $(i^{c(i,j)}, j^{c(i,j)})$ as (i^c, j^c) when there is no confusion. A tentative order (i, j) $(\in O_i^T)$ is associated with a given requested delivery date d_{ij}^r specified by i^c, and a contracted order (i, j) $(\in O_i^T)$ is associated with a given due date d_{ij} imposed by i^c. Order (i, j) may consist of a set of operations specified by its process plan. In addition, it may need K_{ij} raw materials or component parts, each associated with a request for bids. Let the set of bids received for the kth ($k = 1, 2, \ldots, K_{ij}$) raw material or component part of (i, j) be denoted as S_{ijk}. Order (i, j) may, therefore, be associated with an order or a tentative order $(i^{s(i,j)}, j^{s(i,j)})$ of a supplier organization $i^{s(i,j)}$. To simplify the notation, $i^{s(i,j)}$ will be denoted as i^s and $(i^{s(i,j)}, j^{s(i,j)})$ as (i^s, j^s) when there is no confusion. Order (i, j) is, thus, related to a tentative delivery date $d^t_{i^s j^s}$ offered by i^s if $(i, j) \in O_i^T$, or a promised delivery date $d^p_{i^s j^s}$ if $(i, j) \in O_i^C$. Organization i's decision variables include the tentative delivery date d^t_{ij} to its customer if $(i, j) \in O_i^T$, and the promised delivery date d^p_{ij} if $(i, j) \in O_i^C$. In addition, Organization i determines the requested delivery date $d^r_{i^s j^s}$ to be imposed on its potential supplier i^s if $(i, j) \in O_i^T$, and the due date $d_{i^s j^s}$ if $(i, j) \in O_i^C$. Additional decision variables are operation beginning times for all the operations within Organization i.

For simplicity, but without loss of generality, it is assumed that the production of (i, j) can only start after all the required raw materials or component parts have been received, the transportation times for such deliveries are negligible, and only one supplier will finally be selected for each request for bids.

Organization internal constraints: For each organization, internal constraints include operation precedence constraints, operation processing requirements, and resource capacity constraints.

1) *Operation precedence constraints*. Each order may consist of a set of operations as specified by its process plan. An operation cannot be started until all its preceding operations have been finished.
2) *Operation processing requirements*. Each operation should be processed by a specified set of resources for a particular duration of time.
3) *Resource capacity constraints*. The utilization of a resource should be less than or equal to the capacity of that resource for each time interval. A resource may be a machine type or a class of operators.

Details of the above constraints can be found in [12] and [14], and for brevity, they will not be elaborated here. We shall simply highlight two versions of operation precedence constraints internal to i and relevant to our derivation here. First, (i, j) cannot be delivered until its last operation has been completed, plus possibly a required slack time. For $(i, j) \in O_i^T$, this is

$$c_{ij} + s^c_{ij} \leq d^t_{ij} \tag{1}$$

where c_{ij} is the order completion time (the completion time of its last operation) and s^c_{ij} is the required slack time. For $(i, j) \in O_i^C$, this is

$$c_{ij} + s^c_{ij} \leq d^p_{ij}. \tag{2}$$

Second, $(i, j) \in O_i^T$ can only be started after all the required materials or component parts have been received (as internally characterized by the requested delivery date $d^r_{i^s j^s}$) plus possibly a required slack time, i.e.,

$$d^r_{i^s j^s} + s^b_{i^s j^s} \leq b_{ij} \quad \forall (i^s, j^s) \in S_{ijk} \tag{3}$$

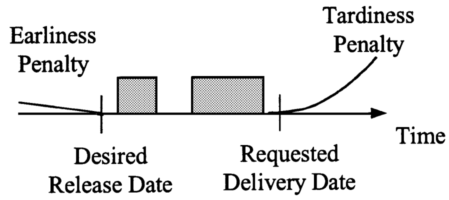

Fig. 3. Penalty function for a tentative order.

where b_{ij} is the order beginning time (the beginning time of its first operation), and $s^b_{i^s j^s}$ the required slack time. Similarly, for $(i, j) \in O_i^C$, this is

$$d^r_{i^s j^s} + s^b_{i^s j^s} \leq b_{ij} \quad \forall (i^s, j^s) \in S_{ijk}. \tag{4}$$

Organization objective functions: Assuming that quality and cost aspects are satisfied, an organization in a make-to-order environment wants to ensure on-time delivery of orders while minimizing its inventory. The objective function considered is, therefore, a weighted sum of order tardiness and earliness penalties following [12] and [14]. For $(i, j) \in O_i^T$, the tardiness is defined as $T_{ij} = \max[0, c_{ij} - d^r_{ij}]$, and earliness $E_{ij} = \max[0, \bar{b}_{ij} - b_{ij}]$, where $\bar{b}_{ij}$ is the desired release date calculated, for example, based on the d^r_{ij} and the required processing times. The cost for (i, j) is a weighted sum of quadratic tardiness and linear earliness penalties, [1] i.e.

$$f_{ij}(T_{ij}, E_{ij}) = w_{ij} T^2_{ij} + \beta_{ij} E_{ij}, (i, j) \in O_i^T \tag{5}$$

where parameters w_{ij} and β_{ij} are nonnegative penalty coefficients. As shown in Fig. 3, $w_{ij} T^2_{ij}$ represents the importance of on-time delivery, and $\beta_{ij} E_{ij}$ the importance of low work-in-process inventory.

For $(i, j) \in O_i^C$, the cost is similarly defined, except that tardiness is calculated with respect to its due date d_{ij} (as opposed to d^r_{ij}), i.e., $T_{ij} = \max[0, c_{ij} - d_{ij}]$. In addition, a step penalty for missing the due date is added

$$f_{ij}(T_{ij}, E_{ij}) = w_{ij} T^2_{ij} + \beta_{ij} E_{ij} + \gamma_{ij} \mathrm{Step}(T_{ij}), (i, j) \in O_i^C \tag{6}$$

where $\mathrm{Step}(T_{ij})$ equals one if $T_{ij} \geq 0$, and zero, otherwise. This last term represents a stiff penalty to discourage the violation of the due date as shown in Fig. 4.

The objective function of Organization i is then the sum of penalties for all its orders

$$J_i = \sum_j f_{ij}(T_{ij}, E_{ij}). \tag{7}$$

2) Formulation of the Overall Problem:

Interorganization precedence constraints: Interorganization precedence relationships impose constraints across organizations. For $(i, j) \in O_i^T$, the tentative delivery

[1]The derivation is not restricted to quadratic or linear penalty functions. The only requirement is that they should be order-wise additive to ensure the separability of the overall formulation.

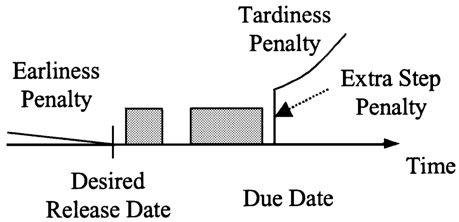

Fig. 4. Penalty function for a contracted order.

date d_{ij}^t should be less than or equal to the requested delivery date specified by i^c, i.e.

$$d_{ij}^t \leq d_{ij}^r \quad \forall (i,j) \in O_i^T. \tag{8}$$

For $(i,j) \in O_i^C$, the original tentative delivery date became the promised delivery date d_{ij}^p, and the requested delivery date became the order's due date d_{ij}. The due date, however, may not be met during rescheduling in view of disruptions caused by urgent new orders or other uncertainties. For coordination purposes, a new promised delivery date d_{ij}^p and a new requested delivery date d_{ij}^r are, therefore, established subject to the following constraint:

$$d_{ij}^p \leq d_{ij}^r \quad \forall (i,j) \in O_i^C. \tag{9}$$

For each of the above interorganization constraints (8) and (9), a token is established to determine whether i or i^c should update the corresponding interorganization price. Only the organization holding the token can update the price, and the token is exchanged between i and i^c.

Overall objective function: For the benefit of the entire supply network, the overall objective function is assumed to be the sum of individuals' objectives

$$J = \sum_{i=1}^{I} J_i = \sum_i \sum_j f_{ij}(T_{ij}, E_{ij}). \tag{10}$$

Different objective functions can be considered to reflect different aspects of supply network coordination. The only requirement is that the function should be organization-wise additive. Organizations in a supply network, however, could have conflicting goals. Although maximizing the total welfare of member organizations is a sensible thing for many cases, this may not be reasonable when the objectives of organizations are too far apart. This case, however, is out of the scope of this paper.

As the above formulation is developed in a bottom-up fashion, it is a decentralized model. From another point of view, the formulation is "separable," since the interorganization constraints (8) and (9) that couple organizations together and the overall objective function (10) are organization-wise additive. A decomposition and coordination approach will be developed next.

IV. Solution Methodology

A. Overview

Through relaxing interorganization constraints by using "soft" prices or Lagrange multipliers, the overall snapshot problem can be decomposed into a set of subproblems, one for each organization. An organization solves its own subproblem by using, for example, the Lagrangian relaxation technique [12] and [14] based on the information received from customers and suppliers and obtains a schedule for both contracted and tentative orders. For the subset of orders whose tokens are held by the organization, the associated prices are updated based on the degrees of interorganization precedence constraint violation. The organization then sends the resulting requested delivery dates, tentative or promised delivery dates, as well as new prices and tokens, to its suppliers and customers and waits for their responses. After the organization received new information from suppliers or customers and the corresponding tokens, the subproblem is resolved, prices are readjusted, and the process continues until interorganization prices are close to convergence. Suppliers for tentative orders are then selected from possibly multiple potential suppliers based on their tentative delivery dates and the corresponding prices. In view that interorganization precedence constraints (8) and (9) have been relaxed in the iterative optimization process, subproblem solutions, when put together, may not constitute a feasible solution. Heuristics based on mutually agreed-upon rules are thus applied to generate a feasible schedule across the network satisfying (8). A tentative order then becomes a contracted order, with its supplier's tentative delivery date becoming the promised delivery date, and the corresponding requested delivery date becomes the order's due date. In addition, an extra step penalty is added to the cost to discourage missing the due date. Upon the arrivals of new orders or upon unexpected disruptions, rescheduling is triggered and another cycle of the price-updating process is carried out. In rescheduling, most decision variables are reoptimized except the due dates, which remain fixed unless agreed upon by both organizations involved. This dynamic process, including snapshot problem solving and rescheduling, is summarized in Fig. 5, and the derivations of specific steps are presented next.

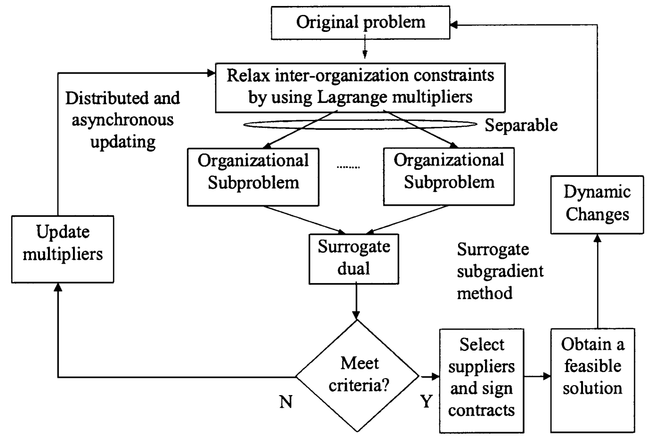

Fig. 5. Summary of the solution process.

B. Problem Decomposition

To solve the snapshot problem, the "hard" interorganization precedence constraints (8) and (9) are first relaxed by using "soft" prices or Lagrange multipliers. Let λ_{ij} be the price between (i,j) and (i^c, j^c), the relaxed snapshot problem is given by

$$\min L, \text{with } L \equiv \sum_i \sum_j f_{ij}\left(T_{ij}, E_{ij}\right) + \sum_i \left[\sum_{(i,j)\in O_i^T} \left(\lambda_{ij}\left(d_{ij}^t - d_{ij}^r\right)\right)\right] + \sum_i \left[\sum_{(i,j)\in O_i^C} \left(\lambda_{ij}\left(d_{ij}^p - d_{ij}^r\right)\right)\right] \tag{11}$$

subject to internal constraints of all organizations. In view of the separability of the original formulation, the relaxed problem can be decomposed into I subproblems, one for each organization. Collecting all the terms related to Organization i in (11), the subproblem for i is given by (12), shown at the bottom of the page, subject to the internal constraints of Organization i. In (12), (i,j) is to be delivered to i^c as a component part of (i^c, j^c) and is associated with due date d_{ij} and price λ_{ij}. Similarly, (i,j) requires the delivery of (i^s, j^s) from i^s, with $\lambda_{i^s j^s}$ as the associated price. If $(i,j) \in O_i^T$ has multiple potential suppliers providing bids for the kth raw material or component part, then each one is associated with such a price as will be explained at the end of the next subsection.

C. Individual Organization Decision Making

With the above decomposition, subproblem (12) for i is to minimize its objective function, which is its original objective (7) modified by penalty terms depending linearly on prices, subject to i's internal operation precedence, resource capacity constraints, and operation processing requirements. For $(i,j) \in O_i^T$, the decision variables are the beginning times of various operations, tentative delivery date d_{ij}^t to be offered to its potential customer i^c, and requested delivery date $d_{i^s j^s}^r$ to be imposed on its potential supplier i^s. The decision variables are similar for $(i,j) \in O_i^C$, with tentative delivery date replaced by promised delivery date d_{ij}^p.

This subproblem is similar to a job-shop scheduling problem and is NP-hard. Although many methods can be used to solve it, the Lagrangian relaxation technique of [12] and [14] is selected since it can efficiently obtain near-optimal solutions and is consistent with the interorganization price coordination framework. In the method, the coupling resource capacity constraints are relaxed by using another set of Lagrangian multipliers, which are "intraorganization prices" for resource utilization. Details of the method can be found in the references, and we shall only comment on how to select suppliers.

Suppose that i is seeking a supplier from the set of bids received, S_{ijk}, for the kth raw material or component part of tentative order (i,j). When (12) is solved, these bids are considered one at a time, with the corresponding cost for (i,j) calculated by using dynamic programming as presented in [12]. The potential supplier with the minimal cost is then selected to obtain the solution for (12). At the convergence or near-convergence of the interorganization price-updating process, the tentative order (i^s, j^s) associated with the minimal order cost is selected and assumed fixed during rescheduling unless agreed upon by both organizations.

D. Coordination Procedure and Convergence

As described earlier, interorganization prices are iteratively updated to coordinate activities across organizations. Given a price vector λ that includes all the interorganization prices, i.e.,

$$\lambda = \left(\lambda_{ij}\right)^T \tag{13}$$

the dual problem is given as

$$\max_{\lambda \geq 0} q(\lambda) \tag{14}$$

where $q(\lambda)$ is the optimal value of the relaxed problem (11). An optimal λ that maximizes $q(\lambda)$ is denoted as

$$\lambda^* = \left(\lambda_{ij}^*\right)^T. \tag{15}$$

An effective method to solve such a dual problem is the "surrogate subgradient method" [15]. In contrast to traditional subgradient methods where all the subproblems are optimally solved to obtain a subgradient direction to update the multipliers, this method solves only a subset of subproblems to obtain a surrogate subgradient direction which forms an acute angle with the direction toward λ^*. A step size is then selected to ensure that λ moves closer to λ^*. In the current framework, a price component is adjusted by the two associated organizations based on their subproblem solutions. Since not all the subproblems are solved at the same time, surrogate subgradients, as opposed to subgradients, are available. In addition, since subproblems are solved in a distributed and asynchronous manner, a distributed and asynchronous version of the surrogate subgradient method needs to be developed. In the following, the distributed and asynchronous surrogate

$$\min\ L_i, with\ L_i \equiv \sum_j f_{ij}\left(T_{ij}, E_{ij}\right) + \sum_{(i,j)\in O_i^T} \left(\lambda_{ij} \cdot d_{ij}^t\right) - \sum_{(i,j)\in O_i^T} \left[\sum_{k\ such\ that (i^s,j^s)\in S_{ijk}} \left(\lambda_{i^s j^s} \cdot d_{i^s j^s}^r\right)\right] + \sum_{(i,j)\in O_i^C} \left(\lambda_{ij} \cdot d_{ij}^p\right) - \sum_{(i,j)\in O_i^C} \left[\sum_{k\ such\ that (i^s,j^s)\in S_{ijk}} \left(\lambda_{i^s j^s} \cdot d_{i^s j^s}^r\right)\right] \tag{12}$$

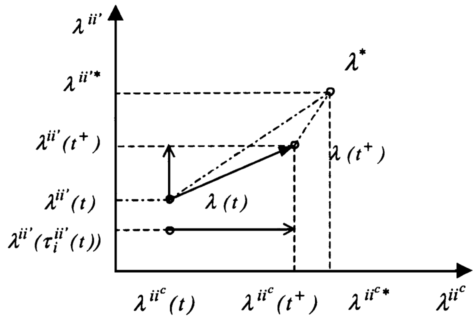

Fig. 6. Price components moving to optima.

subgradient method for price updating is first presented. It will be shown that by appropriately updating price components, λ will converge to λ^* and $q(\lambda)$ is maximized. The selection of step sizes in algorithm implementation is then addressed.

1) Overview of Price Adjustment Process: Consider the price-adjustment process between i and i^c. In view that there could be multiple interorganization precedence relationships between i and i^c, let such prices form a vector λ^{ii^c}, which is a component of λ. This λ^{ii^c} is updated by either i or i^c, and its optimum, λ^{ii^c*}, is a component of λ^*. In view of communication delays, i at time t may have delayed knowledge of λ^{ii^c} up to time $\tau_i^{ii^c}(t)$ only, with $0 \leq \tau_i^{ii^c}(t) \leq t$. It is assumed that the total asynchronism assumption [5, p. 430] holds, i.e., all price components are updated infinitely often and old price information is eventually purged from organizations. This implies that for any given time t_1, there exists a time $t_2(> t_1)$ such that for all $t \geq t_2$, $\tau_i^{ii^c}(t) \geq t_1$ for all i and i^c.

To present the updating of price components, consider that i possesses the token to update λ^{ii^c} at time t, and thus, $\tau_i^{ii^c}(t) = t$. At time t, i may have delayed knowledge of prices between i and its customers or suppliers other than i^c, i.e., $\lambda^{ii'}(\tau_i^{ii'}(t))$, where i' is a customer or suppler of i and $0 \leq \tau_i^{ii'}(t) \leq t$. The updating completes at time $t^+(> t)$. Assume that λ^{ii^c} can be updated toward its optimum λ^{ii^c*} along a proper direction based on delayed $\lambda^{ii'}(\tau_i^{ii'}(t))$, then the overall distance between λ and λ^* is reduced from $\lambda(t)$ to $\lambda(t^+)$. This can be shown by an example in Fig. 6, where for simplicity, price vector λ has two scalar components, λ^{ii^c} and $\lambda^{ii'}$.

In the figure, $\lambda^{ii'}$ is also changed from $\lambda^{ii'}(t)$ to $\lambda^{ii'}(t^+)$ by i' during the updating of λ^{ii^c}.

As the dual function $q(\lambda)$ is concave and the dual problem (14) is subject to positive orthant constraint $\lambda \geq 0$ only, the updating can be performed iteratively without being trapped at a local maximum or terminated prematually. In addition, as the method uses subgradients only, there is no need for global information such as Hessian in the Newton method. Consequently, by iteratively updating individual price components as presented next to their optima, the distance between λ and λ^* will be continually reduced until λ converges to λ^*.

2) Distributed and Asynchronous Surrogate Subgradient Method: Consider that λ^{ii^c} is updated by i at time t. Similar to the synchronous surrogate subgradient method, the updating of λ^{ii^c} is along the surrogate subgradient direction, i.e.,

$$\lambda^{ii^c}\left(t^+\right) = \lambda^{ii^c}(t) + \alpha^{ii^c}(t) g^{ii^c}(t). \tag{16}$$

In the above, α^{ii^c} is the step size and g^{ii^c} is a surrogate subgradient representing the degree of constraint violation. For $(i,j) \in O_i^T$, the component g_{ij} of g^{ii^c} is given by

$$g_{ij} = d_{ij}^t - d_{ij}^r \tag{17}$$

following (8) and for $(i,j) \in O_i^C$, g_{ij} is given by

$$g_{ij} = d_{ij}^p - d_{ij}^r. \tag{18}$$

The updating in (16) is to reduce the distance between λ^{ii^c} and the optimal λ^{ii^c*}, and requires g^{ii^c} to form an acute angle with the direction toward λ^{ii^c*}, i.e.

$$(\lambda^{ii^c*} - \lambda^{ii^c}(t))^T g^{ii^c}(t) > 0. \tag{19}$$

Assuming that (19) can be evaluated by i. If it holds, a positive step size α^{ii^c} is taken, satisfying

$$0 < \alpha^{ii^c}(t) < \frac{2\left(\lambda^{ii^c*} - \lambda^{ii^c}(t)\right)^T g^{ii^c}(t)}{||g^{ii^c}(t)||^2}. \tag{20}$$

It will be proved in *Proposition 1* that λ^{ii^c} moves closer to λ^{ii^c*}. If (19) is not satisfied, a null step is taken, i.e.

$$\alpha^{ii^c}(t) = 0. \tag{21}$$

After updating, tentative delivery dates for tentative orders, promised delivery dates for contracted orders, $\lambda^{ii^c}(t^+)$, and the corresponding tokens are sent to i^c. In order for i^c to evaluate (19) and to compute its own step size, the subproblem cost of i is also sent to i^c, and the process will then be carried out by i^c.

To ensure convergence, it is required that proper directions satisfying (19) can be found so that the distance between price vector λ to λ^* is reduced until λ^* is reached. In addition, to implement the method, organization i needs to evaluate $(\lambda^{ii^c*} - \lambda^{ii^c}(t))^T g^{ii^c}(t)$ and determine the step size. Due to communication delays, this is performed based on the delayed price information, e.g., $\lambda^{ii'}(\tau_i^{ii'}(t))$. In Section IV-D.3, it will be shown that λ converges to λ^* by appropriately updating price components following (16)–(21), where (19) is satisfied for at least one component under the total asynchronism assumption. In Section IV-D.4, the evaluation of (19) and the calculation of step size in the absence of λ^{ii^c*} will then be presented based on a two-organization problem involving i and i^c, where the problem is formulated based on the delayed price information.

3) Convergence of the Price Adjustment Process: In the following, *Propositions 1* and *2* provide the conditions for price components to move closer to their optima. *Proposition 3* shows that these conditions are satisfied for at least one price component under the total asynchronism assumption, so that the distance from λ to λ^* is reduced until λ^* is reached. *Theorem 1* then establishes the convergence of λ to λ^*.

Proposition 1: If (19) is satisfied and a step size α^{ii^c} is taken following (20), then

$$\left\|\lambda^{ii^c*} - \lambda^{ii^c}\left(t^+\right)\right\| < \left\|\lambda^{ii^c*} - \lambda^{ii^c}(t)\right\|. \tag{22}$$

Otherwise, a null step size is taken and

$$\left\|\lambda^{ii^c*}-\lambda^{ii^c}\left(t^+\right)\right\|=\left\|\lambda^{ii^c*}-\lambda^{ii^c}(t)\right\|. \quad (23)$$

Consequently

$$\left\|\lambda^*-\lambda\left(t^+\right)\right\|\le\|\lambda^*-\lambda(t)\|. \quad (24)$$

Proof: See Appendix I.

From *Proposition 1*, λ^{ii^c} gets closer to λ^{ii^c*} when i updates λ^{ii^c} with a positive step size and λ^{ii^c} remains the same with a null step size.

Proposition 2: Suppose that i^c has the token and updates λ^{ii^c} at time t_2 following (19)–(21), and i holds $\lambda^{ii^c}(t_1)$ at time t_2 with $t_2 > t_1$. At time $t_3(> t_2^+)$, i received the updated value $\lambda^{ii^c}(t_2^+)$ from i^c, then

$$\left\|\lambda^{ii^c*}-\lambda^{ii^c}\left(t_2^+\right)\right\|\le\left\|\lambda^{ii^c*}-\lambda^{ii^c}\left(t_1\right)\right\|. \quad (25)$$

Proof: See Appendix II.

From *Proposition 2*, λ^{ii^c} gets closer to or remains the same distance from λ^{ii^c*} when i^c updates λ^{ii^c}.

Proposition 3: Let $\lambda(t)=(\lambda_{ij}(t))^T$ be the entire price vector at time t. Suppose that the total asynchronism assumption holds. For any $t_0\ge 0$, if $\lambda(t_0)$ is not an optimal λ^*, then there exists $t_2 > t_0$ such that

$$\|\lambda^*-\lambda(t_2)\|<\|\lambda^*-\lambda(t_0)\|. \quad (26)$$

Proof: See Appendix III.

Proposition 3 implies that there exists at least one price component satisfying (19) to have a positive step size so that λ moves closer to λ^* until λ^* is reached. The above three propositions lead to the following theorem.

Theorem 1: Under the total asynchronism assumption, the distributed and asynchronous surrogate subgradient algorithm presented in (16)–(21) converges, i.e., λ converges to λ^*.

4) Step-Size Selection: To practically implement the method, one has to evaluate (19) and select a step size following (19)–(21). In view that λ^{ii^c*} is unknown, $(\lambda^{ii^c*}-\lambda^{ii^c}(t))^T g^{ii^c}(t)$ needs to be approximated by i or i^c. This is similar to the difficulties faced by most subgradient or surrogate subgradient methods. However, the approximation in the distributed and asynchronous surrogate subgradient method needs to be performed componentwise, without the knowledge of $q(\lambda)$ and with communication delays. To do this, a separate dual function $q_{ii^c}(\lambda^{ii^c},t)$ is first found, so that is its surrogate subgradient. Using the following property of surrogate dual presented in [15]

$$q_{ii^c}(\lambda^{ii^c*},t)-\tilde{q}_{ii^c}(\lambda^{ii^c}(t),t)\le(\lambda^{ii^c*}-\lambda^{ii^c}(t))^T g^{ii^c}(t) \quad (27)$$

where $\tilde{q}_{ii^c}(\lambda^{ii^c},t)$ is a surrogate dual of $q_{ii^c}(\lambda^{ii^c},t)$, then $(\lambda^{ii^c*}-\lambda^{ii^c}(t))^T g^{ii^c}(t)$ can be approximated by $q_{ii^c}(\lambda^{ii^c*},t)-\tilde{q}^{ii^c}(\lambda^{ii^c}(t),t)$.

To obtain $q_{ii^c}(\lambda^{ii^c},t)$, a new two-organization problem focusing on i or i^c will be formulated with delayed price information from customers and suppliers. It will be shown that the subproblems of this new problem are the same as the original subproblems for i and i^c; and $g^{ii^c}(t)$ is a surrogate subgradient of $q_{ii^c}(\lambda^{ii^c},t)$, the associated dual function. The value $(\lambda^{ii^c*}-\lambda^{ii^c}(t))^T g^{ii^c}(t)$ can then be approximated by using (27), where $q_{ii^c}(\lambda^{ii^c*},t)$ and $\tilde{q}^{ii^c}(\lambda^{ii^c}(t),t)$ are computed or approximated based on subproblem solutions of i and i^c.

Two-Organization Problem: Suppose that λ^{ii^c} is updated by i at time t. A snapshot two-organization problem for i and i^c is defined by freezing the prices other than λ^{ii^c} held by i and i^c

$$\min J_{ii^c}(t), \quad J_{ii^c}(t)\equiv\hat{L}_i(t)+\hat{L}_{i^c}(t) \quad (28)$$

where $\hat{L}_i(t)$ is similar to L_i in (12) except that all the terms involving λ^{ii^c} are excluded, and other prices $(\lambda_{ij},\lambda_{i^sj^s}\notin\lambda^{ii^c})$ take delayed values used in updating λ^{ii^c}, i.e., $\lambda_{ij}(\tau_i^{ij}(t))$ and $\lambda_{i^sj^s}(\tau_i^{i^sj^s}(t))$. Similarly, $\hat{L}_{i^c}(t)$ is defined for i_c.

Problem (28) is subject to the interorganization precedence constraints between i and i^c and constraints internal to i and i^c. To solve this problem, the interorganization constraints between i and i^c are relaxed by using λ^{ii^c}, and the relaxed problem is given by

$$\min L_{ii^c}(t), \quad L_{ii^c}(t)\equiv L_i(t)+L_{i^c}(t) \quad (29)$$

where L_i is given by (12) and L_{i^c} is similarly defined. With a token introduced, it is clear that this relaxed problem can be decomposed into two subproblems, one for i and the other for i^c, and these two subproblems are the same as the original subproblems for i or i^c. The dual function of $J_{ii^c}(t)$ is then defined as

$$q_{ii^c}\left(\lambda^{ii^c},t\right)=\min L_{ii^c}(t)=\min\left[L_i(t)+L_{i^c}(t)\right] \quad (30)$$

with the surrogate dual denoted as $\tilde{q}_{ii^c}(\lambda^{ii^c},t)$. In view that the subproblems related to (30) are the same as the original ones, $g^{ii^c}(t)$ is a surrogate subgradient of $q_{ii^c}(\lambda^{ii^c},t)$.

Step-Size Selection: To approximate $(\lambda^{ii^c*}-\lambda^{ii^c}(t))^T g^{ii^c}(t)$ based on (27), consider $q_{ii^c}(\lambda^{ii^c*},t)$ and $\tilde{q}_{ii^c}(\lambda^{ii^c}(t),t)$. The term $\tilde{q}^{ii^c}(\lambda^{ii^c}(t),t)$ can be evaluated by i based on subproblem costs of i and i^c, and $q_{ii^c}(\lambda^{ii^c*},t)$ can be approximated by an upper bound, e.g., a feasible cost obtained by using heuristics for the two-organization problem (28) (additional communication between i and i^c may be needed for the heuristics). Let ϕ denote the approximate value of the left-hand side of (27), i.e.

$$\phi\approx q_{ii^c}\left(\lambda^{ii^c*},t\right)-\tilde{q}^{ii^c}\left(\lambda^{ii^c}(t),t\right) \quad (31)$$

then the step sizes can be obtained as

$$\alpha^{ii^c}(t)=\gamma\frac{\phi}{\|g^{ii^c}(t)\|^2}, \qquad \text{if } \phi\ge 0 \quad (32)$$

$$\alpha^{ii^c}(t)=0, \qquad \text{if } \phi<0 \quad (33)$$

where γ is a parameter satisfying $0<\gamma<2$.

5) Feasible Solution and Rescheduling: At the convergence or near-convergence of the interorganization price-updating process, tokens are terminated and prices are fixed. Tentative orders then become contracted orders, with their tentative delivery dates becoming promised delivery dates, requested delivery dates becoming due dates and extra step penalty functions for missing order due dates added to cost functions. Note that subproblem solutions, when put together, may not represent a feasible schedule across the network since interorganization constraints have been relaxed. To obtain a feasible solution, heuristics based on mutually agreed rules and subproblem solutions are used. In the heuristics, if the tentative delivery date of the selected supplier for a particular request for bids is larger than the requested delivery date, the customer organization adjusts its schedule by using the heuristics of [14], so that the particular order starts later than the tentative delivery date plus possibly a required timeout. Similar is done if the promised delivery date for a contracted order is larger than the requested delivery date. This process is applied across the network from upstream suppliers to the downstream customers. Since dual costs are lower bounds to the optimal cost, the quality of a schedule for a snapshot problem can be quantitatively evaluated by comparing its cost with its dual cost. In view that prices are updated in a distributed and asynchronous manner, however, the true dual cost is not readily available. To obtain a dual cost, the subproblems of all organizations are solved once, based on the same set of prices at convergence. The evaluation should, nevertheless, be carefully done, since the objective function has been changed when tentative orders are converted to contracted orders.

Upon the arrivals of new orders or upon unexpected disruptions, rescheduling is triggered, new tokens created, and another cycle of the price-updating process is carried out. In view that the prices can be initialized at their previous values, convergence should be fast. The process for a three-tier network consisting of a manufacturer and its customers and suppliers is summarized in Fig. 7.

With prices dynamically updated and schedules adjusted, this approach tries to fulfill existing commitments while maintaining agility to take on new orders. Since prices represent the sensitivity of the overall cost with respect to interorganization precedence violations, they are marginal values of a time unit for the early or late delivery of orders. These prices thus reflect the pressure on order delivery and provide quantitative and dynamic answers to the old question, "Time is money, but how much?"

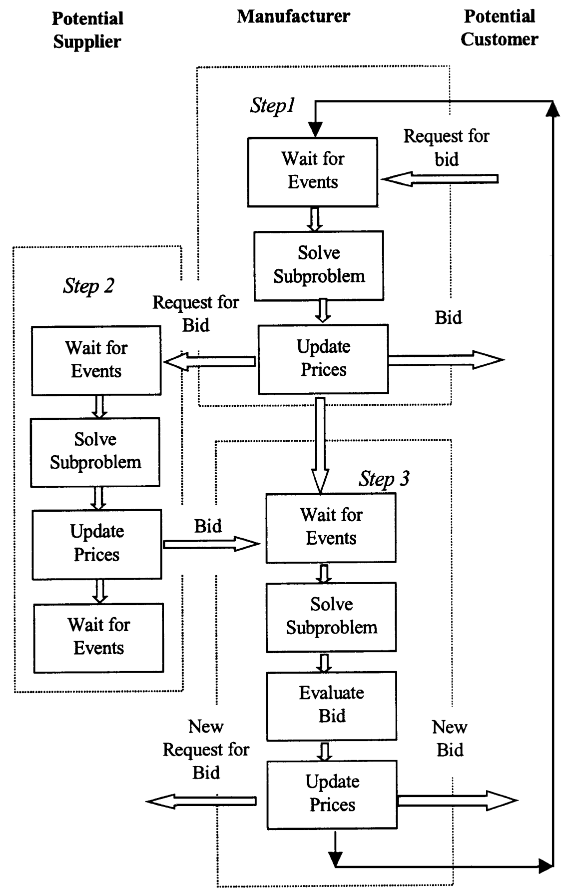

Fig. 7. Coordination process.

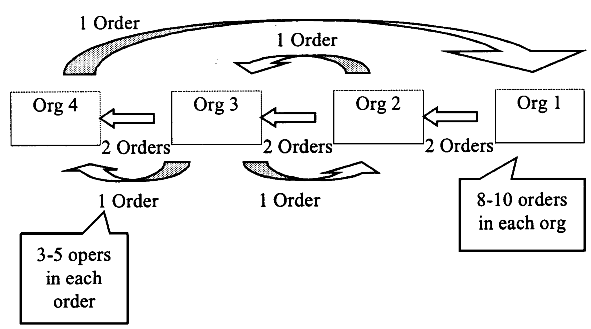

Fig. 8. Supply network for *Example 1*.

V. Implementation and Testing Results

The method presented above has been implemented on a network of PCs. Each organization is associated with a dedicated computer and equipped with the Lagrangian relaxation-based scheduling method of [14] in C++. The communications across the network is implemented in Java. Three examples are presented below.

Example 1: This example is to illustrate the convergence of the distributed and asynchronous price-updating method. Consider a four-tier supply network as shown in Fig. 8. There are four organizations with a total of ten "network orders," each involving two or three organizations. In addition, each organization may have its own "local orders" with no suppliers or customers within the formulation. The total number of orders per organization ranges from eight to ten, each containing three to five operations. In the implementation, two Pentium III 700-MHz and two Pentium II 400-MHz PCs are used, one per organization.

The convergence of interorganization prices is shown in Fig. 9. In view that multiple PCs of different speeds were used and communications were not optimized, 180 interorganization price iterations took 3.4 min.

Example 2: This example is to demonstrate that prices serve as coordination signals in a dynamic setting and to compare the performance of our approach with that of the "contract net only" approach. In the example, there are three organizations as shown in Fig. 10.

At the beginning, Organization 1 has four orders and one of them, Order (1, 1), requires a component part to be provided by either Organization 2 or 3. This order is, thus, associated with two prices, one for each potential supplier. Organization 1 later

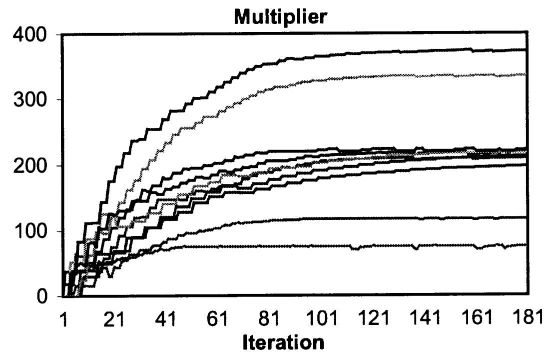

Fig. 9. Convergence of prices for *Example 1*.

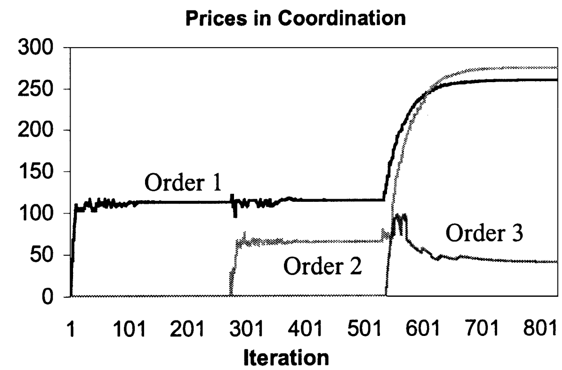

Fig. 11. Changing of prices over time.

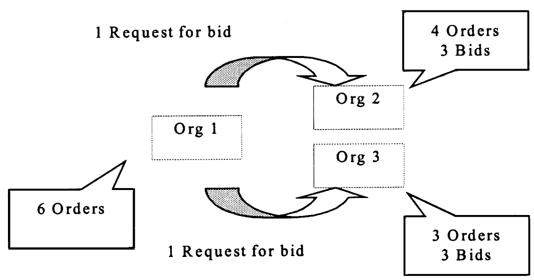

Fig. 10. Supply network for *Example 2*.

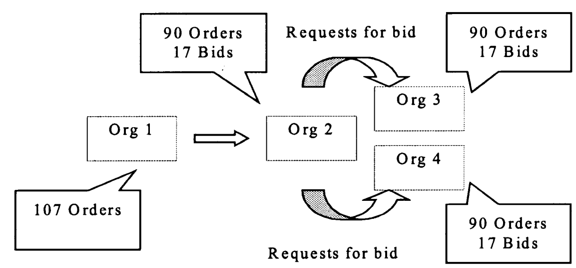

Fig. 12. Supply network for *Example 3*.

receives two new orders, Orders (1, 2) and (1, 3), at different times. Each of them also requires a component part to be provided by either Organization 2 or 3. Each order, in turn, contains three to five operations. Rescheduling is performed after the arrival of a new order.

The interorganization prices associated with selected suppliers for the three network orders are plotted against the number of iterations in Fig. 11. It can be seen that the prices converge, although they may change to different values upon the arrivals of new orders. They represent the costs for delaying delivery by one time unit and reflect the pressure on order deliveries. The time to obtain a schedule after a new order arrives is about 2.8 min.

The "contract net only" approach was also implemented within the same framework by fixing interorganization prices to zero and without rescheduling contracted orders. To compare the performance of the two approaches on an equal footing, all costs are recalculated based on the original requested delivery dates and the realized delivery dates without step penalties. The overall cost for the price-based approach is 16 323, whereas the cost for the contract net only approach is 18 946, 12% higher than that of the price-based approach.

Example 3: This example is to demonstrate the performance of our approach for a larger problem and to compare it with that of the "contract net only" approach. There are four organizations as shown in Fig. 12. At the beginning, Organization 1 has 99 orders and nine of them require component parts to be provided by Organization 2. To produce these component parts, Organization 2 needs to select Organization 3 or 4 as its supplier for each bid. Organization 1 later receives eight new orders at two different times, four at each time. Each of them also requires a component part to be provided by Organization 2, which in turn needs Organization 3 or 4. Each order may contain three to five operations and rescheduling is performed after the arrival of a new batch of orders.

The overall cost for the price-based approach is 25 834 770, whereas the cost for the "contract net only" approach is 28 529 142, which is 10% higher than that of the price-based approach. The computation time to obtain a schedule is about 48 min, where a significant amount of time is used to set up data structures and for I/O operations and the time to solve an organization subproblem is about 28–40 s.

VI. Conclusion

In this paper, a novel approach combining mathematical optimization and contract net protocol is presented for make-to-order supply network coordination based on a decentralized model. Activity coordination is carried out in a distributed and asynchronous manner without accessing others' private information or intruding on their decision-making authority, and the convergence of the price-updating method is established. With prices dynamically updated and schedules adjusted, this approach optimizes fulfilling existing commitments while maintaining agility to take on new orders. Achieving this balance may be a key to survival and profitability in today's time-based competition. In addition, since prices represent the sensitivity of the overall cost with respect to interorganization precedence violations, they are marginal values of a time unit for the early or late delivery of orders. These prices thus reflect the pressure on the delivery of orders with wide managerial implications and provide quantitative and dynamic answers to the perennial question, "Time is money, but how much?"

APPENDIX I

Proof of Proposition 1

If (19) is satisfied, i.e.,

$$\left(\lambda^{ii^c*} - \lambda^{ii^c}(t)\right)^T g^{ii^c}(t) > 0$$

we have

$$\begin{aligned}\left\|\lambda^{ii^c*} - \lambda^{ii^c}(t')\right\|^2 &= \left\|\lambda^{ii^c*} - \lambda^{ii^c}(t) + \lambda^{ii^c}(t) - \lambda^{ii^c}(t')\right\|^2 \\ &= \left\|\lambda^{ii^c*} - \lambda^{ii^c}(t)\right\|^2 \\ &\quad - 2\alpha^{ii^c}(t)\left(\lambda^{ii^c*} - \lambda^{ii^c}(t)\right)^T g^{ii^c}(t) \\ &\quad + \alpha^{ii^c}(t)^2 \left\|g^{ii^c}(t)\right\|^2 \\ &= \left\|\lambda^{ii^c*} - \lambda^{ii^c}(t)\right\|^2 \\ &\quad - \alpha^{ii^c}(t)\left\|g^{ii^c}(t)\right\|^2 \\ &\quad \cdot \left[\frac{2\left(\lambda^{ii^c*} - \lambda^{ii^c}(t)\right)^T g^{ii^c}(t)}{\|g^{ii^c}(t)\|^2} - \alpha^{ii^c}(t)\right].\end{aligned}$$

Considering the range of step size in (20), we have $\|\lambda^{ii^c*} - \lambda^{ii^c}(t')\|^2 < \|\lambda^{ii^c*} - \lambda^{ii^c}(t)\|^2$. Therefore, $\|\lambda^{ii^c*} - \lambda^{ii^c}(t')\| < \|\lambda^{ii^c*} - \lambda^{ii^c}(t)\|$.

If (19) is not satisfied, then $\alpha^{ii^c}(t) = 0$ and $\lambda^{ii^c}(t') = \lambda^{ii^c}(t)$. Therefore

$$\left\|\lambda^{ii^c*} - \lambda(t')\right\| = \left\|\lambda^{ii^c*} - \lambda^{ii^c}(t)\right\|.$$

Consequently

$$\|\lambda^* - \lambda(t')\| \le \|\lambda^* - \lambda(t)\|.$$

Q.E.D.

APPENDIX II

Proof of Proposition 2

From the procedure of price updating, we know that Organization i^c updates λ^{ii^c} from $\lambda^{ii^c}(t_1)$ to $\lambda^{ii^c}(t_2^+)$ and then sends $\lambda^{ii^c}(t_2^+)$ to Organization i. As the updating follows (19)–(21), we have

$$\left\|\lambda^{ii^c*} - \lambda^{ii^c}\left(t_2^+\right)\right\| \le \left\|\lambda^{ii^c*} - \lambda^{ii^c}(t_1)\right\|$$

based on *Proposition 1*. Q.E.D.

APPENDIX III

Proof of Proposition 3

Suppose that $\|\lambda^* - \lambda(t)\| = \|\lambda^* - \lambda(t_0)\|$ for all $t > t_0$. Then from *Proposition 1*, step sizes for all price components should be zero for all $t > t_0$ because a positive step size will lead $\lambda(t)$ closer to λ^*. Since all old price information will eventually be purged from organizations under the total asynchronism assumption, there exists a $t_1(> t_0)$ such that the price component held by each organization at t_1 takes value from $\lambda(t_0)(= \lambda(t_1))$. Consequently, a surrogate subgradient becomes a subgradient at time t_1 and according to the definition of subgradient [4, p. 711]

$$\begin{aligned}(\lambda^* - \lambda(t_1))^T g(t_1) &= \sum_{i,j}\left(\lambda_{ij}^* - \lambda_{ij}(t_1)\right) g_{ij}(t_1) \\ &\ge q(\lambda^*) - q(\lambda(t_1)).\end{aligned}$$

Since $\lambda(t_1)$ is not an optimal solution to the dual problem, $q(\lambda^*) - q(\lambda(t_1)) > 0$. As λ^{ii^c}, $\forall i, i^c$, is a part of λ, it follows that there is at least one $\lambda^{ii^c}(t_1)$ that satisfies

$$\left(\lambda^{ii^c*} - \lambda^{ii^c}(t_1)\right)^T g^{ii^c}(t_1) > 0.$$

According to *Proposition 1*, there exists a $t_2(> t_1)$ such that

$$\|\lambda^* - \lambda(t_2)\| < \|\lambda^* - \lambda(t_1)\| = \|\lambda^* - \lambda(t_0)\|$$

which contradicts the original assumption that $\|\lambda^* - \lambda(t)\| = \|\lambda^* - \lambda(t_0)\|$ for all $t > t_0$. Q.E.D.

REFERENCES

[1] D. L. Anderson and H. L. Lee, “Synchronized supply chains: The new frontier,” *Achieving Supply Chain Excellence Through Technology (ASCET)*, vol. II, 2000.
[2] M. S. Fox, M. Barbuceanu, and R. Teigen, “Agent-oriented supply-chain management,” *Int. J. Flexible Manuf. Syst.*, vol. 12, no. 2–3, pp. 165–188, 2000.
[3] J. C. Beck and M. S. Fox, “Supply chain coordination via mediated constraint relaxation,” in *Proc. 1st Canadian Workshop Distributed Artificial Intelligence*, Banff, AB, Canada, May 15, 1994.
[4] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, 1999.
[5] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[6] L. Gou, P. B. Luh, and Y. Kyoya, “Holonic manufacturing scheduling: Architecture, cooperation mechanism and implementation,” *Comput. Ind.—Intell. Manuf. Syst.*, vol. 37, pp. 213–231, 1998.
[7] K. J. Lee, Y. S. Chang, and J. K. Lee, “Time-bound negotiation framework for electronic commerce agents,” *Decision Supporting Syst.*, vol. 28, pp. 319–331, 2000.
[8] N. M. Sadeh, D. W. Hildum, D. Kjenstad, and A. Tseng, “MASCOT: An agent-based architecture for dynamic supply chain creation and coordination in the internet economy,” *Prod. Planning Contr.*, vol. 12, no. 3, pp. 212–222, 2001.
[9] W. Shen, F. Maturana, and D. H. Norrie, “MetaMorph II: An agent-based architecture for distributed intelligent design and manufacturing,” *J. Intell. Manuf.—Distributed Manuf. Syst.*, vol. 11, no. 3, pp. 237–251, June 2000.
[10] R. G. Smith, “The contract net protocol: High-level communication and control in the distributed problem solver,” *IEEE Trans. Comput.*, vol. C-29, pp. 1104–1113, Dec. 1980.
[11] J. M. Swaminathan, S. F. Smith, and N. M. Sadeh, “Modeling supply chain dynamics: A multi-agent approach,” *Decision Sci.*, vol. 29, no. 3, pp. 607–632, 1998.
[12] J. Wang, P. B. Luh, X. Zhao, and J. Wang, “An optimization-based algorithm for job shop scheduling,” *SADHANA, J. Indian Academy of Sci.—Competitive Manuf. Syst.*, pt. 2, vol. 22, pp. 241–256, Apr. 1997.
[13] S. D. Wu and H. Golbasi, “Manufacturing planning over alternative facilities: Modeling, analysis and algorithms,” in *Supply Chain Management: Models, Applications and Research Directions*, J. Geunes, P. M. Pardalos, and H. E. Romeijn, Eds. Dordrecht, The Netherlands: Kluwer, 2002.
[14] Y. Zhang, P. B. Luh, K. Narimatsu, T. Moriya, T. Shimada, and L. Fang, “A macro-level scheduling method using Lagrangian relaxation,” *IEEE Trans. Robot. Automat.*, vol. 17, Feb. 2001.
[15] X. Zhao, P. B. Luh, and J. Wang, “Surrogate gradient algorithms for Lagarangian relaxation,” *J. Optim. Theory Applicat.*, vol. 100, no. 3, pp. 699–712, 1999.

Peter B. Luh (S'77–M'80–SM'91–F'95) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in 1973, the M.S. degree in aeronautics and astronautics engineering from the Massachusetts Institute of Technology, Cambridge, in 1977, and the Ph.D. degree in applied mathematics from Harvard University, Cambridge, MA, in 1980.

Since 1980, he has been with the University of Connecticut, Storrs. Currently, he is the Southern New England Telephone (SNET) Endowed Professor of Communications and Information Technologies in the Department of Electrical and Computer Engineering, and the Director of the Booth Center for Computer Applications and Research. He is interested in manufacturing planning, scheduling, and supply chain coordination, and has developed near-optimal and computationally efficient methods to improve on-time delivery of products and reduce work-in-progress inventory. He is also interested in schedule/bid optimization and load/price forecasting for power systems and has developed near-optimal and computationally efficient unit commitment, hydro-thermal coordination, transaction/bidding and load/price forecasting methods to minimize total cost or maximize the profit. He has been a Visiting Professor for National Taiwan University and a Visiting Researcher for Pratt & Whitney, Toshiba Corporation, and United Technologies Research Center. He owns one U.S. patent.

Dr. Luh received three Best Paper Awards. He is the Editor-in-Chief of IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, and an Associate Editor of *IIE Transactions on Design and Manufacturing*, *Discrete Event Dynamic Systems* and *International Journal of Intelligent Control and Systems*. He was also an Editor of *IEEE Robotics and Automation Magazine* (1996–1998) and an Associate Editor for IEEE TRANSACTIONS ON AUTOMATIC CONTROL (1989–1991).

Ming Ni received the B.S. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1993, and the M.S. degree in electrical engineering from Sichuan University, Chengdu, China, in 1996. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering, University of Connecticut, Storrs.

His research interests include optimization theory and algorithms and their applications to manufacturing and control systems.

Haoxun Chen received the B.Sc. degree in applied mathematics from Fudan University, Shanghai, China, in 1984, the M.S. degree in systems engineering from Shanghai Jiaotong University, Shanghai, China, in 1987, and the Ph.D. degree in systems engineering from Xi'an Jiaotong University, Xi'an, China, in 1990.

Since 1990, he was with Xi'an Jiaotong University, where he was an Associate Professor from 1993 to 1996. He visited INRIA-Lorraine, France, as a Visiting Professor, in 1994, the University of Magdeburg, Germany, as a Research Fellow of the Alexander von Humboldt Foundation in 1997 and 1998, and the University of Connecticut as a Research Assistant Professor in 1999 and 2000. He is currently an Associate Professor at the Technology University of Troyes, Troyes, France, teaching and doing research on logistics and supply chain management. His research interests include discrete event system control, production planning and scheduling and supply chain management. He has published more than 60 papers in technical journals and conference proceedings.

Dr. Chen received the 1998 King-Sun Fu Memorial Best Transactions Paper Award from the IEEE Robotics and Automation Society.

L. S. Thakur received the Eng.Sc.D. degree from Columbia University, New York, NY, in 1971, and the B.Sc. from Bombay University, Bombay, India, in 1963.

He was a Visiting Professor of Operations Research at Yale University School of Management, New Haven, CT, from 1985 to 1987. He is currently with the Department of Operations and Information Management, University of Connecticut, Storrs. He has served as a consultant to IBM Corporation as well as a senior consultant and Director of Management Science in a consulting organization. His primary research interests are in the development and applications of linear, nonlinear, and integer programming methods in Management Science and function approximations in optimization mathematics. He has published in many journals, including *Management Science*, *Mathematics of Operations Research*, *SIAM Journal on Applied Mathematics*, *SIAM Journal on Optimization and Control*, IEEE TRANSACTIONS, *Journal of Mathematical Analysis and Applications*, and *Naval Research Logistics*. He is an Associate Editor of *Naval Research Logistics*. His current research focuses on production scheduling, product design and facility location problems.

Dr. Thakur has received a National Science Foundation grant for his research on scheduling (with Dr. P. B. Luh). He is a member of the Operations Research Society of America, The Institute of Management Sciences, and the Mathematical Programming Society.