2005

Electronic Commerce Negotiation in a Supply Chain via Constraint Evaluation

R. Wilhelm

B. Chu

R. Sun

Electronic Commerce Negotiation in a Supply Chain via Constraint Evaluation

R. Wilhelm,[1] B. Chu,[2] and R. Sun[3]

1. Department of Mechanical Engineering, University of North Carolina, Charlotte, Charlotte, North Carolina, USA
2. Department of Software and Information Systems, University of North Carolina, Charlotte, Charlotte, North Carolina, USA
3. Oracle Corporation, Redwood City, California, USA

Abstract

Negotiation is of critical importance in e-commerce applications where the supply chain is dynamic and reconfiguring. In this research supply chain negotiation problems are addressed as constraint-satisfaction problems. In general each negotiation is handled in the largest scope possible to avoid the sub-optimality that can result from many local solutions. This global approach, however, must be balanced with time constraints that apply in e-commerce supply chain execution. In this paper, we describe a new approach for e-commerce supply chain negotiation via constraint evaluation. As well, results from prototype software, distributed across the internet, are discussed. Beyond the general formulation, we describe a more particular problem of kitted demand where a collection of purchased items must be acquired within the same time horizon. To address slow convergence a time-based penalty function is proposed.

Keywords: manufacturing, constraint, control

1. Introduction

A supply chain is an interrelated collection of enterprises that act to satisfy consumer demands by collecting, processing, and delivering materials and finished products. In earlier business cycles, these chains tended to be static because of the large costs of establishing the chains. New developments in electronic commerce promise to reduce the cost and complexity of these supply chains while promoting fast and flexible configuration [1]. From

these requirements, a new class of problems has emerged and is being addressed by cooperative frameworks for manufacturing networks [2] [3] [4] [5].

Supply chains range from very large and slowly changing organizations, such as those modeled by SCOR [6] and CPFR [7], to very shallow and quickly changing trading arrangements on the internet. In the former—for example, a heavyweight supply chain—a manufacturer typically commits to buy a fixed amount of product from a supplier over a fixed and long time horizon. Information sharing, typically through private networks, may allow small variations (e.g., 10%) around this negotiated level. In the latter, a lightweight supply chain, a company may choose to buy products on the open market in a way similar to consumers buying products from the web.

A hybrid approach, between these extremes, is commonly employed by companies of all sizes. In this hybrid approach, acceptable suppliers are qualified in advance but prices and capacity to deliver change over time. Discrete time horizons are used to plan production and, hence, purchases. Manufacturers that are generating demands for parts attempt to minimize their inventory while maximizing profit from delivered assemblies. Manufacturers that are satisfying demands attempt to maximize profit for shipped goods while satisfying capacity constraints and minimizing inventory costs.

A fundamental problem in the creation and management of supply chains is the timely negotiation of purchase attributes such as price, quantity, and delivery date for purchased items. While this negotiation can be broadly thought of as a type of auction [8] [9] [10], constraints due to lot size, inventory, shipping, delivery dates, and capacity pose problems that are best solved with a more global optimization strategy [11].

In this paper we address these negotiation problems as constraint-satisfaction problems. Each problem is solved in the largest scope possible to avoid the suboptimality of local solutions. This global approach, however, must be balanced with time constraints that always apply in ecommerce supply chain purchasing and execution. As well, results from prototype software, distributed across the internet, are discussed. Beyond the general formulation, we describe a more particular problem of kitted demand where a collection of purchased items must be acquired within the same time horizon. To address slow convergence, a time-based penalty function is proposed. We conclude with a summary of results.

2. Supply Chain Negotiation

We consider a common problem of the hybrid model: supplier selection and negotiation. Owing to the recursive structure of a supply chain, without loss of generality, we focus on a manufacturer (buyer) and its suppliers (sellers).

We assume that the buyer has selected a set of sellers (which change over time) with negotiated standard purchase agreements. The buyer also may have a ranking system of the sellers regarding quality and reliability of delivery. Our objective is to use a multi-agent system [12] [13] to aid or automate buyer selection and negotiation. This is important for the following reasons.

This is a difficult task for people to carry out when one is confronted with even a modest number of choices (e.g., 10 suppliers each with 10 options). With the advent of Electronic Commerce, a buyer will have many more alternative sellers on a global scale. Agents can

be used to carry out routine selections and negotiations so that people are involved only in exceptional cases. This capability also enables buyers to examine more options and find the most cost-effective set of sellers. Agents can also monitor the execution of the committed orders. They can inform each other early about potential changes and make real-time adjustments. As a result the buyer and sellers can reduce their inventory levels.

This selection/negotiation process is illustrated in Figure 1.

- A buyer sends requests for proposal (RFP) of quotes to a set of sellers. From several quotes, the buyer chooses which seller(s) to order from. The buyer then sends order proposals to selected sellers.

- Buyers sometimes combine demands from different time periods into a single order to take advantage of volume prices offered by a supplier.

- The seller may not be able to satisfy a particular order at a given time. In this case, the seller may counteroffer. After considering actual scheduled demands, the buyer may reply and accept the counteroffer.
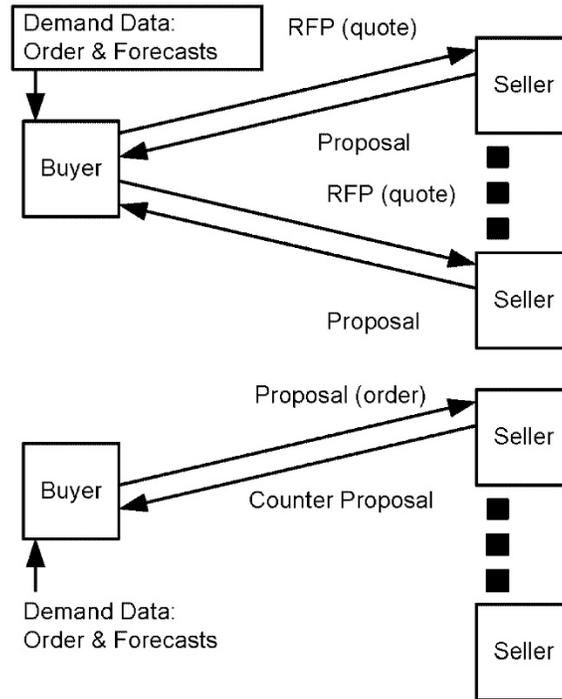

Figure 1. Supplier Selection and Negotiation Process

3. Supply Chain Negotiation as a Constraint Satisfaction Problem

A Constraint Satisfaction Problem (CSP) is defined by a set of variables, domains for each variable, and a set of constraints [14]. In our supply chain negotiation problem, the varia-

bles are times and quantities for demands that must be satisfied while the constraints describe relationships between these time and quantity variables. The objective function is the total cost, which includes product cost, freight, inventory, and other costs.

3.1 Variables

Supplier
Supplier ID s ∈ [Allowable Labels, i.e., Company A]

Supply Lead Time
M_s^j is the largest lead time for product j from supply s.

Supplier Transportation Time
N_s is the least transportation lead-time for suppliers.

Order
A vector of variables that describes demand for each order/suborder. An order is defined of product type j at due date i: O_i^j which may generate suborders.

Release Date
r is the release date for $SO_i^j, 0 \leq r \leq i - N_s - M_s^j$

Shipping Date
b is the shipping date for $SO_i^j, r + M_s^j \leq b \leq i - N_s$

Suborder
Suborder $SO_{i,s,r}^j$ includes:
Quantity $Q_{i,s,r,b}^j \in [0, 1, 2, \ldots, \text{Quantity of } O_i^j]$
Transportation lead time $P_{i,s,r,b}^j \in$ [Allowable Lead times for supplier s]

3.2 Constraints
For each suborder SO_i^j, s, r, b, the following constraints apply when the associated quantity $Q_{i,s,r,b}^j$ is positive.

Quantity Conservation
$- Q_{i,s,r,b}^l = \text{Quantity of } O_i^l$
  all s, r, b

Temporal Consistency
for (b − r) ∈ [shipping lead time set of supplier s]
$(\Sigma Q_{i,s,r,b}^l \leq (\text{quantity limit of supplier s}))$
  all b, i

Shipping Feasibility

$(b + P_{i,s\ r,b}^l) \leq i$

3.3 Objective

The objective is divided into two stages. First it is necessary to verify whether there are feasible solutions with respect to $Q_{i,s,r,b}^j$ and $P_{i,s,r}^j$. In the second stage, if a feasible solution exists, a minimal cost solution is sought. A relaxation is sought if there are no feasible solutions.

4. Negotiation Steps

A buyer agent takes the following steps in order.

- Open the bid by sending RFPs to prospective sellers. Sellers will respond with proposals.

- In response to proposal messages, the buyer agent evaluates proposals and generates a purchase plan.

- Upon approval (of human decision makers), the buyer agent negotiates with prospective sellers. Buyer agents and seller agents exchange counteroffers and present options to human decision makers.

A seller agent takes the following steps in order.

- Respond to RPFs from prospective buyers.

- Evaluate (proposed) orders from buyers.

- The seller may also counterpropose and present options to human decision makers.

5. Negotiation Strategies

The negotiating process is described via constraint logic programming (CLP) predicates [15]. This explanation is restricted to the case of one product but can be easily generalized to multiple products.

During each negotiation step, multiple strategies are followed. For example, the seller agent may split orders into multiple shipments to offer counterproposals. An alternative strategy would be to propose that a set of orders be delayed. A default strategy is associated with each step. If this strategy fails (e.g., the counteroffer was rejected), a human decision maker may select one of the alternative strategies. In the following section, we discuss some example strategies that we have implemented.

5.1 Strategies for opening bid/response to RFP

A buyer agent uses this strategy in response to a "start negotiation" event. The inputs to this event are demand data described earlier. The agent will solicit ordering rules from a set of suppliers. A typical seller's response may be:

Supplier s1 can supply as many as 500 pieces at the unit price of $20 with a lead time no greater than 4 weeks.

5.2 Strategies for generating purchase plan (proposal)

An order is represented as a predicate Order (S, Q, D, P) where S is the supplier, Q is the quantity ordered, D is the due date, and P is the unit price. Conditioned quotes can be represented via the following rule that means that seller s1 can supply 500 pieces at the unit price of $20 given a lead time of 4 weeks.

Order (s1, Q, D, $20) :- Q<=500, D-today>=4weeks.

To generate a purchase plan we solve the CLP problem of finding a set of orders that satisfy a total quantity before a given due date and under a given price cap. This is represented as a predicate OrderProposals (SL, QL, DL, PL, TQ, DD, PC) where SL is a list of suppliers, QL is a list of quantities to be ordered, DL is a list of due dates, PL is a list of unit prices, TQ is the total quantity needed, DD is the due date, and PC is the price cap. For example, the problem of generating a purchase plan for 1,000 pieces by May 15 with total price not exceeding $20,000 can be viewed as solving for:

OrderProposals (SL, QL, DL, PL, 1,000, May15, $20,000).

The answer to this question may be:

OrderProposals ([s1 ,s2], [500, 500], [May13, May14], [20, 18], 1,000, May15, $20,000)

It represents the following orders:

Order (s1, 500, May13, $20) and Order (s2, 500, May14, $18)

Purchase plan generation may be governed by the following rules:

OrderProposals ([S], [Q], [D], [P], TQ, DD, PC) :-
Order (S, Q, D, P), Q<=TQ, D<=DD, P*Q<=PC.

That is, if we can find a single supplier who can satisfy all the requirements, then this supplier will be selected. Otherwise, we must split the total order among more than one supplier. This strategy can be represented as:

OrderProposals ([S|RS], [Q|RQ], [D|RD], [P|RP], TQ, DD, PC)
:-   Order (S, Q, D, P),   Q<TQ,   D<=DD,   P*Q<PC,
OrderProposals (RS, RQ, RD, RP, TQ-Q, DD, PC-P*Q).

Here the notation [H|T] indicates list concatenation with H as the head of the list and T as the rest of the list.

The problem presented here represents a minimum case and many embellishments are used in practical cases. While, the CLP/CSP approach does not offer the most efficient solution approach for this minimum case, it is used to allow for the expression of additional constraints that occur in practical applications.

5.3 Strategies for order proposal evaluation

When a seller receives an order proposal from a buyer, the seller checks manufacturing capacity to see if the order can be met. CSP/CLP is as an effective tool to solve scheduling and capacity estimation problems [16]. Assume that predicate Capacity(TQ,D) solves the scheduling/capacity problem with a CSP. Then, a seller is able to deliver a total of TQ by date D. The problem of determining whether an order is feasible can be expressed as the rule:

FeasibleOrder (Q,D) :- Capacity (TQ,D), Q<=TQ.

If the order is infeasible, the seller may counterpropose and negotiate with the buyer. The process of generating counterproposals and negotiation is modeled as contrast relaxation. In our approach, this relaxation is represented as a library of predefined negotiation strategies and these strategies are invoked at appropriate times.

5.4 Strategies for counterproposal generation

In this approach, an order is split into multiple shipments with the first shipment on or very close to the date requested by the buyer.

Predicate Split (OriginalOrder(Q,D), Order1 (Q1,D1), Order2(Q2,D2)) means to split an original offer OriginalOrder(Q,D) into two orders Order1(Q1,D1) and Order2(Q2,D2).

The seller must first evaluate whether this split is feasible. This can be achieved by solving:

Split(OriginalOrder(Q,D), Order1 (Q1 ,D1), Order2(Q2,D2)) :- Q1 + Q2 = Q, D1<=D, D2>D, FeasibleOrder(Q1,D1), FeasibleOrder(Q2, D2).

If two-way splitting is infeasible, three-way splitting may be tried, but further recursive splitting is not pursued. Alternatively, a partial shipment may counteroffered.

5.5 Strategies for order negotiation

During order negotiation, the buyer negotiates with sellers. Messages are sent one at a time, though a more parallel strategy is possible. For each proposal, the seller may accept, reject, or counterpropose. If the seller accepts, demand information is updated. If the seller rejects, the generate order proposals strategy may be employed again while ignoring this seller. If the seller counterproposes, the buyer assesses the counterproposal. If acceptable,

it is treated as an accepted proposal; otherwise, it is treated as a rejection with this particular supplier's rule (seller constraint) modified based on the counterproposal. The generate purchase plan step is then invoked again.

6. Implementation

We have prototyped a set of buyer and seller agents for a large industrial consortium developing solutions for integrated supply chains. The CSP/CLP model provides a convenient means for problem formulation and supports integration with legacy systems; a key requirement from our industrial partners. Figure 2 illustrates the current prototype. Supplier selection rules and demands are the initial input.
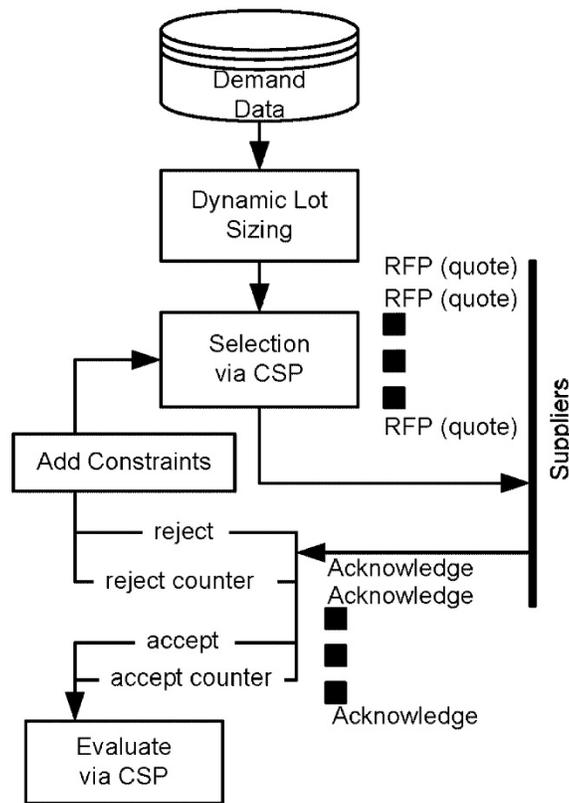

Figure 2. Supplier Selection Prototype Implementation

From the initial demands, a reduced set of order quantities are developed to minimize inventory and ordering costs via the dynamic lot sizing algorithm of Wagner-Whitin [17]. These order quantities are treated as CSP constraints to identify suppliers that can provide required quantities at minimum cost. RFPs are solicited from suppliers accordingly. At each RFP step, the supplier may accept, reject, or counterpropose. Counterproposals are

assessed using an augmented CSP where the counterproposal forms an additional constraint. In cases where RFPs are rejected or counterproposals fail, an additional CSP is solved to identify additional RFPs that should be directed to suppliers. The process continues until all demands have been satisfied. Our current prototype is implemented using ILOG Solver [18], a C++ constraint solver.

7. Discussion

The initial negotiation problem treats each demand independently and applies when there are a large variety of demands and these demands may interchangeably be used in a variety of orders. For high-value assemblies with low-volume shipments, demands must be satisfied simultaneously or not at all. We refer to this problem as the kitted demand problem since demands must be satisfied in kits that are structured according to functional requirements.

We have defined a weak ordering heuristic for kitted demand. All items in a kit are ordered by total cost, and high total cost demands are addressed first. As lower cost demands are handled, restrictions on shipment dates, shipment quantities, and inventory costs may be relaxed to insure that demands are satisfied. The initial negotiation problem ignored time constraints associated with communication of demands and proposals for supply. This assumption is valid when the time for an RFP transaction and response is very small compared to the time required for shipping or production. These transaction times often take seconds, while shipping and production times are days or weeks. For kitted demand, this assumption may not be valid, as the purchase cycle time could be hours or days.

As the transaction times grow larger, transactions may become invalid due to temporal inconsistency. For example, a promise of a shipment may not be finalized until after the promised date or time has passed. Recovery from this error is possible but may slow convergence of the negotiation. This is solved by using penalty functions within the CSP. Two penalties are modeled for this approach. First, the penalty of not satisfying all demand during a fixed time horizon is considered. Second, the penalty of a feasible but well suboptimal purchase is considered. While we typically use the CSP to identify the minimum cost solution, we can also use it to find the maximum cost of a feasible solution. This upper bound may be infinite but in practice it will be finite because suppliers promise shipments according to a finite time horizon. This upper bound serves as barrier for the penalty function.

8. Summary

Negotiation via constraint evaluation is defined to support supplier selection in a hybrid supply chain. Strategies are defined via constraint logic programming predicates for proposal request, proposal generation, proposal evaluation, counterproposal generation, and order negotiation. The approach has been implemented as a set of negotiating processes operating across the internet. As negotiation continues, the CSP formulation is augmented

to evaluate responses to RFPs and generate new solutions in the face of rejected RFPs. Beyond this general formulation, kitted demand is described where a collection of purchased items must be simultaneously acquired within the same time horizon. To address slow convergence, a time-based penalty function is proposed.

9. Acknowledgments

10. References

[1] Thomas, D. J., Griffin, P. M., 1996, Coordinated supply chain management, European Journal of Operational Research, 94/1:1–15.

[2] Wiendahl, H.-P., Lutz, S., 2000, Production in Networks, Annals of CIRP, 51/2:573–586.

[3] Tseng, M. M., Lei, M., Su, C. J., Merchant, M. E., 1997, A collaborative control system for mass customization manufacturing, Annals of CIRP, 46/1:373–376.

[4] Schuh, G., Bergholz, M., Westkamper, E., 2003, Collaborative Production on the Basis of Object Oriented Software Engineering Principles, Annals of CIRP, 52/1:393–396.

[5] Scholz-Reiter, B., Hohns, H., Hamann, T., 2004, Adaptive Control of Supply Chains: Building blocks and tools of an agent-based simulation framework, Annals of CIRP, 53/1:353–356.

[6] AMR Supply Chain Council: www.supply-chain.org.

[7] VICS CPFR website: www.cpfr.org

[8] Turban, E., 1997, Auctions and Bidding on the Internet: An Assessment, International Journal of Electronic Markets, 4/7:www.electronicmarkets.org

[9] McAfee, R. P., McMillan, J., 1987, Auctions and Bidding, Journal of Economic Literature, 25/2:699–738.

[10] Milgrom, P., 1989, Auctions and Bidding: A Primer, Journal of Economic Perspectives, 3/2:3–22.

[11] Bruccoleri, M., Lo Nigro, G., Federico, F., Noto La Diega, S., Perrone, G., 2003, Negotiation mechanisms for capacity allocation in distributed enterprises, Annals of CIRP, 52/1:397–402.

[12] Thannhuber, M., Tseng, M. M., Bullinger, H.-J., 2001, An Autopoietic Approach for Building Knowledge Management Systems in Manufacturing Enterprises, Annals of CIRP, 50/1:313–318.

[13] Ueda, K., Vaario, J., Takeshita, T., Hatono, I., 1999, An emergent synthetic approach to supply networks, Annals of CIRP, 48/1:377–380.

[14] Tsang, E., 1995, Foundations of Constraint Satisfaction, Academic Press Ltd., Burlington, USA.

[15] Marriott, K., Stuckey, P. J., 1998, Programming with Constraints: An Introduction, MIT Press, Boston, MA, USA.

[16] Beck, J. C., Davenport, A. J., Sitarski, E. M., Fox, M. S., 1997, Beyond Contention: Extending Texture-Based Scheduling Heuristics, Proc. of NCAI, 241–248.

[17] Wagner, H. M., Whitin, T. M., 1958, Dynamic Version of the Economic Lot Size Model, Management Science. 5/1:89–96.

[18] ILOG Optimization Suite, http://www.ilog.com