Aggressive Pricing to Exploit Market Niches in Supply Chains

Sabyasachi Saha and Sandip Sen, *University of Tulsa*

E-markets' growing popularity has given business-to-business trading an unprecedented boost. In supply chains connecting enterprises that trade services and goods,[1] an organization's success depends on its ability to maintain stable and profitable relationships with other supply chain participants. To stay ahead of the competition, organizations must deploy increasingly complex trading strategies.

Robust, opportunistic scheduling strategies can significantly improve suppliers' competitiveness by identifying market opportunities and strategically positioning and pricing available resources to exploit them.

Globalization has made supply chains enormously complex to maneuver. Geographically distributed suppliers, retailers, and manufacturers can be difficult to integrate into an efficient supply chain management system, and today's supply chain managers must move materials from geographically distributed suppliers to global manufacturing facilities. Decision support tools that accurately model supply chain dynamics can help managers achieve these goals.[2] Here we evaluate how predictive scheduling strategies and aggressive pricing schemes can help suppliers exploit market niches in B2B supply chains.

Efficient supply chain management seeks to

- Reduce waste by minimizing duplication and inefficient production
- Reduce inventory costs by efficiently planning future demand
- Reduce lead time by optimally allocating subtasks
- Improve product quality
- Develop strong, long-term partnerships
- Increase profitability

Multiagent systems researchers have modeled a supply chain as a decentralized network of software agents (see the "Related Work in Supply Chain Management" sidebar for more information). Structured conversations achieve effective coordination among agents in a supply chain.[3] A business entity's dependability determines the stability of its relationships. A manufacturing enterprise's net productivity, for example, depends on the performance of its supply chain's downstream components, which in turn depends on the supplier entities' performance index. Manufacturers' tasks differ in priority and deadline, and different priorities represent market demand dynamics.

We consider here a three-level supply chain of primary and secondary manufacturers and suppliers: Primary manufacturers reside in the upper level, and the second level contains secondary manufacturers linked with suppliers that reside in the lowest level. At each level is a group of competitive agents, and agent functionalities differ in each layer. We use a contracting framework to connect suppliers to manufacturers:[4]

1. Manufacturers announce contracts for tasks with given specifications (deadline and processing time).
2. Suppliers bid on these tasks with prices.
3. An auction allocates the contract to the supplier that fulfills all task constraints and offers the lowest price.

Competitive scheduling helps agents improve profitability by creating market niches.[4] Suppliers able to accommodate dynamically arriving tasks increase their profit. Here, we investigate how different pricing mechanisms affect suppliers' profitability under varying conditions, including task mix and group composition. A supplier that meets deadlines few others can handle will be able to demand more for its services. When competition is high, a supplier reduces its price to win the contract.

Our mechanism is especially suited for supply chain environments where many tasks are being contracted in an open environment based primarily on

Related Work in Supply Chain Management

The problem of supply chain management has recently drawn the attention of multiagent systems (MAS) researchers. In MAS, a supply chain is conceptualized as a group of collaborative autonomous software agents.[1] It's argued that managers can better coordinate and schedule processes by distributing the organization-wide business management system to autonomous problem-solving agents. This approach works better for geographically distributed organizations that can't manually control trading. Christopher Beck and Mark Fox have investigated supply chain coordination using partial constraint satisfaction by mediating agents.[2] Ye Chen and colleagues have proposed negotiation-based supply chain management, where the negotiating software agents establish a virtual supply chain when an order arrives.[3] MASCOT uses the blackboard architecture, a proven methodology for integrating multiple knowledge sources for problem solving, to implement a mixed-initiative agent-based architecture for supply chain planning and scheduling.[4]

Jayashankar Swaminathan and colleagues proposed a framework for efficient supply-chain formation.[5] Olivier Labarthe and colleagues presented a heterogeneous agent-based simulation to model supply chains.[6] MAS research has emphasized the emergence of the optimal supply chain configuration. William Walsh and colleagues demonstrated optimal dynamic task allocation in a supply chain using combinatorial auction.[7] Given a task composed of a group of subtasks, they showed the dynamic formation of the supply chain that produces maximum profit.

In 2003, the Trading Agent Competition (www.sics.se/tac) introduced the TAC SCM game to simulate the challenges of supporting dynamic supply chain practices. In this framework, which uses stochastic task generation, customers require computers with varying configurations. The agents bid for customers' contracts and also contract with different suppliers for raw materials. An agent manufactures the computers in its own factory when it has the required raw materials and then delivers the order to the customer. The agent's challenge is to design efficient strategies for factory scheduling, contracting with suppliers, competitive and profitable bidding to customers, inventory cost reduction, and reducing penalties for late delivery or order cancellation.

References

1. N.R. Jennings et al., "Autonomous Agents for Business Process Management," *Int'l J. Applied Artificial Intelligence*, vol. 14, no. 2, 2000, pp. 145–189.
2. J.C. Beck and M.S. Fox, "Supply Chain Coordination via Mediated Constraint Relaxation," *Proc. 1st Canadian Workshop Distributed Artificial Intelligence*, 1994; www.eil.utoronto.ca/iscm/papers/beckfox-cwdai94.pdf.
3. Y. Chen et al., "A Negotiation-Based Multi-Agent System for Supply Chain Management," *Proc. 3rd Conf. Autonomous Agents, Workshop on Agent-Based Decision Support for Managing the Internet-Enabled Supply Chain*, ACM Press, 1999, pp. 15–20.
4. N. Sadeh et al., "MASCOT: An Agent-Based Architecture for Coordinated Mixed-Initiative Supply Chain Planning and Scheduling," *Proc. 3rd Conf. Autonomous Agents, Workshop on Agent-Based Decision Support for Managing the Internet-Enabled Supply Chain*, ACM Press, 1999, pp. 133–138.
5. J.M. Swaminathan, S.F. Smith, and N.M. Sadeh, "Modeling Supply Chain Dynamics: A Multiagent Approach," *Decision Sciences*, vol. 29, no. 3, 1998, pp. 607–632.
6. O. Labarthe et al., "A Heterogeneous Multi-Agent Modeling for Distributed Simulation of Supply Chains," *Proc. 1st Int'l Conf. Industrial Applications of Holonic and Multi-Agent Systems* (HoloMAS 2003), LNCS 2744, Springer-Verlag, 2003, pp. 134–145.
7. W.E. Walsh and M.P. Wellman, "Decentralized Supply Chain Formation: A Market Protocol and Competitive Equilibrium Analysis," *J. Artificial Intelligence Research*, vol. 19, 2003, pp. 513–567.

price quotes. In particular, this means that tasks aren't contracted on the basis of long-term relationships between parties. We also assume a steady distribution of task types that lets suppliers form flexible schedules for accommodating highly profitable tasks. For this strategy to create market niches effectively, sufficient opportunities must exist for high-priority tasks to be contracted on short notice. Such short-notice contracting can occur in domains where schedules have to be quickly readjusted because suppliers fail to meet their deadlines or have machine or personnel problems.

Pricing strategies

Suppliers can use adaptive pricing schemes to exploit market opportunities. A supplier might benefit by hiking prices when it sees less competition for contracts and lowering them as competition increases. But if it doesn't have current information about competitors, such schemes could backfire. Although a supplier might benefit by aggressively exploiting sleeping market opportunities in the short term, it should take a more conservative approach in the face of competition.

We've defined three supplier pricing strategies. In all three, the supplier increases the bid price when it's winning contracts continuously. But a supplier can't bid more than the secondary manufacturer's reserve price. Suppose a supplier s gets invited to bid for a task t by a secondary manufacturer m, whose reservation price is R_m. The task length is $l(t)$, and the deadline $dl(t)$ determines priority $\mathcal{P}(t)$. We assume that a task with an immediate deadline is considered high priority (priority 1) and a task with a normal deadline is of ordinary priority (priority 0).

In a *linear strategy*, if the supplier won a previous contract for this type of task t, it will increase its bid by a constant α from its previous bid, provided the new bid won't exceed the reserve price of the secondary manufacturer it's bidding to. Here, for a fair comparison, we've used the same α for all suppliers. Similarly, if the supplier loses its previous contract for a similar task, it will reduce its bid by α from its previous bid, provided the bid isn't less than its own reserve price R_s. So, if the supplier bid b for the previous contract of the same length and priority as that of task t, then for task t it will bid

$$bid = \begin{cases} \min\left(b+\alpha, R_m\right) & \text{if it won last contract} \\ \max\left(b-\alpha, R_s\right) & \text{if it lost last contract} \end{cases} \quad (1)$$

A *defensive strategy* uses a more cautious calculation to increase (or decrease) the supplier's previous bid when it's winning (or losing) contracts. If the supplier won k contracts in a row for similar tasks, it will increase its bid by

$$incr = \begin{cases} \alpha - (k \times \delta) & \text{if } \alpha > k \times \delta \\ 0 & \text{otherwise} \end{cases}$$

provided the increased bid won't exceed the secondary manufacturer's reserve price. Here, δ is another constant parameter. This supplier will decrease its bid by the same amount—that is, *incr*—from its previous bid if it loses k contracts in a row for similar tasks, provided the decreased bid isn't less than its own reserve price. If the supplier's last bid for the same task type was b, we find the new bid from Equation 1 using $\alpha - (k \times \delta)$ instead of α.

A supplier following an *impatient strategy* increases or decreases its bid sharply after winning or losing contracts, respectively. If the supplier won k contracts in a row for similar tasks, it increases its bid from the previous bid by $\alpha + (k \times \delta)$, provided the new bid doesn't exceed the secondary manufacturer's reserve price. Similarly, if the supplier loses k contracts in a row, it decreases its bid by $\alpha + (k \times \delta)$, provided it won't be less than its own reserve price. If its last bid was b, we find the new bid from Equation 1 using $\alpha + (k \times \delta)$ instead of α.

We assume that, at the simulation's outset, each supplier has the same bid for a particular task type.

Scheduling strategies

Suppliers can use pricing schemes to exploit market niches if the arriving task mix presents such opportunities and if suppliers can create a flexible local schedule to accommodate profitable tasks. Smart, predictive scheduling and bidding decisions are key factors to producing market niche opportunities. Here we evaluate four scheduling strategies for suppliers. The goal is to allocate a task t of length $l(t)$ and deadline $dl(t)$, but each strategy has distinct motivations:

- The *first-fit* strategy searches forward from the current time and assigns t to the first empty interval of length $l(t)$ on the calendar. This produces a compact, front-loaded schedule.
- The *best-fit* strategy searches the entire feasible part of the schedule (between the current time and deadline $dl(t)$) and assigns t to an interval with minimal empty slots around it. This strategy produces clumped schedules, but the clumps need not be at the front of the schedule.
- The *worst-fit* strategy searches the entire feasible part of the schedule (between the current time and deadline $dl(t)$) and assigns t to an interval with maximum empty slots around it. This produces an evenly loaded schedule.
- Agents using the *expected utility-based* (EU) strategy, described in the next section, use knowledge of periodic patterns in the task arrival distribution to decide whether to bid on a new task. The first three strategies listed schedule tasks "greedily"—they bid for a task whenever they can schedule it. But a supplier might want to refrain from bidding, keeping the option open to bid for a more profitable task later. If the task arrival pattern isn't known a priori, the agent can learn it provided the pattern doesn't change drastically in a short time. Such opportunistic scheduling carries the risk that some production slots remain unused if the expected high-profit tasks never materialize. An agent using this algorithm bids for a task if its expected utility is positive. The task has a negative EU, and the agent won't bid on it if the agent's expectation that it can make more profit later overpowers the risk of not bidding now.

> Three strategies schedule tasks "greedily"—they bid for a task whenever they can schedule it. But a supplier might want to refrain from bidding, keeping the option open to bid for a more profitable task later.

Utility of task scheduling

For each arriving task $t \in \Upsilon$, where Υ is the set of all tasks, the utility for the task-scheduling agent is given by the function $u(l(t), \mathcal{P}(t))$, where $l(t)$ is the task's length and $\mathcal{P}(t)$ is its priority. Let $esd(t)$ and $dl(t)$ be the earliest start date and the deadline for processing the task, respectively. An agent calculates the number of empty slots, $fs(d)$, for each day $d \in D$, where D is the set of all days on calendar C between and including $est(t)$ and $dl(t)$. For each of these days, the agent generates two sets of task combinations, $\mathcal{T}^k_{fs(d)}$ and $\mathcal{T}^k_{fs(d)-l(t)}$, where

$$\mathcal{T}^k_\beta = \left\{ T | T \subset \Upsilon,\ \beta = \textstyle\sum_{t \in T} l(t), \right. \\ \left. \mathcal{P}(t) > k \forall t \in T \right\}$$

is the set of all task combinations in which the length of the tasks in each combination adds up to β and each task has a priority higher than k. We compare the utility of scheduling this task now and scheduling the remaining slots later with all possible ways of scheduling the currently empty slots. We choose to schedule the task now if the corresponding utility dominates all other ways of filling up the empty slots without scheduling the current task.

Let $n^T_{i,j}$ be the number of tasks in T of length i and priority j. $Pr(i, n^T_{i,j}, d, \mathcal{H})$ is the probability that at least $n^T_{i,j}$ number of tasks of length i and priority j will arrive later on day d, where $\mathcal{H}$ is the history of tasks that have already arrived. Given the task distribution, we calculate the probabilities $Pr(i, n^T_{i,j}, d, \mathcal{H})$ using the multinomial cumulative probability function.

The expected utility of scheduling the current task t on day d on calendar $\mathcal{C}$ given the history of task arrivals $\mathcal{H}$ is

$$EU(t, d, \mathcal{H}, \mathcal{C}) = u(l(t), \mathcal{P}(t)) + den(\mathcal{C}, est(t)) \\ \times (AU(t, d, \mathcal{H}, fs(d) - l(t)) \\ - AU(t, d, \mathcal{H}, fs(d))),$$

where

$$AU(t, d, \mathcal{H}, h) = \frac{1}{\left|\mathrm{T}^{\mathcal{P}(t)}_h\right|} \times \\ \sum_{T \in T^{\mathcal{P}(t)}_h} \sum_{k > \mathcal{P}(t)} \sum_{i=1}^{l_{\max}} Pr(i, n^T_{i,k}, d, \mathcal{H}) n^T_{i,k} u(i,k)$$

is the average expected utility of scheduling h hours on day d with tasks of higher priority than task t given the history of task arrivals $\mathcal{H}$. $l_{\max}$ is a task's maximum possible length, and the function *den* returns the calendar's density (the percentage of scheduled slots) up to a given date (to facilitate scheduling, we use $den(\mathcal{C}, 0) = 1$).

For a given calendar day, the EU expression adds the sum of the current task's utility to the difference of the average utility of all possible ways to fill the calendar with higher-priority tasks with or without the current task being scheduled. We schedule the task if it's positive for at least one of the days considered and the day chosen provides the maximum expected utility—that is, arg $\max_{d \in D} EU(t, d, \mathcal{H}, \mathcal{C})$.

Experimental framework

We designed our simulations to evaluate the different pricing schemes' relative effectiveness under varying environmental conditions. In particular, we expect to identify when smart scheduling strategies produce market niches that let suppliers use aggressive pricing to exploit such opportunities.

In our simulations, each period consists of a five-day work week, each day having six slots. We vary the arrival rate of different task types and vary the percentage of priority tasks over different simulations. Each task is generated and allocated to a manufacturer depending on whether the manufacturer can accomplish the task using one of its suppliers. A manufacturer selects one supplier over another using a first-price sealed-bid auction protocol.

We used a three-level supply chain for our simulations (see Figure 1), each level being populated by one or more enterprises having similar functional capabilities. There is one main manufacturer in level one, six secondary manufacturers in level two, and 12 suppliers at level three. We assume a whole task T to be a combination of parts—that is, $T = L_m + L_{sm} + L_{su}$, where L_m, L_{sm}, and L_{su} are the main manufacturer's, secondary manufacturers', and suppliers' tasks, respectively. The main manufacturer contracts part of each task it must complete to a secondary manufacturer, which in turn contracts part of each task it wins to a supplier in some subset of all suppliers. We call this subset the secondary manufacturer's *supplier window*. Our experiments use enough secondary manufacturers and suppliers that each supplier can receive contracts from exactly two different secondary manufacturers.

A task assigned to the main manufacturer has two properties:

- *Priority* has a value of either 0 (ordinary) or 1 (high). High-priority tasks have a one-day deadline, and low-priority tasks have a one-week deadline.
- *Length* refers to how many time units (slots) the task requires for completion.

A task must be scheduled in consecutive slots on the same day. Higher-priority tasks are more profitable.

In our simulation, tasks are generated stochastically, and we vary the task distribution to verify the different algorithms' robustness. In one scenario, most tasks arrive early in the week, primarily low-priority with a deadline of one week, with fewer high-priority, short-deadline tasks generated later in the week (see Table 1). Another task distribution type generates tasks of different lengths and arrival dates with equal probability. Table 1 presents the probability of generating high-priority and low-priority tasks of varying length on each day of the week.

Suppliers can use one of the three pricing strategies and one of the four scheduling algorithms described earlier. Our simulation doesn't permit preemptive scheduling, which would preclude using leveled commitment protocols.[5] So, a supplier can't undo its commitment to a contract to serve another contract. All suppliers use the same initial bid price: this value equals the task's length for an ordinary task and is twice the task's length for a high-priority task. Subsequently, on the basis of their win/loss record for past contracts, suppliers adjust their bids using one of the price adjustment schemes discussed earlier. Suppliers can bid for that task only if their current schedules can accommodate a task announced by a secondary manufacturer and they are within that manufacturer's supplier window. The supplier with the minimum bid price wins the contract. When a supplier wins a bid, it increases its *wealth* by an amount equal to its bid price. Initially, all suppliers have zero wealth.

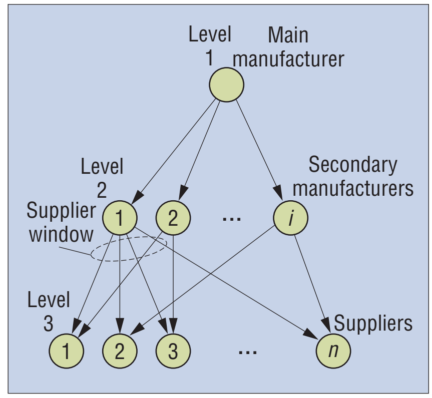


Figure 1. The supply chain structure used in the market simulations.

Experimental results

We also investigated the effectiveness of different pricing strategies. To describe our results, we represent first-fit, best-fit, worst-fit, and expected utility-based scheduling algorithms using FF, BF, WF, and EU, respectively. We refer to suppliers' different pricing strategies—linear, defensive, and impatient—as LIN, DEF, and IMPT, respectively. If a supplier uses scheduling strategy X and pricing strategy Y, we call that supplier X_Y. For example, we refer to a supplier using the EU scheduling algorithm and the LIN pricing strategy as EU_LIN.

The first experiment evaluates the linear pricing strategy's relative merits in exploiting market niches created by the different scheduling strategies. We use 12 suppliers, three each of FF_LIN, BF_LIN, WF_LIN, and EU_LIN, and generate 200 tasks per week. Figure 2 shows the average wealth these strategies generate on different days of the week. We assume that after one agent is assigned a task, it will complete it successfully, so it generates the wealth whenever the task is assigned. Initially, FF, BF, and WF suppliers generate more wealth by winning more contracts, while the EU strategy passes over some of these tasks. On the fourth and fifth

Table 1. Probabilities of generating tasks.

Task length (days)	Monday		Tuesday		Wednesday		Thursday		Friday	
	High priority	Low priority	High priority	Low priority	High priority	Low priority	High priority	Low priority	High priority	Low priority
1	.001	.14	.005	.005	.002	.010	.03	.01	.04	.01
2	.001	.14	.005	.005	.002	.010	.03	.01	.04	.01
3	.001	.15	.005	.005	.001	.005	.02	.02	.03	.01
4	.001	.15	.005	.005	.001	.005	.02	.02	.03	.01

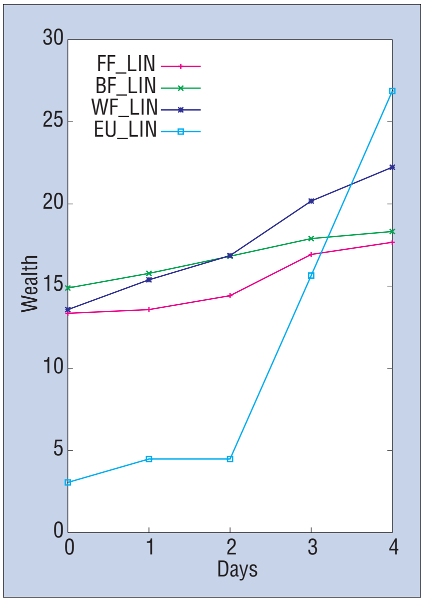


Figure 2. Average wealth earned by different supplier types using linear pricing and when three suppliers use each of four scheduling strategies.

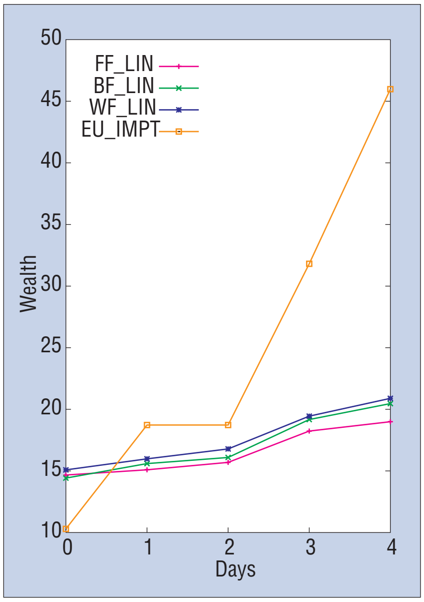


Figure 3. Average wealth earned by different supplier types using linear or impatient pricing and when three suppliers use each of four scheduling strategies.

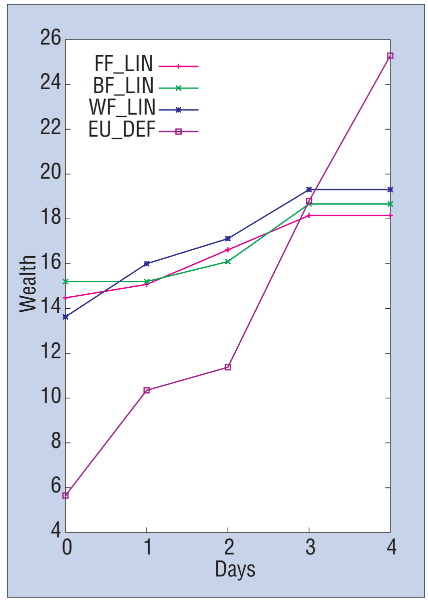


Figure 4. Average wealth earned by different supplier types using linear or defensive pricing and when three suppliers use each of four scheduling strategies.

days, EU accommodates more tasks, especially higher-priority ones, thus accumulating more wealth than the other supplier types. In Figure 2, we see that responsive scheduling lets EU suppliers create a market niche and win profitable contracts with less competition.

To follow up on this observation, we investigated whether the EU suppliers could further exploit the new market niche using a more aggressive pricing scheme. The second experiment, therefore, replaces three EU_LIN suppliers with three EU_IMPT suppliers but otherwise resembles the first one. Figure 3 shows the average wealth these different agents generated. As before, EU-based scheduling creates the market niche and accommodates the high-priority tasks in the last two days of the week. More importantly, EU_IMPT creates more wealth compared to EU_LIN in the last experiment. As suppliers win more contracts in the last two days, they further increase their bids to more aggressively exploit the market niche in the absence of competition.

To evaluate less aggressive pricing schemes in this context, the third experiment replaces the EU_IMPT suppliers with three EU_DEF suppliers, keeping the rest of the population unchanged. We find that EU suppliers still generate more wealth than those using other scheduling strategies, but the defensive or cautious pricing strategy extracts less wealth than the EU_IMPT agents do given the same market niche (see Figure 4).

So, the EU suppliers generate maximum wealth using impatient pricing, because this strategy takes full advantage of the opportunity to bid very high when little or no competition exists—that is, few agents can meet the scheduling requirements for the high-priority tasks generated in the last two days of the week. Therefore, if the scheduling strategy distribution is somewhat even and high-priority tasks come in bursts with imminent deadlines, aggressive pricing schemes similar to the impatient strategy used here can significantly increase wealth for predictive schedulers.

Next, we evaluate whether the impatient pricing strategy would benefit suppliers using the FF, BF, and WF scheduling strategies as well. Figure 5 compares the results of two experiments, both with 12 suppliers, three for each scheduling algorithm. In one, FF, BF, and WF suppliers follow the IMPT pricing strategy and EU suppliers use LIN pricing. In the other, all suppliers use LIN pricing. We find that FF, BF, and WF suppliers perform better when they use IMPT pricing. With uniform scheduling strategy distributions, then, all suppliers can benefit from aggressive pricing as market niches of differing size and wealth are produced for all scheduling strategies.

Next, we evaluate the pricing schemes' relative performance when all suppliers use the EU scheduling strategy. We again use 12 suppliers, four each using LIN, IMPT, and DEF pricing strategies, and reduce the tasks to 100 per week. Figure 6 shows that the IMPT supplier perform best, followed by LIN and then DEF. The impatient strategy performs well here for somewhat opposite reasons than in the previous experiments. Because all suppliers are opting for similar high-priority jobs, competition is high and more suppliers fail to win contracts. IMPT suppliers lower their bids sharply and win more auctions by underbidding suppliers with other pricing strategies. So, aggressive pricing can dominate even when all suppliers use the same scheduling strategy.

Is aggressive pricing preferable when everyone uses it? We ran three experiments, in each case keeping only suppliers following the EU scheduling strategy. So, we ran three experiments with 12 suppliers using EU_LIN, EU_DEF, or EU_IMPT schemes, respectively. Figure 7 shows that in this case suppliers with defensive pricing generate maximum wealth while suppliers with impatient pricing perform the worst. Without clear market niches, impatient suppliers get into a downward spiral of price wars, significantly eroding everyone's profitability. Defensive agents reduce their bids less aggressively when they lose contracts, and this more patient attitude rakes in more profit in the long run. In essence this is the opposite scenario of more wealth generated by aggressive pricing in the pres-

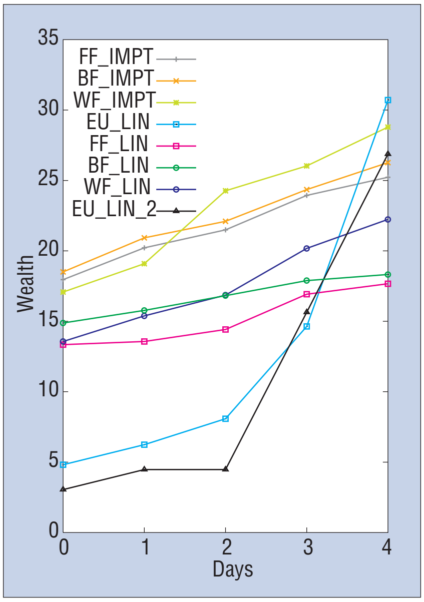


Figure 5. Average wealth earned by various supplier types. In one experiment, FF, BF, and WF use linear pricing; in the other, they use impatient pricing. EU suppliers use linear pricing in both experiments.

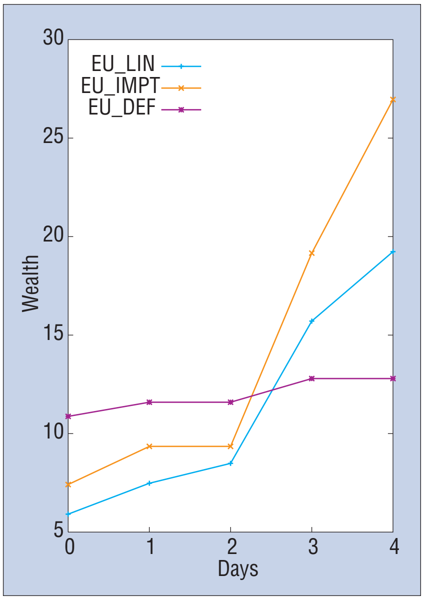


Figure 6. Average wealth earned by different supplier types all using EU scheduling and different pricing strategies.

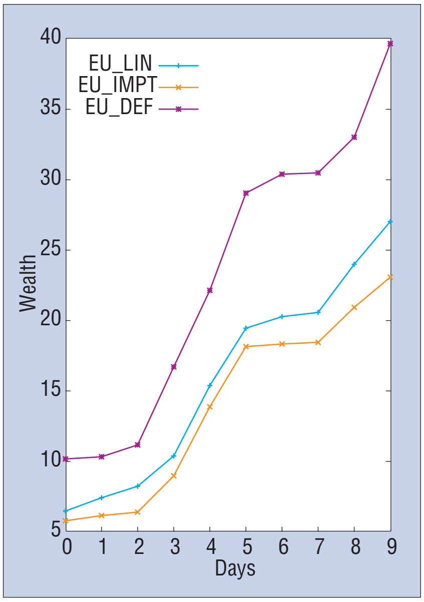


Figure 7. Average wealth earned by different suppliers using EU scheduling and different pricing mechanisms when they are the only type of supplier in the population.

ence of market niches. So, whereas aggressive pricing generates more wealth when market niches exist, less aggressive pricing mechanisms produce more profits when they don't.

Our approach offers a novel combination of opportunistic scheduling and aggressive pricing to improve suppliers' profitability in supply chains in the context of B2B trading. To build on this work, we plan to develop an analytical model to predict and choose the best scheduling and pricing strategy combination, whether given or having formed expectations of competing agents' strategy choices and task arrival distributions. We also plan to augment this work by adaptively selecting the scheduling strategy from observed job mixes.

To better represent real-life supply chains, our model should incorporate the concept of decommitment and penalty in the contracting process. We plan to investigate the effect of trust in this context. We also plan to study resource swapping based on side payments, which would let effective coalitions develop to better respond to uncertain market needs. ■

Acknowledgments

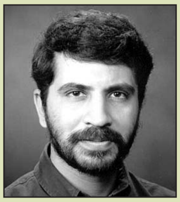
US National Science Foundation award IIS-0209208 partially supported this work.

References

1. W. Shen and D. Norrie, "An Agent-Based Approach for Manufacturing Enterprise Integration and Supply Chain Management," *Globalization of Manufacturing in the Digital Communications Era of the 21st Century: Innovation, Agility, and the Virtual Enterprise*, G. Jacucci et al., eds., Kluwer Academic Publishers, 1998, pp. 579–590.
2. J. Swaminathan, S. Smith, and N. Sadeh, "Modeling the Dynamics of Supply Chains," *Proc. AAAI-94 SIGMAN Workshop*, AAAI Press, 1994, pp. 113–132.
3. M. Barbuceanu and M.S. Fox, "Coordinating Multiple Agents in the Supply Chain," *Proc. 5th Workshop Enabling Technologies: Infrastructure for Collaborative Enterprises* (WET-ICE 96), IEEE CS Press, 1996, pp. 134–142.
4. S. Saha and S. Sen, "Opportunistic Scheduling and Pricing in Supply Chains," *KI—Zeitschrift Künstliche Intelligenz* [AI—J. Artificial Intelligence], vol. 18, no. 2, 2004, pp. 17–22.
5. T. Sandholm and V. Lesser, "Leveled Commitment Contracting: A Backtracking Instrument for Multiagent Systems," *AI Magazine*, vol. 23, no. 3, 2002, pp. 89–100.

The Authors

Sabyasachi Saha is a PhD student in computer science at the University of Tulsa. His primary research interests are multiagent systems and machine learning. He received an MS in computer science from the Indian Statistical Institute, Calcutta, and an MS in statistics from the University of Calcutta. Contact him at the Math and Computer Science Dept., Univ. of Tulsa, 600 S. College Ave., Tulsa, OK 74104; saby@utulsa.edu.



Sandip Sen is a professor of computer science at the University of Tulsa. His primary research interests include multiagent systems, machine learning, and genetic algorithms. He has a PhD in intelligent distributed scheduling from the University of Michigan. Contact him at the Math and Computer Science Dept., Univ. of Tulsa, 600 S. College Ave., Tulsa, OK 74104; sandip@utulsa.edu.