Opportunistic Scheduling and Pricing in Supply Chains

Sabyasachi Saha, Sandip Sen

Supply chain management systems play an important role in the globalized trading markets where sub-contracts and sub-tasks are given to other manufacturer or suppliers through competitive auction-based contracts. We believe that the scheduling efficiency of the suppliers play an important role in their profitability in a supply chain. The suppliers can decide their scheduling strategy for completing the contracted tasks depending on their capacity, the nature of the contracts, the profit margins, and other commitments and expectations about future contracts. Such decision mechanisms can incorporate task features including the priority and length of the task, scheduling windows, etc. Task scheduling decisions for a supplier should be targeted towards creating a schedule that is flexible about accommodating high priority tasks at short notice and, in general, those tasks which produce larger profits to the supplier. In previous work, we have evaluated the competitiveness of suppliers who use established scheduling heuristics like first fit, best fit and worst fit. We believe that a more opportunistic, adaptive, proactive, and robust scheduling approach can significantly improve the competitiveness of the suppliers and the efficiency of the entire supply chain. In this paper, we present an expected utility-based scheduling strategy for the suppliers and experimentally compare its effectiveness with previously used scheduling heuristics.

1 Introduction

With the increasing popularity of e-markets, business-to-business ($B2B$) trading has received an unprecedented boost. To stay ahead of the competition in this scenario, complexity of business strategies is increasing proportionally. New mechanisms and models are needed to enable modern businesses to adapt to this rapidly changing and evolving marketplace that present new possibilities and new risks. Agent-based systems have been proposed to optimize the functional modules and business strategies of an enterprise [1]. Stable and profitable $B2B$ relationships are predicated on the effectiveness of the supply chains linking these organizations. We believe that optimizing supply chain management is a key step towards developing more profitable businesses. Globalization, however, has made supply chains enormously complex to maneuver. With suppliers, retailers and manufacturers distributed geographically, integration of these entities for an efficient supply chain management system is a very challenging task. One major factor influencing the success of a supply chain management system is that of efficient scheduling of tasks and resources. Meeting deadlines of production, supply and transportation is key to a successful business. Moreover successful business strategies must provide for flexible and rapid responses to changes in market conditions as influenced by changed demand and supply scenarios. In this paper we focus on this critical aspect, and identify different scheduling strategies that can effectively accommodate market dynamics.

In $B2B$, dependability of the business entities determine the stability of the inter-business relationships. The net productivity of a manufacturing enterprise, for example, depends on the performance of the downstream components of its supply chain, which in turn depends on the performance index of the supplier entities. More, the efficiency of suppliers in meeting market demands with improved scheduling of delegated tasks, can increase the performance of manufacturers and strengthen the trading commitments between the parties. In this paper we use simulations to show the effects of different scheduling algorithms in their ability to process dynamically arriving task requests. The tasks which the manufacturers have to complete differ in priority or the deadline before which they must be completed. Tasks of different priorities represent the dynamics of market demand. We use two types of tasks: *priority tasks* with imminent deadlines and *ordinary tasks* with a relatively larger time window available for processing.

We use a contracting framework for connecting the suppliers to the manufacturers: manufacturers announce contracts for tasks with given specifications (deadline and processing time); suppliers bid on these tasks with prices; and the contract is allocated by an auction to a supplier who fulfills all task constraints [13].

A supplier who can meet deadlines that few others can meet will be able to demand more from the manufacturer for their services. We investigate the effect of different scheduling algorithms on the profitability of the suppliers under varying task mixes and group composition. To achieve this, we simulate a three-level supply chain with primary and secondary manufacturers and a group of suppliers. Supplier scheduling strategies are chosen from a set of reasonable heuristics for scheduling dynamically arriving tasks. We analyze the ability of different strategies to produce more flexible schedules. The more flexible a schedule the more readily it can accommodate late arriving and high priority tasks. We analyze an agent-based price adjustment mechanism by which suppliers can react to different contract situations. This allows for flexible suppliers to benefit monetarily. Our aim is to identify scheduling policies that benefit suppliers in increasing profitability by allowing them to accommodate demands that others cannot.

2 Responsive supply chains

In the present-day markets, business enterprises have integrated the functional groups within a supply chain [17]. Additionally, the success of a modern enterprise depends, in turn, on its success to maintain stable and profitable relationships with other enterprises that it is functionally coupled to. In [2] we find how software agents following *structured conversations* achieve effective coordination in the context of a supply chain. However, in the present age of globalization, achieving this goal is a daunting task. Today's supply-chain managers have to consider supplier enterprises who are geographically distributed. They have to manage the flow of materials from the distributed suppliers to global manufacturing facilities. Emphasis should be given on keeping low inventory, minimizing operations cost, having flexibility and providing efficient customer service. Effective decision support tools on modeling the dynamics of the supply chain offer an edge of performance to managers in achieving these goals [18].

While a company might adopt a myopic policy of adjusting the logistics activities individually to attain overall excellence, they will fail to visualize the interdependency-tradeoffs present among the activities. It is acknowledged that *integrated logistics* is the key to better business. Equally crucial is the issue of operational flexibility to accommodate the fluctuating demand of the market. In a successful $B2B$ trading scenario, *integrated logistics* coupled with this flexibility can guarantee business growth. Adoption of a suitable *supply-chain strategy* as well as an effective *supply-chain planning* is deemed necessary. Supply-chain strategy refers to the policy tradeoff between cost and service that the company would adopt (at what point on the cost-service curve does the company wishes to place itself?). Supply-chain planning is the company's working policy to achieve the adopted strategy (what operations dynamics should the company adopt to move itself from the current position to the target position?).

The motivation behind our work has been this aspect of a supply chain being responsive to dynamically arriving tasks which is hinged upon the performance of the suppliers. We believe that a distributed approach where each supplier tries to maximize its profit will enable a robust, scalable solution to the responsiveness issue. We use simulations to evaluate various scheduling algorithms used by suppliers to schedule the tasks allocated to them. We have designed a utility based strategy which uses a model of periodic variations in the distribution of arriving tasks. Tasks with varying priorities (deadline) are stochastically generated, and arrive sequentially. Suppliers bid for tasks in auctions to win contracts. Suppliers with a more flexible scheduling policy can expect to win larger number of more profitable contracts. Each suppliers goal is to maximize local profit. Suppliers with more flexible schedules can meet the deadline requirements of tasks which few others can meet and hence hike their winning bids, leading to increased profitability.

3 Scheduling strategies

We evaluate four scheduling strategies for the suppliers. The goal is to allocate a task t of length $l(t)$ and deadline $dl(t)$ (the deadline determines priority $\mathcal{P}(t)$: we have assumed that a task with immediate deadline is considered to be of high priority or priority 1 and a task of normal deadline is of ordinary priority or priority 0.) on the calendar $\mathcal{C}$ given the current set of scheduled tasks on in $\mathcal{C}$. The following scheduling strategies have distinct motivations:

First fit: This 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.

Best fit: This strategy searches the entire feasible part of the schedule (between the current time and deadline $dl(t)$) and assigns t to one of the intervals such that there is a minimal number of empty slots around it. This strategy produces clumped schedules, but the clumps need not be at the front of the schedule.

Worst fit: This strategy searches the entire feasible part of the schedule (between the current time and deadline $dl(t)$) and assigns t to one of the intervals such that there is maximum empty slots around it.

Expected Utility Based (EU): All the strategies discussed above schedule the tasks in a greedy way. They bid for a task whenever it can schedule it. In reality, there are periodic patterns in the task arrival distribution. Hence, it might be prudent at times to not bid for a task with the expectation of keeping the option open to bid for a more profitable task in the future. In the EU strategy, agents use the knowledge of the task arrival pattern and whenever a new task arrives a decision is made whether it is worthwhile to bid on it. If the task arrival pattern is not known *a priori*, it can be learned over time provided the pattern does not change drastically in a short time. Unfortunately, there is a risk involved in such opportunistic scheduling as some production slots may remain unused if the expected high-profit tasks never materialize. According to this algorithm, an agent bids for a task if its expected utility is positive. It is negative if the expectation that it can make more profit later overpowers the risk of not bidding now. So, it will not bid for this task. We discuss the expected utility based bidding strategy in the next section.

4 Utility of scheduling a task

For each arriving task $t \in \Upsilon$, where Υ is the set of all tasks, the utility for the agent for scheduling that task is given by the function $u(l(t), \mathcal{P}(t))$, where $l(t)$ is the length and $\mathcal{P}(t)$ is the priority of that task. Let $esd(t)$ and $dl(t)$ be the earliest start date and the deadline for processing the task. 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 the calendar $\mathcal{C}$ 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, $\mathcal{T}^k_\alpha = \{T | T \subset \Upsilon, \alpha = \sum_{t \in T} l(t), \mathcal{P}(t) > k \, \forall t \in T\}$ is the set of all those task combinations where the lengths of the tasks in each combination adds up to α 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 $n^T_{i,j}$ to 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 at a later time on day d, where $\mathcal{H}$ is the history of tasks that have already arrived. Given the task distribution (discussed in the Section 5), the probabilities $Pr(i, n^T_{i,j}, d, \mathcal{H})$ can be calculated by using 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

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

where

$$\begin{aligned} AU(t,d,\mathcal{H},h) = \;\; &\frac{1}{|T^{\mathcal{P}(t)}_h|} \sum_{T \in T^{\mathcal{P}(t)}_h} \sum_{k > \mathcal{P}(t)} \sum_{i=1}^{l_{max}} \\ &Pr(i, n^T_{i,k}, d, \mathcal{H}) n^T_{i,k} u(i,k), \end{aligned}$$

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 the maximum possible length of a task, and the function den returns the density of the calendar, i.e., the percentage of slots that have been scheduled, upto a given date (to facilitate scheduling we use $den(\mathcal{C}, 0) = 1$).

For a given day on the calendar, the expected utility expression adds the sum of the utility of the current task to the difference of the average utility of all possible ways of filling up the calendar with higher priority tasks with or without the current task being scheduled. We schedule the task provided that the expected utility is positive for at least one of the days considered and the day chosen is the one which provides the maximum expected utility, i.e., $\arg\max_{d \in D} EU(t,d,\mathcal{H},\mathcal{C})$.

5 Experimental Framework

In our simulations, each period consists of a work week of 5 days, each day having six slots. We vary the arrival rate of different task types with the percentage of priority tasks in different simulations. Each of the tasks are generated and allocated to a manufacturer depending on whether the manufacturer can accomplish the task by using one of its suppliers. Selection of one supplier over another by a manufacturer is performed through a first-price sealed-bid auction procedure.

The supply chain that we have used for our simulations consists of 3 levels, each 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 twelve *suppliers* at level three. The supply chain structure is shown in Figure 1. We have later changed the number of suppliers from this configuration to show the effect of too many suppliers of the same type. The main manufacturer will contract a part of each task that it has to complete to one of the secondary manufacturers. Each secondary manufacturer, in turn, contracts part of each task it wins to a supplier in some subset of all suppliers. We call this subset of suppliers the *supplier window* of the secondary manufacturer. In our experiments, we have chosen the number of secondary manufacturers and suppliers such that each supplier can receive contracts from exactly two different secondary manufacturers.

A task assigned to the main manufacturer has two properties: (i) *priority*, which can have a value of either 0 (indicating an ordinary task) or 1 (indicating a priority task); the corresponding *deadline*, which is the latest time, from the start of the next task execution period, by which the task should be completed. High priority tasks have a *deadline* of one day and low priority tasks have *deadline* one week, and (ii) *length*, which is the number of time units (slots) the task requires to be completed. A task must be scheduled on consecutive slots on the same day. Higher priority tasks are more profitable. Here we assume that initially the utility of an ordinary task is equal to the length of the task and the utility of the high priority task is twice the length of the task.

In our simulation, the tasks are generated stochastically. We use different types of task distributions to verify the robustness of the different algorithms. In the first type we assume that high priority tasks arrive in the later part of the week. Most of the tasks come in the first days in the week and they are mostly of low priority having a deadline of the whole week. Some tasks are generated in the last part of the week and with an immediate deadline. In another type that we have experimented with, all tasks of different length and arrival date, are generated with equal probability. In table 1, we present an example of such a task distribution.

Suppliers at level 3 of the supply chain can employ any one of the four scheduling algorithms mentioned in Section 3. Preemptive scheduling is not allowed in our simulation which precludes use of leveled commitment protocols [14]. So, a commitment to a contract by a supplier cannot be undone by any task that arrives to the supplier later. The suppliers (those in the supplier window of the manufacturer) who bid for a task from a secondary manufacturer have the same initial bid price. Only those suppliers who can accommodate, within their current schedule, a task announced by a secondary manufacturer and are within the supplier window of that manufacturer, can bid for that task. The sup-

l	Probabilities									
1	.001	.14	.005	.005	.002	.01	.03	.01	.04	.01
2	.001	.14	.005	.005	.002	.01	.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

Table 1: Probabilities of Generating Tasks: the row number gives the length of the task; the column $(2i-1)$ shows the probability of generating high priority task in the i^{th} day and the column $2i$ shows the probability of generating a low priority task.

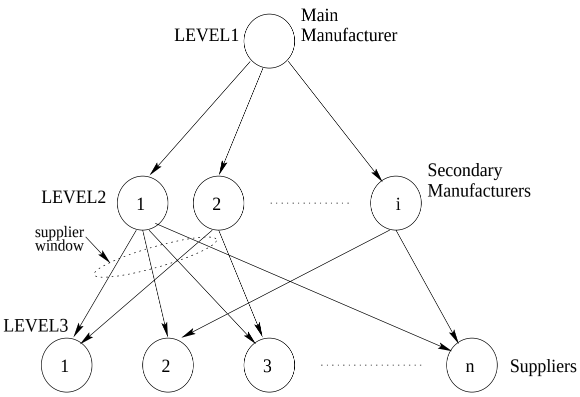

Figure 1: Supply chain structure.

plier with the minimum bid price wins the contract for the task from the secondary manufacturer, i.e., contracts are awarded by a first-price-sealed-bid auction. Winning a task gives the supplier incentive to increase its bid price for that type ($[priority, length]$ combination) of task. A corresponding decrease in the bid prices occurs if a supplier loses a bid. We have used a constant value of 1 for the increment and decrement in the bid prices, with the constraint that the bid price cannot go below the initial bid price. When a supplier wins a bid, it increases its *wealth* by an amount equal to its bid price. Initially, wealth of all suppliers is set to zero.

6 Experimental Results

To describe our experimental results we use the following conventions: suppliers using the worst fit algorithm are represented by WF, those using first fit by FF, those using best fit by BF and those using expected utility based strategy by EU.

In the first set of experiments, we have 3 suppliers of each type and the task distribution is as mentioned in the table 1. A run consists of 20 periods (weeks) of 5 day each. The number of tasks per week is 150. We study the variation in the wealth earned by suppliers using different types of scheduling algorithms. The results are shown in Figure 2. The average wealth earned by EUs are better compared to the other three scheduling types. Since the EU strategy is not myopic, the suppliers using it wait for more profitable high priority tasks, ignoring low-priority tasks that arrived earlier,

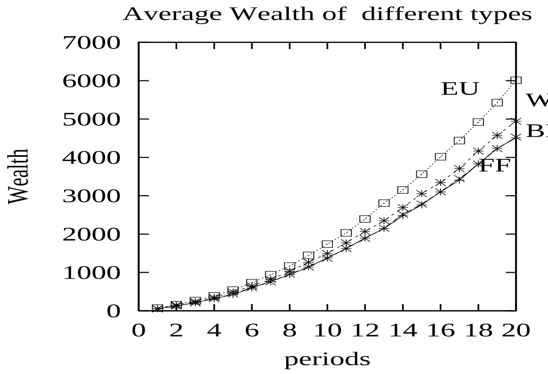

Figure 2: Average Wealth earned by different supplier types. There are 3 suppliers using each of four scheduling strategies. Task distribution is as shown in Table 1.

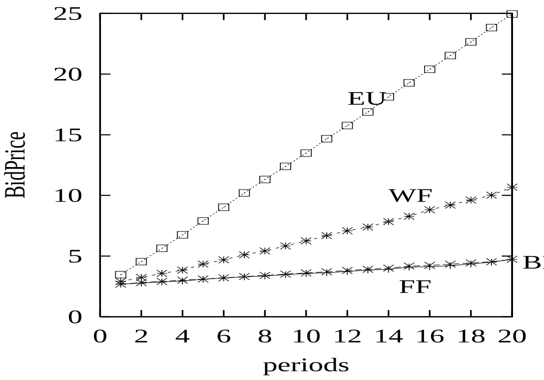

Figure 3: Increase in bid prices for high priority tasks over the course of a run.

and make more profits throughout the period of execution of 100 days. The EUs benefit from earning more than the other types of suppliers by winning the bids for these high-profit tasks. Other, myopic suppliers, do not consider passing on arriving tasks, and hence fill up slots by committing to tasks arriving early in the run, and therefore fail to accommodate and bid for high priority tasks that arrive later. The EUs are followed in performance by WFs. Since the worst fit algorithm produces an evenly-loaded schedule, suppliers using this type of scheduling can better accommodate the high priority tasks (and ordinary tasks too) that arrive more frequently later in the contracting period. We note that at the end of the run, the prices for the high priority task charged by EU suppliers increase considerably (see Figure 3). Interestingly, Figure 4 shows that the EU scheduling strategy not only garners more profit, but also accommodates more tasks compared to the other scheduling strategies. The reason for this is that in the task distribution, more high priority tasks are of smaller length. In this metric too, the WF strategy comes in second: they accommodate more tasks compared to BF and FF.

In the second set of experiments, we alter the task dis-

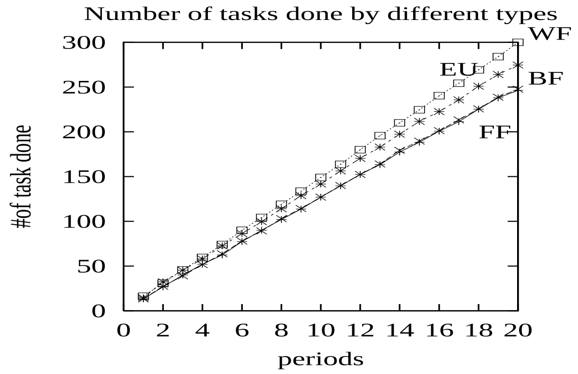

Figure 4: Number of contracts won by different supplier types.

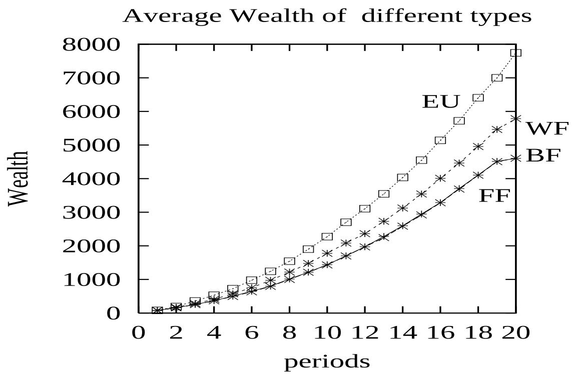

Figure 5: Average Wealth earned by different supplier types with uniform task distributions.

tribution and generate the tasks such that all task types and their arrival dates are equally likely. Here, tasks per week is decreased to 100, to see if lower task loads will allow the other strategies to compete more effectively with the EU strategy. We present the wealth distribution from this set of experiments in Figure 5. Here also the EU suppliers do better than all the other types of suppliers. The reason is that though all of the suppliers are accommodating some high priority tasks, the other strategies also contract a number of ordinary tasks. The EUs, however, focus more on the high priority tasks. Since, it is highly probable that different high priority tasks will come everyday, they will try to contract only the high priority tasks and increase their profit. Here the bid price for all supplier types increase significantly, but the increment is much more for EUs. So, we see that the EU suppliers are robust and can outperform suppliers using other strategies under different task distributions.

In the third set of experiments, we assume that the EU suppliers are not knowledgeable about the task distribution. So, they try to estimate the distribution in the first two periods and to bid for all tasks and use a first fit strategy during this time. From the third period onward, however, they use the expected utility strategy based on the estimated task dis-

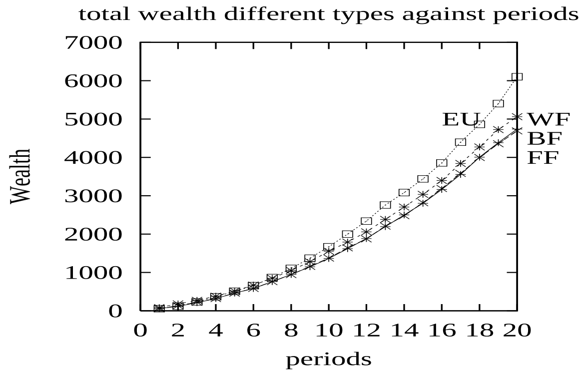

Figure 6: Average Wealth earned by different type of suppliers, run with 3 of each suppliers. Task distribution is learned by EU suppliers.

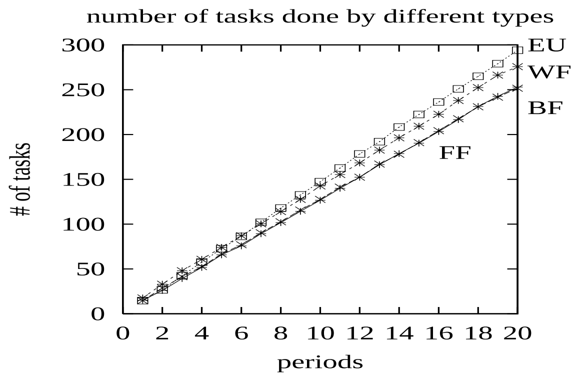

Figure 7: Number of contracts won by different suppliers. Task distribution is learned by EU suppliers.

tributions. In Figure 6 we can see that EUs utility is less in the first two periods. But the EUs dominate the other strategies in the later periods. Similar observations hold for the number of tasks contracted by each type of suppliers shown in Figure 7. In this experiment set the EU continues to estimate, in every period, the task arrival probability distribution with the task arrival data of last two periods using relative frequency. Unlike the earlier experiments they use this estimate of task distribution instead of the real task distribution for calculating expected utility. So, this EU scheduling strategy works well even when the task generating distribution is not known *a priori*.

In the fourth set of experiments, we investigate the effect of increasing the number of EU suppliers in the population. In this experiment we increase the number of the EUs and keep number of supplier under a secondary manufacturer equal and change tasks proportionately. From Figure 8 we observe that if the number of EU suppliers increase, their individual utility decreases. This decrease is due to the presence of too many "smart" people (EUs) in the population. The competitive advantage of EU suppliers is predicated on the absence of similar strategic schedulers who keep their

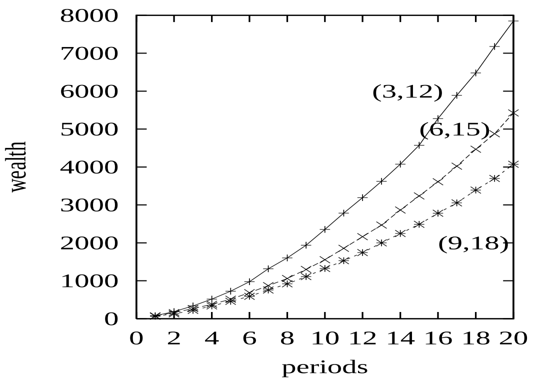

Figure 8: Average wealth varies as the number of EU increases,in the figure the number of EU and total number of suppliers are mentioned in the brackets.

options until late in the run with the goal of accommodating late-arriving high-payoff tasks. In absence of competition it can hike the price and garner larger profits. But, presence of many strategic bidders increases competition for late-arriving high-priority tasks and eliminates such opportunistic profit making. Though EUs still perform better than other strategies, the differences in performances decreases because they now start bidding for a larger number of low-priority tasks after the first period. This happens because the $winchance$ value is less than in the previous cases.

We note that, in these experiments, we have supplied a relatively large number of tasks. We use at least a total of 100 tasks for 12 suppliers per weak (which means 360 empty slots per week). We have noticed that if the number of tasks per period is very small, the profits of different suppliers or the number of tasks they do do not differ significantly. This is because then scheduling strategies do not have a significant effect on accommodating late-arriving tasks as most of the slots remain empty.

7 Related work

Scheduling in the context of supply chain management has long been an active area of research among the operations research community. Typically, the operations research approach was to search for optimum schedules from a single objective function [9]. Finding the optimal schedule, however, becomes an intractable problem in complex real-world situations. Faced with such real-world problems, researchers adopted the scheduling heuristic approach [11] that gives the scheduler the option of evaluating alternative solutions computed locally.

Apart from the operations research community, solving scheduling problems have also been the focus of AI researchers. Scheduling by expert systems has been one of the premiere approaches in AI [4], that attempts to capture the decision-making ability of a human expert using situation-specific "if-then" rules contained in a knowledge base. Iterative techniques like genetic algorithms and simulated annealing that generate high-fidelity solutions using neighborhood-based search form another important AI approach to scheduling.

More recently the problem of supply chain management has drawn the attention of multiagent systems (MAS) researchers. In MAS, a supply chain is conceptualized as a group of collaborative autonomous software agents [8]. It is argued that managers can better coordinate and schedule processes by distributing the corporate-wide business management system into autonomous problem solving agents. This approach is more appropriate when the participating enterprises are geographically distributed and manually controlling $B2B$ trading is not possible. Tackling coordination in supply chains using partial constraint satisfaction problems by mediating agents form the essence of [3]. Negotiation-based supply chain management has been proposed in [5], where a *virtual* supply chain evolves by the negotiating software agents when an order arrives. The *blackboard architecture* is a proven methodology for integrating multiple knowledge sources for problem solving [6]. Using this architecture, a mixed-initiative agent-based architecture for supply chain planning and scheduling is implemented in MASCOT [12].

The contract-bid-award framework used in our simulation is similar to the contract-net framework that has been used widely in multi agent systems research to solve the "connection problem", i.e., finding people who can do the job that one needs to get done [15].

Scheduling for increased flexibility for accommodating late-arriving tasks has been studied in the context of distributed meeting scheduling in the form of search biases [16]. The supply-chain setting do not involve the concurrent timing agreement requirement of the meeting scheduling domain. On the other hand the meeting scheduling domain does not involve the competitive bidding framework.

We have not experimented with sophisticated price adjustment strategies as in [10] or observed complex price dynamics as in [7]. [10], for example, observe interesting cyclical price-wars arising out of concurrent price learning schemes as each seller tries to respond to price adjustments of other sellers. In our domain the price adjustment by a supplier is directed not at under-cutting the other suppliers. Rather price hikes are in response to the realization that competition is limited or non-existent. Price adjustments are therefore largely an effect of being able to fulfill announced task constraints and are a by-product of scheduling decisions.

8 Conclusions and future work

We have experimentally evaluated the relative effectiveness of different scheduling heuristics in accommodating dynamically arriving mix of tasks in the context of $B2B$ trading. We note that when a significant number of high priority tasks are present, the expected utility based strategy and the worst-fit strategy provides greater flexibility. On the contrary, when such tasks are rare and also there are longer tasks in the mix, best-fit strategies may be good enough.

Additionally we have studied the competitive advantage to be gained by suppliers in increasing their price when they can schedule effectively to win task contracts. As such, our approach provides a novel combination of opportunistic scheduling and pricing to improve profitability of suppliers in the context of a supply chain.

We need to augment this work by adaptively selecting the scheduling strategy from observed job mixes. We also plan to study swapping of resources based on side-payments. This allows for effective coalition to develop to better respond to uncertain market needs.

Acknowledgments This work has been supported in part by an NSF award IIS-0209208.

References

[1] Sena Arpinar, Asuman Dogac, and Nesime Tatbul. An open electronic marketplace through agent-based workflows: MOPPET. *International Journal on Digital Libraries*, 3(1):36–59, 2000.

[2] M. Barbuceanu and M. S. Fox. Coordinating multiple agents in the supply chain. In *Proceedings of the Fifth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 134–142, 1996.

[3] J. Christopher Beck and Mark S. Fox. Supply chain coordination via mediated constraint relaxation. In *Proceedings of the First Canadian Workshop on Distributed Artificial Intelligence*, May 1994.

[4] E. Bensana, M. Correge, G. Bel, and D. Dubois. An expert-system approach to industrial job shop scheduling. In *Proceedings of the International Conference on Robotics and Automation*, pages 1645–1650, 1986.

[5] Y. Chen, Y. Peng, T. Finin, Y. Labrou, R. Cost, B. Chu, R. Sun, and R. Willhelm. A negotiation-based multi-agent system for supply chain management. In *Working Notes of the ACM Autonomous Agents Workshop on Agent-based Decision-support for Managing the Internet-enabled Supply-chain*, May 1999.

[6] L. D. Erman, F. Hayes-Roth, V. R. Lesser, , and D. R. Reddy. The hearsay-ii speech-understanding system: Integrating knowledge to resolve uncertainty. *Computing Surveys*, 12(2):213–253, June 1980.

[7] Amy Greenwald, Jeffrey O. Kephart, and Gerry J. Tesauro. Strategic pricebot dynamics. In *Proceedings of the First ACM Conference on Electronic Commerce*, pages 58–67, New York, NY, 1999. ACM Press.

[8] N. R. Jennings, P. Faratin, T. J. Norman, P. O'Brien, and B. Odgers. Autonomous agents for business process management. *International Journal of Applied Artificial Intelligence*, 14(2):145–189, 2000.

[9] S. M. Johnson. Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1(1):61–68, 1954.

[10] Jeffrey O. Kephart and Gerald J. Tesauro. Pseudo-convergent q-learning by competitive pricebots. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 463–470, San Francisco: CA, 2000. Morgan Kaufmann.

[11] I. Kurtulus and E. W. Davis. Multi-project scheduling: Categorization of heuristic rules performance. *Management Science*, 28(2):161–172, 1982.

[12] N. Sadeh, D. Hildum, D. Kjenstad, and A. Tseng. Mascot: An agent-based architecture for coordinated mixed-initiative supply chain planning and scheduling. In *Working notes of the ACM Autonomous Agents Workshop on Agent-Based Decision Support for Managing the Internet-Enabled Supply Chain*, May 1999.

[13] Sabyasachi Saha and Sandip Sen. Scheduling to be competitive in supply chains. In *Third International Conference on Intelligent Systems Design and Applications*, pages 459–468, August 2003.

[14] Tuomas Sandholm and Victor Lesser. Leveled commitment contracting: A backtracking instrument for multiagent systems. *AI Magazine*, 23(3):89–100, 2002.

[15] Tuomas W. Sandholm and Victor R. Lesser. Issues in automated negotiation and electronic commerce: Extending the contract net framework. In *First International Conference on Multiagent Systems*, pages 328–335, Menlo Park, CA, 1995. AAAI Press/MIT Press.

[16] Sandip Sen and Edmund Durfee. Unsupervised surrogate agents and search bias change in flexible distributed scheduling. In *First International Conference on Multiagent Systems*, pages 336–343, Menlo Park, CA, 1995. AAAI Press/MIT Press.

[17] W. Shen and D. Norrie. An agent-based approach for manufacturing enterprise integration and supply chain management. In *In G. Jacucci, et al (eds.), Globalization of Manufacturing in the Digital Communications Era of the 21st Century: Innovation, Agility, and the Virtual Enterprise*, pages 579–590. Kluwer Academic Publishers, 1998.

[18] J.M. Swaminathan, S. Smith, and N. Sadeh. Modeling the dynamics of supply chains. In *Proceedings the AAAI-94 Workshop on Reasoning About the Shop Floor*, August 1994.

Kontakt

Sabyasachi Saha, Sandip Sen
600 South College Avenue
Department of Mathematical and Computer Sciences
University of Tulsa
Tulsa, Oklahoma, 74104, USA
`sahasa,sandip@ens.utulsa.edu`

Bild

Sabyasachi Saha is a PhD student of Computer Science in the University of Tulsa. His primary research interests are Multiagent Systems and Machine Learning. He completed MS in Computer Science from ISI, Calcutta, India and in Statistics from University of Calcutta, India.

Bild

Sandip Sen is an Associate Professor of Computer Science in the University of Tulsa with primary research interests in multiagent systems, machine learning, and genetic algorithms.