2009

A multi-agent simulation framework for automated negotiation in order promising

Tung Bui
University of Hawaii, tung.bui@hawaii.edu

Hans-Juergen Sebastian
RWTH-Aachen, sebastian@or.rwth-aachen.de

Christoph Hempsch
RWTH-Aachen, chrish@or.rwth-aachen.de

Timo Bosse
RWTH-Aachen, gino.bosse@gmx.de

A multi-agent simulation framework for automated negotiation in order promising

Tung Bui
University of Hawaii
tung.bui@hawaii.edu

Hans-Juergen Sebastian
RWTH-Aachen
sebastian@or.rwth-aachen.de

Christoph Hempsch
RWTH-Aachen
chrish@or.rwth-aachen.de

Timo Bosse
RWTH-Aachen
gino.bosse@gmx.de

ABSTRACT

The purpose of this paper is to propose and test a multi-agent-based system for automated multi-attribute negotiation in order promising. In a make-to-order production model, it is not always possible to satisfy the Available-to-Promise (ATP) and Capable-to-Promise (CTP) conditions. Therefore, it is important to quickly explore alternate solutions that would satisfy both the customer and the supplier. We adopt the concepts of evolutionary system design that advocates for continuous exploration of new solutions based on extensive search and multi-attribute simulations that help identify for better negotiated solutions based on real-life ordering situations – changes of delivery date, price adjustments, addition/modifications of value-added services as part of the order. Results of our simulations showed that negotiation procedures did reduce the number of rejected orders and increase the overall revenue when negotiation concepts are introduced.

Keywords

Operational supply chain management, order promising, available-to-promise (ATP), capable-to-promise (CTP), negotiation, multi-agent system

INTRODUCTION

The Capable-To-Promise (CTP) function supports order promising in a short-term, order-based production environment. For incoming customer orders, it decides whether or not it is possible to fulfill the desired order quantity and delivery date. Under high costs of inventory of scarce raw materials and variable production capacity, orders may be rejected or not fulfilled based on the initial terms of the customers. Yet, a major weakness of the CTP function is its lack of support of any kind to help finding a quick alternate and mutually agreeable solution between the producer and the customer upon rejection of the initial order. When implementing a CTP function in practice this negotiation task is done “manually”, and the quality of the outcomes relies on the skills and experience of the person in charge of processing the order.

The purpose of this paper is to propose a multi-attribute negotiation process and to design an architectural framework to support communication, bargaining and negotiation activities in the CTP in a supply chain network. We seek to derive alternative offers in case a customer order must be denied.

The paper is organized as follows. First, we briefly introduce the domain of order promising within a make-to-order production environment and suggest how negotiation concept can be relevant to the problem at hand. We next attempt to operationalize the negotiation concepts with multi-attribute utility functions. A multi-agent based system is next introduced to allow us conduct the ability to satisfy orders based using negotiation concepts. Using two example scenarios, we discuss simulation runs and report the results.

ORDER PROMISING IN OPERATIONAL SUPPLY CHAIN MANAGEMENT

Available-To-Promise (ATP) and Capable-To-Promise (CTP) are known activities in the management of supply chains. Within a make-to-order or configure-to-order production environment, production or configuration is not initiated until the producer receives a customer order demanding the specific product. Due to the fact that the quantity of materials or components in stock or the production resources may be limited at a given point in time and cannot be replenished or extended before the desired date of delivery the producer has to decide on the

- quantity,
- due date and
- price

to commit to each customer order (Kilger and Schneeweiss, 2005).

ATP and CTP functions are widely discussed in literature (e.g., Ball et al., 2004; Kilger and Schneeweiss, 2005; Stadtler, 2005). In general terms, ATP is understood as a simple function that looks up the producers finished products inventory and reserves the quantity ordered by the customer. CTP in turn takes the whole production process into consideration to look ahead what quantity may be available within a certain time frame. (Some authors denote the functionality of CTP as Advanced ATP (Chen et al., 2001).

Figure 1 shows a basic workflow of the ATP and CTP functions. First, customer orders are received by the producer. Such a customer order usually contains a set of ordered products (or order positions) and the desired quantities as well as a delivery due date. Normally, a price is specified with the order as well. To check whether the order can be fulfilled, the ATP and CTP functions are executed consecutively. If the ATP function is able to reserve the ordered quantity from existing stock, the contract is fulfilled and the products are expected to be delivered as scheduled. If not, the CTP function checks if the production of an appropriate amount of the ordered products is possible on time. In case that the CTP function can fulfill the order, production and delivery are started. Otherwise, the customer order has to be rejected.

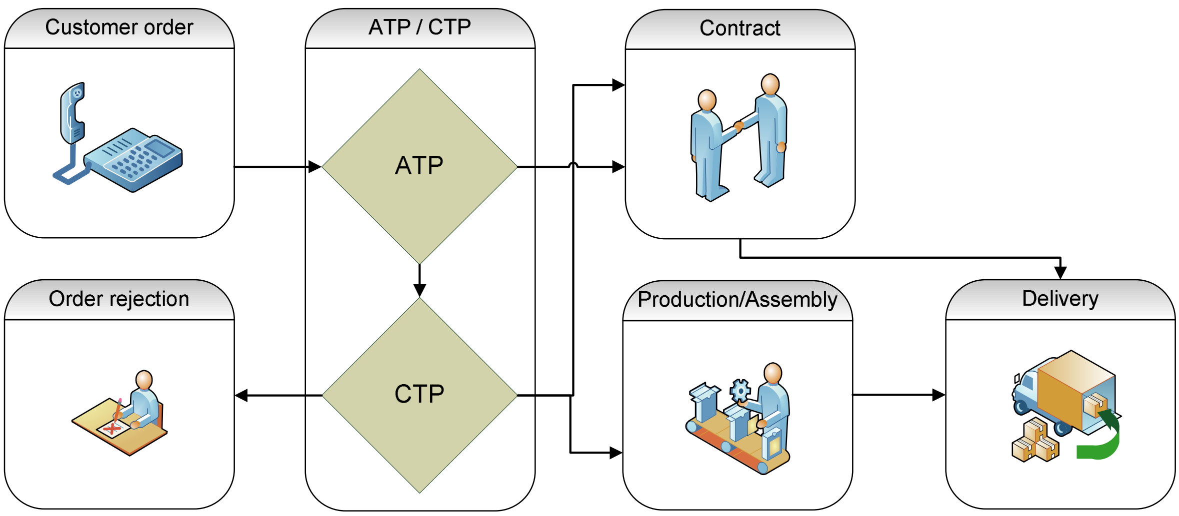

Figure 1: Basic workflow of Available-to-Promise (ATP) and a Capable-to-Promise (CTP) functions

The main objective of the producer in order promising, of course, is to maximize revenue and earnings by selling as many products to as many customers as possible by satisfying the demand. As we are considering a pull-based production environment customer satisfaction is of high importance in the long run. In general, there are three critical factors that determine the quality of an order promising system from a customer satisfaction point of view:

- Reaction time of the system: For customer satisfaction, the duration of the decision making process should be as short as possible.
- Quality of promised due date: The customer desires a short delivery time and a reliable prediction on it.
- Order acceptance rate: Only a small number of customer orders should be rejected unless the selection of accepted orders is solely based on short term profit maximization considerations. A rejected customer may buy the product from another producer – or now and probably also in the future.

There are multiple decisions apart from the ones mentioned above that are commonly incorporated into ATP and CTP to achieve these objectives. For example, to be able to accept more customer orders, order splitting or quantity splitting may be introduced. Order splitting allows the delivery of order positions of a customer order at different dates. Quantity splitting allows to split the ordered quantity into multiple orders and deliver these orders at different delivery dates.

The workflow around the CTP function described above is very generic and broadly discussed in literature. But how to proceed once a customer order is rejected by the CTP function is not yet studied in depth in literature.

AUTOMATED NEGOTIATION

To get a clearer understanding of negotiations, we first refer to the classification by Bichler et al. (2003): side negotiation procedures and models and side negotiation media and systems (e.g, negotiation support systems (NSS) and negotiation mediation systems (NMS). NMS implement negotiation processes between multiple entities. Their aim is to improve the efficiency of the negotiation processes through communications support and assistance toward integrative bargaining. Most NSS seeks to improve outcome of the party that uses the system. In contrast, and as its name suggests, NMS they are used to help the negotiation party to gain a more effective result. In this paper, we define negotiation in its broadest context, that is any activity that helps avoid a solution impasse, or better yet, one that would yield a win-win situation to both customers and suppliers. Acknowledging the existence of more than one issue in a typical negotiation, the general literature in multiple attribute utility theory advocates for the possibility of finding a compromise that would give each of the antagonists what they want most (e.g., Bui, 1987). Furthermore, the ability to identify new solutions that were not initially thought of could help resolve an impasse. Next, should the search for new alternate solutions fail to find a mutually agreeable solution (i.e., solution space), a NSS should try to guide the protagonists re-consider their expectations (i.e., goal space) based on updated situations. If this attempt fails and for the sake of finding a solution in a partnership-based supply chain network, NSS should explore new partners (i.e., actor space) that would qualify for the orders. This concept is known as evolutionary in the design of negotiation processes (Bui and Shakun, 2002). As shown in the next section, the consideration to split the quantity of an order that cannot be fulfilled or the adding of some additional services to a late delivery are examples of evolving the initial solutions to a new feasible set of possible solutions that are acceptable to all involved parties.

The notion of automated negotiation implies that some aspects of a negotiation are either conducted or at least supported by autonomous computer agents or parties. In the context of ATP or CTP, this automated negotiation could be of routine procedures (for example, fast and expanded search of “matching solutions”, quick estimation of delivery time, or instantaneous reporting of inventory levels). Furthermore, the agents could also be pre-programmed to act as a trained mediator looking for heuristics-based solutions. For example, the first procedural rule of an automated agent would be to immediately acknowledge the reception of a customer’s order, and the generation of alternate solutions should the initial order cannot be satisfied. In a distributed platform linking customers to suppliers, the automation of negotiation processes could be implemented by a series of simple to more functional agents, thus a multi-agent system. These agents can work in a sequential or parallel mode until a matched solution can be found and accepted.

POTENTIAL BENEFITS OF INTRODUCING NEGOTIATION CONCEPTS TO CTP

As mentioned above, the success of an order promising system depends on three critical factors, i.e., short reaction time, quality of promised due date and a high-order acceptance rate. Unfortunately, the producer’s and customer’s objectives regarding the order attributes are, at least in some cases, divergent (see Table 1). This needs to be taken into consideration by the producer whenever a customer order is rejected by the order promising system and a counter offer is computed. Table 2 shows suggested strategies on four negotiable order attributes for the producer. Obviously, a negotiation support system for the computation of counter offers may enhance the overall efficiency of the order promising system.

Attributes	Producer	Customer
Due date	late	early
Quantity	high	ordered amount
Price	low	high
Value-added services (VAS)	low	high

Table 1: Multiple issues in CTP and conflicting objectives

Attributes	Pre-Decision	Decision	Post-Decision
Due Date	Forecast arriving orders and build stock and production capacity accordingly	Produce in advance or negotiate later date	Evaluate forecast accuracy, and if necessary adjust forecast techniques
Quantity	Forecast arriving orders and build stock and production capacity accordingly	Reduce quantity or split it	Evaluate forecast accuracy, and if necessary adjust forecast techniques
Price	Conduct market research on competitive pricing	Reduce price to compensate for later due date and/or smaller quantity	Check if pricing was right
Value-Added Services (VAS)	Build up competence in customer services and preferences research	Offer customers value-added services to compensate for late delivery and/or delivery with smaller quantity	Assess customer satisfaction

Table 2: Suggested negotiation strategies to deal with CTP issues

The potential need for and benefit of introducing negotiation support to the domain of order promising has been already discussed in literature (e.g., Rupp and Ristic, 2004). Yet, most authors consider negotiation just for contracting before the ATP or CTP functions are executed, i.e., producer and customer settle on fixed values or intervals for quantities and due dates (e.g., Zadeh et al., 2003; Shin and Leem, 2002). Other authors claim in their research that negotiation processes have been implemented, but they do not explain or even formalize these processes in details (e.g., Makatsoris and Chang, 2008). To our knowledge, very few specific negotiation processes or systems have been proposed to support post ATP/CTP negotiation. Dudek and Stadtler (2005) discuss a system for negotiation-based collaborative planning between supply chain partners which are supplier and buyer. The supplier offers an initial quantity that can be revised by the buyer. Yet, their work focuses on collaborative partners and not a producer and end customers with divergent objectives.

Thus, there is potential for further research on enhancing the efficiency and effectiveness of order promising systems by introducing negotiation concepts and systems.

ARCHITECTURAL CONSIDERATIONS

We have developed a MAS prototype that consists of different agents representing the retailers business and its customers as shown in Figure 2. The implemented multi-agent system focuses on the decision column of Table 2, but it might be extended to include the pre- and post-decision phases. We provide in this section a short description of Multi-agent Systems, and then explain the architectural design of the system. The goal here is to offer the readers with sufficient background information to follow our discussion on using simulation runs to assess the usefulness of the proposed system.

Multi Agent Systems

Multi-agent systems (MAS) are information systems that have been of great interest in research over the last years. They consist of several intelligent agents which can exchange information or objects with each other. By doing so, agents can be designed to address complex problems which would be very difficult or impossible to solve with a single intelligent agent. In a distributed environment, MAS and their agents are naturally well suited to represent real-world organizations or units. Agent-based technology can today be found in a wide range of applications like disaster response and modeling social systems (e.g., Jennings et al., 1998).

The intelligent agents of a MAS share some important characteristics: They are mostly autonomous; They only have a limited, local view of the global environment; And there is no single agent that is able to control all the others. The agents are defined by their objectives, attributes and behavior (Julka et al., 2002).

System architecture

To illustrate our negotiation framework, we use the case of a computer retailer (producer). The customers place orders that typically consist of a specified number of computer systems. Since the retailer cannot accurately predict these orders and the configurations and reliability of ordered products, it has little choice but adopting a make-to-order environment.

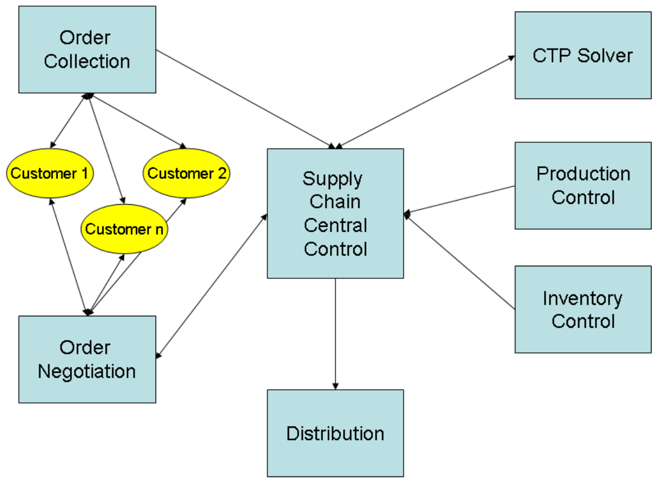

Figure 2: System architecture of a negotiation-assisted make-to-order environment

The *Order Collection* agent receives customer orders and passes them on to the *Supply Chain Central Control* unit. This agent communicates with *Production and Inventory Control* to get the necessary information to call the *CTP solver*. A linear program is used to decide whether or not the orders are accepted or rejected. These decisions are in turn returned to the *Central Control* agent. The latter attempts to derive alternatives for the rejected orders. These counter offers are then passed on to the *Order Negotiation* agent which uses an algorithm described later to modify the counter offers using new price and value-added services as terms of negotiation with the hope that they will be considered and accepted by the customer. The *Order Negotiation* then offers the counter offers to the customer who are asked to take positions. A UML sequence diagram of this CTP and negotiation process shows the lifespan of and communication between the agent processes (Figure 3).

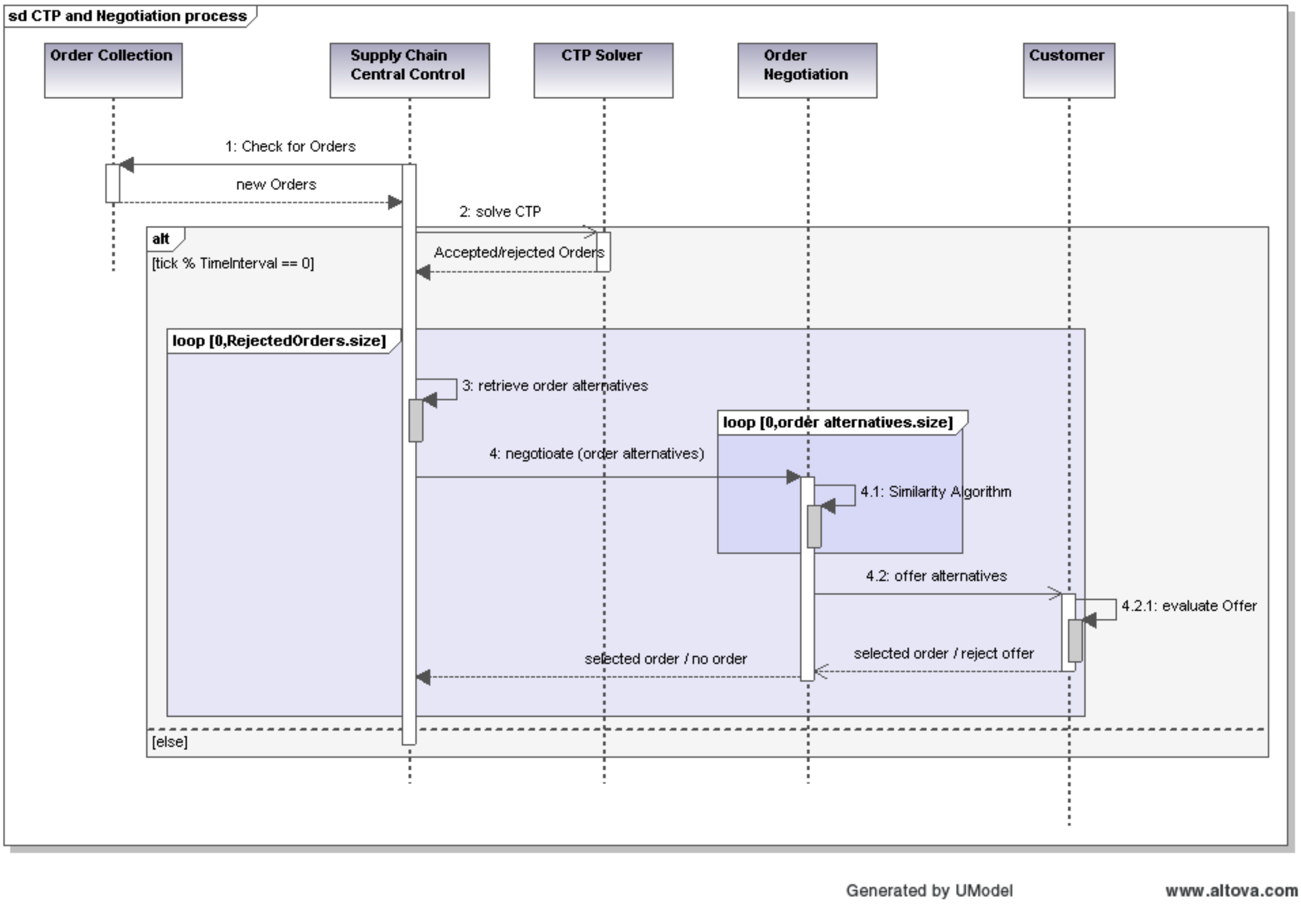

Figure 3: Sequence diagram of the CTP and negotiation processes

Software framework and tools

The MAS was implemented using the Repast Simphony framework (North et al.'s website). This Java-based environment provides a graphical user interface for running simulations within a MAS. The different agents are implemented using plain Java classes. For solving the CTP model the GNU Linear Programming Kit (GLPK Website) and its Java interface (GLPKJNI website) are used.

System details

This next section gives a more detailed description of the different agents and their behavior. It also focuses on the CTP model and the negotiation process in particular. In order to explain the different processes, it is first necessary to mention the attributes of an order which are listed in Table 3.

Attribute	**Description**
Arrival time	Time at which the retailer receives the order
Due date	Desired time of delivery (upper and lower bound)
Quantity	Number of units (upper and lower bound)
Price	Price per unit
Value-added services (VAS)	VAS associated with the order (e.g., extended warranty)
Assembly time	Time in periods to assemble the order

Storage cost	Storage cost of one assembled unit for one period
Penalty cost	Associated cost if the order is rejected
Number of components	Number of components per unit
Type of component	General type of the component (e.g. CPU, RAM etc.)
Manufacturer of component	Desired manufacturer / brand of the type of component

Table 3: Order attributes

At each step in time (e.g., one-hour interval), the Supply Chain Central Control (SCCC) agent checks if the *Order Collection* agent holds any newly arrived orders. These new orders will then remain at the SCCC until the end of a period. The period length is defined by the parameter *time interval*. If the end of a period is reached, SCCC will request information from the Production Control agent and the Inventory Control agent about production capacities, stock and planned replenishment of components. The SCCC agent then calls the CTP Solver and passes it these information as well as the new and already scheduled orders. In the solver agent a mixed-integer linear program (MILP) is solved to determine which orders should be fulfilled to maximize profit under given resources. For this purpose, we have implemented a modification of the model introduced by Chen et al. (2001) where we dropped the penalty cost for under-utilization and the security level in inventory for future important orders.

If the MILP solver returns an optimal solution, the accepted orders are scheduled for production and returned to SCCC. The SCCC agent also receives the new production capacities, inventory and rejected orders. The later are then marked for negotiation. From these orders alternatives have to be derived that can be fulfilled. In order to achieve this goal there are three options:

- Reduce quantity
- Move due date to a later time period
- Split order quantity

As such, there can potentially be three modified orders derived from one original order. Those alternatives are passed on to the *Order Negotiation* agent. Because these modified orders do not have the same utility value for the customer either the price has to be reduced, some additional services have to be promised or a combination of the two has to be applied for the customer to accept one of these new offers. This task is done by the negotiation agent using a modification of the similarity algorithm proposed by Faratin et al. (2002). Since we do not want to achieve a new offer that has the same utility value as the original one for the retailer but for the customer, we try to derive a new offer that is similar to the modified one and still has the same utility for the customer as the original one placed by him. But since the retailer does not know the customer's real utility function, he has to estimate this function. This estimation can be modeled as follows. We first assume a value function for each of the four order attributes (due date, quantity, price and VAS):

$$V(x) = \begin{cases} \dfrac{x_{\max} - x}{x_{\max} - x_{\min}} & if\ decreasing \\ \dfrac{x - x_{\min}}{x_{\max} - x_{\min}} & if\ increasing \end{cases}$$

In our context, a *decreasing* condition means that the score decreases as the value of x increases; An *increasing* condition means that the score increases as the value of x increases. The maximum and minimum values are shown in Table 4.

Attribute	Minimum	Maximum
Price	1500	1500 + 5 * cost of components
VAS	0	10 % of total price
Due Date	current period	current period + 21
Quantity	0	Upper bound of ordered quantity

Table 4: Minimum and maximum values for scoring functions

The calculated values are then weighted and added to get a combined utility value. Both retailer and customer use the same value function to calculate the combined utility but the weights are different. This is an attempt to deal with the uncertainty described above. After each counter offer is modified by the similarity algorithm they are communicated to the customer for his consideration. The customer then chooses the one offer with the highest utility value based on his weights. If the value $u_{\mathrm{mod}Order}$ is greater than that of the original order ($u_{oriOrder}$), the customer accepts this new offer. If it is lesser, there is still a probability $P(u_{\mathrm{mod}Order})$ that the modified order will be accepted:

$$P(u_{\mathrm{mod}Order}) = ae^{bu_{\mathrm{mod}Order}}$$

with a = 0.001 and $b = \dfrac{-\ln(a)}{u_{oriOrder}}$.

If one of the alternatives has been accepted, it is sent back to SCCC to be scheduled for production and delivery.

COMPUTATIONAL EXPERIMENTS

Using two scenarios, one basic and one advanced, we seek to demonstrate how the proposed system works and to establish and early assessment of the potentiality of augmenting the CTP function with negotiation support features. The parameters of the scenarios are order frequency, production capacity, stock and receipts, number of orders, the attributes of each order as mentioned in Table 3, as well as the different weights used in the utility function of the customers.

Basic scenario

This first scenario is used to test the feasibility of the system. It consists of 10 orders with two orders arriving in each period. The orders all consist of the same components, of just one unit and have an assembly length of one period. Production capacity is set to one unit per period and the delivery time is set in a such way that only one of two orders in a period can be accepted by the CTP function. Thus, the second order in a period has to be negotiated. Stock and receipts are set to a high value so that there will be no shortage of resources.

We ran the scenario twice, once without negotiation and once with negotiation. As expected, in the first run only 5 orders are scheduled for production yielding a revenue of $9,990. The second run applying negotiation yields 9 delivered orders and revenue of $13,500.

Advanced scenario

This scenario is intended to represent a case that is closer to a real-world situation. It consists of 300 orders arriving over 30 time periods. The arrival times are Poisson distributed with mean value of 0.1 periods. The number of units follows a normal distribution with a mean of 10 and a standard deviation of 2.5. The lower bound of the delivery time is also normal ly distributed with a mean of 3.5 time periods plus arrival time and 0.75 periods standard deviation. The upper bound is derived by adding a normally distributed random number (mean: 1 period; standard deviation: 0.5 periods) to the lower bound.

The production capacity is set to 40 units per period. The stock and receipts are shown in Table 5.

Type	Manufacturer	Cost	Storage cost	Initial stock	Replenishment / period
CPU	Intel	50	2.00	45	12
CPU	AMD	45	2.20	50	10
RAM	Samsung	60	1.10	65	13
RAM	Infineon	50	1.05	50	11
Graphics	NVidia	150	3.60	40	15
Graphics	ATI	120	3.70	35	20

Table 5: Inventory

Two simulations were run, one with a production capacity of 40 and the other with 60. To analyze the impact of negotiation, both scenarios were also run without the negotiation module. The results can be seen in Table 6.

	Production capacity 40	
	Without negotiation	**With negotiation**
Delivered orders	206	225
Rejected orders	94	75
Revenue	$2,418,392	$2,496,219
Newly accepted orders		75
Previously accepted orders that were rejected		56
	Production capacity 60	
	Without negotiation	**With negotiation**
Delivered orders	266	284
Rejected orders	34	16
Revenue	$3,395,211	$3,471,117
Newly accepted orders		27
Previously accepted orders that were rejected		9

Table 6: Results using advanced scenarios (production capacity 40)

In both scenarios, the number of accepted orders and the revenue could be increased when using negotiations with the CTP function. The average revenue per order decreases with negotiations because the price is reduced during that process to make offers that are more likely to be accepted by the customers. Additionally, some orders that were accepted without negotiation are rejected with negotiation to allow better counteroffers by the producers. In order words, thanks to the negotiation algorithms that take into consideration the utilities of both buyer and supplier, production capacity is better utilized as soon as it is available.

SUMMARY

Order promising through ATP and CTP is a well studied function in operational supply chain management and many of the ATP/CTP functions have been implemented in different kinds of commercial resource planning software systems. Yet, little research is done about the needed negotiation when customer orders cannot be fulfilled. We present a multi-agent based system to simulate order promising in a make-to-order production environment with an automated negotiation process. We discussed how counteroffers can be computed and analyzed. Results of our simulations suggest that, based on two typical example scenarios, the number of rejected orders can be reduced while the overall revenue increased when introducing negotiation concepts.

The benefit of our work is that the proposed system can automatically suggest alternative counteroffers that are likely to be accepted by the customer. We are not aware of any formalized processes incorporating negotiation concepts to the CTP function with this focus. Thus, we can hardly compare our work to other systems. In this paper, our main goal was to discuss the use of negotiation processes, and we chose not to compare our CTP-Model with other CTP-Models as it is not the essential element of our system and could easily be interchanged with other modules or algorithms. Nonetheless, we intend to further test the proposed system with more simulations before testing it with real-life data. The simulation results are presented as a proof of concept, not as a comprehensive proof of effectiveness. A much broader set of simulations is needed to support the finding that the overall revenue of the producer can be increased when offering 'good' counteroffers to the customer in case of scarce production or limited delivery capacities. Of course, the negotiation system's effectiveness does also depend on the properties of arriving customer orders. If the producer uses all the production capacity that is available at the time of the placement of the order with low margin customer orders early due to successful negotiation, high margin orders may need to be rejected in later time periods.

ACKNOWLEDGMENTS

The authors would like to thank the DAAD for supporting the joint research project on "Analytical and Computational Modeling in Logistics and e-Procurement" of the University of Hawaii at Manoa and the RWTH Aachen University under grant number D/07/10552 (2008-2009).

REFERENCES

1. Ball, M. O., Chen, C.-Y. and Zhao, Z.-Y. (2004) Available to Promise, in: Simchi-Levi, D., Wu, D. and Shen, M. (Eds.) *Handbook of Quantitative Supply Chain Analysis: Modeling in the E-Business Era*. Kluwer Academic Publishers, Boston, p. 447-484.
2. Bichler, M., Kersten, G. and Strecker, S. (2003) Towards a Structured Design of Electronic Negotiations, *Group Decisions and Negotiation* 12, p. 311{335.
3. Chen, C.-Y., Zhao, Z.-Y. and Ball, M. O. (2001) Quantity and Due Date Quoting Available to Promise, *Information Systems Frontiers* 3, Nr. 4, p. 477-488.
4. Dudek, G. and Stadler, H. (2005) Negotiation-based collaborative planning between supply chain partners, *European Journal of Operational Research* 163, p. 668-687.
5. Faratin, P., Sierra, C. and Jennings, N. R. (2002) Using Similarity Criteria to Make Issue Trade-offs in Automated Negotiations, *Artificial Intelligence* 142, p. 205-237.
6. GNU Linear Programming Kit, Version 4.: http://www.gnu.org/software/glpk/.
7. Interface, GLPK 4.8 J.: http://bjoern.dapnet.de/glpk/.
8. Jennings, N. R., Sycara, K. and Wooldridge, M. (1998) A Roadmap of Agent Research and Development, *Int. Journal of Autonomous Agents and Multi-Agent Systems*, 1 (1). pp. 7-38.
9. Julka N. Srinivasan R. Karimi I., (2002) Agent-based supply chain management-1: framework, *Computers and Chemical Engineering*, Volume 26, Number 12, 15 December 2002 , pp. 1755-1769(15)
10. Kilger, C. and Schneeweiss, L. (2005) Demand Fulfillment and ATP. In: Stadtler, H., Kilger, C. (Eds.): *Supply Chain Management and Advanced Planning*. 3rd. Springer, p. 179-195.
11. North M., Howe, N. C. T., Howe, H. T. and Vos, R.: The Repast Symphony runtime system. http://repast.sourceforge.net/.
12. Makatsoris, C. and Chang, Y. S.: Beyond CRM (2008) A System to Bridge the Gap between the customer and the Global Manufacturing Supply Chain, *Proceedings of the 41st Hawaii International Conference on System Sciences*.
13. Rupp, T. and Ristic, M. (2004) Determination and exchange of supply information for co-operation in complex production networks, *Robotics and Autonomous Systems* 49, p. 181-191.
14. Sadeh, N., Hildum, D. and Kjenstad, D. (2003) Agent-Based E-Supply Chain Decision Support, *Journal of Organizational Computing and Electronic Commerce* 13, Nr. 3, p. 225-241.
15. Shin, K and Leem, C. S. (2002) A reference system for internet based inter-enterprise electronic commerce, *The Journal of Systems and Software* 60, p. 195-209.
16. Stadtler, H. (2005) Supply chain management and advanced planning - basics, overview and challenges, *European Journal of Operational Research* 163, p. 575-588.