THE MULTI-AGENT APPROACH TO SUPPLY CHAIN MANAGEMENT OPTIMIZATION

Thanh-Tung DANG , Baltazár FRANKOVIČ, Ivana BUDINSKÁ
Institute of Informatics, Slovak Academy of Sciences, Bratislava, 84507, Dúbravská c. 9, Slovakia,
E-mail: (utrrtung, utrrfran, utrrbudi)@savba.sk

SUMMARY

The aim of this paper is to analyze the dynamic process of agent negotiation within a supply chain management. To solve the declared problem, some methods to perform negotiation are proposed. Agents will be able to flexibly choose strategies while negotiating. Both of the own and global criteria are included within agent deliberation. A model of the negotiation of 3 agents, which has been developed on the basis of the discussed theories, is presented. The model is applicable for SME (Small and Medium Enterprises) and other applications, where negotiation is necessary, and/or useful.

Keywords: *Supply-Chain Management, agents, MAS, negotiation, and optimization*

1. INTRODUCTION

The Multi-Agent System (MAS) approach is one of the powerful manners for simulating, resolving or managing various large-scale systems. MAS has been investigated and applied in many applications and despite of achieving some significant results, the MAS approach, in general, has still a lot of open issues to be resolved [9], [11].

A supply chain management (SCM) problem, chosen as the case study in this paper, is one of ideal applications forapplying the MAS approach. An SCM is understood as a set of enterprises included in processing customer's demands. To simplify we assume that each enterprise is controlled by two persons: a manager and a scheduler. The scheduler's task is to optimize the production process to fulfill the order formulated by the manager. The scheduler is responsible for internal optimization running and the manager, on the contrary, tries to negotiate with other ones from the external environment to achieve better results for his/her enterprise. Such distribution of tasks between the manager and the scheduler might reduce the responsibility for each of them; and they can concentrate on one field and do not have to hold too much information.

The main interests concentrate on cooperation among these managers, which are replaced by software (SW) agent in our model (in the rest of this paper we will use the symbol "agent"). Each agent represents a manager and possesses all strategies or policies that the manager should use while cooperating with other colleagues. Agents join cooperation with the purpose to improve their performance and their task is to negotiate and arrange for a compromise, which satisfies the imagination of most participants.

Many authors and papers dealing with the same topic have shown that there is any generically best strategy for making negotiation for all situations. That fact has motivated us to try to design and implement a set of basic negotiation strategies, rather than some fixed ones; and agents can flexibly switch among these strategies and choose the appropriate one for their current situation.

This paper is organized as follows: Section 2 presents a brief formulation of the problem solving. Section 3 introduces two generically basic negotiation schemes. Section 4 discusses technical realization of the proposed methods. Section 5 presents some simulation results. Comparison with other related works is presented in Section 6.

2. PROBLEM FORMULATION

2.1 General formulation

To simplify for later use in the rest of this work, let us denote:

- a set of agents: $\{A_i | \, i=1,..,n\}$.
- a set of alternatives (methods for execution) $\boldsymbol{AL} = \{a_i | \, i=1,..,m\}$.
- a set of constraints ***Co*** and an initial demand: ***De***.
- a plan that each agent proposes: $\Omega = \{u_1,...,u_n\}$, where $u_j|j=1,.., n \in \boldsymbol{Al}$ is an alternative that agent A_j has to do according to plan Ω.
- a set of all possible plans $Set\_plan = \{\Omega\}$
- local criterion functions – expected (monetary) profits that an agent could gain: q_i, $i=1,..., n$.:

$$q_i = f_i(\mathrm{De}, \mathrm{Co},\Omega) = f_i(\mathrm{De}, \mathrm{Co}, u_1,...,u_n) \quad (1)$$

- a global criterion function:

$$Q(\Omega)= \sum_{i=1}^{n} q_i = \sum_{i=1}^{n} f_i(De, Co, \Omega) \quad (2)$$

- within negotiation, let $q^i_j|i,j=1,.., n$ be a profit that agent A_j will gain in the plan proposed by agent A_i.
- the main aims of agents' cooperation are:
  - to achieve the plan with as good quality ($Q(\Omega)$) as possible,
  - to satisfy each agent's requirement ($q_i|_{i=1,...,n}$) as much as possible.

Before starting the discussion about agents' cooperation within SCM, we should present some general characteristics of the SCM.

2.2 Agent negotiation within Supply Chain Management

Without loss of generality, let us assume that each scheduler is able to calculate his/her optimal production plan, when s/he has all information of other agents' choices. The agent's task, in this case, is only to improve its expected values by persuading other agents to accept its proposition. The scheduler's answer involves all needed data for agent's calculation: e.g., *quality of each product, the time when the order will be completed, the time when machines will be idle*, etc. The corresponding agent will then calculate from these data (according to its predefined criterion function) how much it will gain by applying this plan and decide the next strategy for cooperation with other agents within the SCM.

To understand, let us give a short illustrating example:

Example 1: Let 3 agents {A_1, A_2, A_3} be involved in the supply chain: (*producer, transporter and distributor*). There are also 2 alternatives { a_1 = each of them will work in 4-hour shift, or a_2 = in 8-hour shift}. It is also assumed that each choice of an agent influences the execution of other ones. There are 8 possibilities that might occur; and let's assume that after receiving the results from corresponding schedulers, agents could calculate their possible profits as shown in Table 1.

Order	Plans	q_1	q_2	q_3	$Q(\Omega)$
1	{1,1,1}	25	30	45	100
2	{1,1,2}	35	30	50	115
3	{1,2,1}	0-	0-	0-	0
4	{1,2,2}	45	20	30	95
5	{2,1,1}	45	15	15	75
6	{2,1,2}	40	20	30	90
7	{2,2,1}	25	20	20	65
8	{2,2,2}	30	35	40	105

Tab. 1 The profits for each agent

The results in Table 1 show that plan 2 - {1,1,2} is the best one with respect to the global criterion function (Q =115), but agent A_1 should choose plans 4 or 5, agent A_2 prefers to choose plan 8, and agent A_3 plan 2, since in these plans their expected profit is maximal. In some cases (e.g. plan 3) the expected profits might be zero, if plans are unrealizable due to technical reasons (the set of constraints ***Co*** is not fulfilled).

Due to the fact that each agent tends to gain as much as possible, negotiation is necessary to achieve any agreement. Making compromise guarantees the existence of a final solution, but, on the other hand, this solution usually is not the optimal one, wished by each agent. The final solution (i.e., a plan for execution in this case) that agents agree with, is usually the Nash or Pareto-optimal one [2], [14]. Concerning the characteristics of SCM, where agents must accept any concrete solution, achieving the Pareto-optimal solution is chosen as the purpose of agent negotiation. Before defining the Pareto-optimal solution, the following definition is introduced.

Definition 1: We say that plan Ω_1 dominates plan Ω_2 if and only if:
$\forall\, i = 1,.., n$: $q_i |_{\Omega_1} \geq q_i |_{\Omega_2}$ and at least one value i exists, where the inequality occurs.
For example, plan 8 dominates plan 7 in Example 1 (see Table 1).

Definition 2: Plan Ω is considered as a Pareto-optimal plan if there is no other plan, which dominates that one.

In Example 1, plans 2, 4 and 8 are Pareto-optimal ones, since there is no plan, which dominates them. Pareto-optimal solution is such a state where increasing profit of any agent will cause decreasing the profit of others. For that reason, when a Pareto-optimal plan is achieved, no agent is motivated to get better results by choosing another plan. Such a state could be considered as a stable state of agent negotiation, because it does not motivate the agents to continue the negotiation process. There might be more than one Pareto-optimal plan, but decision which of them is better could be made by voting (where each agent will evaluate a plan by any weight and the one that receives the highest weight is chosen) or based on the global criterion agreed by all agents.

Due to the fact that there might be a number of different manners to negotiate, the main goal of this paper is to design and implement several, not fixed, basic strategies. These strategies could also be combined to create new ones applicable to different situations or applications. In the following section some basic strategies for negotiation are presented.

3. NEGOTIATION STRATEGIES IN SCM

3.1 Negotiation scheme by voting

When an agent chooses a plan, two important factors are usually taken in consideration, namely:

- the personal profit the agent can get ($q_i|_{i=1,\ldots,n}$), and
- the global effect that this plan will bring for all participants (Q (Ω)).

The balance between these factors depends on each agent evaluation. This leads to the following definition of a common method for evaluating plans.

Definition 3: $\forall i \in [1, n]$, let {β_1^i, β_2^i} be the weights by which agent A_i evaluates the personal and the global criteria. β_1^i, $\beta_2^i \geq 0$ and $\beta_1^i + \beta_2^i = 1$. The quality assessment of plan Ω from agent A_i's point of view is defined as:

$$eval_i(\Omega) = \beta_1^i\, q_i|_\Omega + \beta_2^i\, Q(\Omega) \quad (3)$$

Equation (3) shows a simple manner to assess a quality of each plan; however, agent's decisions might be influenced by many other factors, e.g. commitments with other agents, etc. They are not considered in this paper.

On the basis of Definition 3, the following negotiation strategy is proposed (see Fig. 1).

The voting method for negotiating gives each agent a chance to express its opinion about the proposed plans. The value that an agent sets to each plan, expresses its willingness to accept the proposed plan. Each agent can have a different manner for setting these values, but it might be better for the final evaluation, if agents agree with any common method for marking. The method for evaluating plans presented in Step 2.c is based on the assumption that plans are evaluated by the probability of acceptance, or by the fuzzy "conjunction" operator (e.g., [15]). Schedulers might need more time for calculating optimal plans than negotiation among agents, and therefore

Phase 0: *initialization*: setting values $\{\beta_1^i, \beta_2^i\}$, $i\in[1, n]$. Let Θ be a set of examined plans. Initially, $\Theta = \{\varnothing\}$.

Phase 1: *personal agent decision*:

- Choosing randomly k different plans and forwarding them to schedulers to calculate. Adding the selected plans to set Θ (k is a constant agreed by all agents).
- Each agent chooses a plan (or plans) $\Omega_i^*|_{i=1,\dots,n}$ which satisfies a criteria: $\Omega_i^*|eval_i(\Omega_i^*) = \arg\max_{\forall\Omega_j\in\Theta} (eval_i(\Omega_j))$.

Phase 2: *Negotiation and selection* -(*voting*)

a. Agents have to agree with using the voting method for choosing a final plan.

b. Each agent receives minimally n suggested plans (including its plan); let us say s ($s \geq n$): for agent A_i, within plan $\Omega_j^*|_{j=1,\dots,s}$ its expected profit is $q_j^i|_{j=1,\dots,s}$. Agent A_i might evaluate plan Ω_j^* by a value $\alpha_{ij} = \frac{q_j^i}{\sum_{r=1}^{s} q_r^i}$, where $\forall i\in[1,n]$:, $\sum_{j=1}^{s}\alpha_{ij} = 1$ or by other manners.

c. The evaluation of each plan $\Omega_j^*|_{j=1,\dots,s}$ is calculated by: $vote(\Omega_j^*) = \prod_{i=1}^{n}\alpha_{ij}$, or $vote(\Omega_j^*) = \min\{\alpha_j\}|_{i=1,\dots,n}$.

d. The plan with the highest evaluation will be selected: $\Omega_{\text{acceptable}}|\ vote(\Omega_{\text{acceptable}}) = \arg\max_{j=1,\dots,s} vote(\Omega_j^*)$.

e. Comparing the new obtained plan with the currently best one (comparing values $Q(\Omega)$) and repeating Step 1 until all possible plans are explored (Θ = *Set_plan*) or agents agree to stop (e.g., due to the time limitation).

Fig. 1 Negotiation and Selection based on Voting (NSV)

in each cycle only several plans (defined by constant k in Step 1) are chosen for calculation. This process can continue until all plans are examined or the time available for negotiation and calculation expires. Since the aim of agents is to achieve the Pareto-optimal solution, the following proof can confirm that NSV (Negotiation and Selection based Voting) converts to such a solution.

Theorem 1: NSV achieves a Pareto-optimal plan within a set of the examined plans.

Proof: Assume that among already examined plans, there is any plan Ω_1, which dominates the final one agreed by all agents ($\Omega_{\text{acceptable}}$), i.e. $\forall i\in[1,n]$:

$q_i|_{\Omega_1} \geq q_i|_{\Omega_{acceptable}}$ and $\exists\, i\in[1,n]$: $q_i|_{\Omega_1} > q_i|_{\Omega_{acceptable}}$

It follows $Q(\Omega_1) > Q(\Omega_{\text{acceptable}})$. From Equation (3), it is easy to verify an inequality: $\forall i\in[1,n]$: $eval_i(\Omega_1) > eval_i(\Omega_{\text{acceptable}})$. That fact leads to the conclusion that after Step 1, $\Omega_{\text{acceptable}}$ cannot be one among the candidates for negotiation; consequently, it cannot be a final solution, too. Confrontation.

Theorem 1 considers already examined plans only, since agents might finish negotiation earlier than all possible plans are investigated.

Besides using a simple method presented by Equation (3) for measuring a plan quality, agents could exploit other manners to choose plans based on the degree of satisfaction. The methods presented in the next section are based on this principle.

3.2 Negotiation based on the degree of satisfaction

Each plan fulfils agent's requirements to a certain degree. To simplify the negotiation process, let us consider that an agent is maximally satisfied if the selected plan brings the maximally possible profit for it. To maximize the satisfaction, each agent usually tends to choose such plans, in which its expected profit (q_i) is close to the maximal value. Of course, such choice initiates a lot of conflicts and agents usually have to make some concession in order to guarantee any final compromise. To measure the agent's satisfaction we should define the *relative satisfaction* of agents as follows:

Definition 4: *Relative satisfaction* of agent A_i is defined by a ratio between the real and the maximal profit that agent A_i can receive.

$$relsat_i = q_i|_{\Omega_{selected}} / \arg\max_{\forall\Omega}(q_i|_{\Omega}) \tag{4}$$

where Ω_{selected} is the selected plan.

Assumption 1: Let $satisfy_i$ be a minimal bound of the *relative satisfaction* that agent A_i is willing to accept, i.e., A_i accepts only such plans Ω fulfilling the condition:

$$relsat_i|_{\Omega} \geq satisfy_i. \tag{5}$$

Negotiation could leads to an acceptable compromise, if the *relative satisfaction* of each agent achieves minimally the predefined bound $satisfy_i$. That leads to the following definition.

Definition 5: Plan Ω is considered to fulfill all agents' *relative satisfaction*, if $\forall\ i \in [1,n]$: $relsat_i|_\Omega \geq satisfy_i$.

Since all agents have the same importance and play the same role within SCM, they should set the same value of all coefficients $satisfy_i$ when choosing plans. In order to guarantee the existence of the final solution, the initial bound should be changeable, but how to change this bound is another element of agent negotiation. There are many manners applicable for modifying coefficients $satisfy_i$, but due to the restricted frame of this paper, only several of them are considered. The simplest method is that, if a plan is not achieved, then all agents agree to decrease their minimal bound to a newly lower value and repeat the negotiation process. Practically, that means that, each agent does not have to observe what other agents choose and they work simultaneously; as a consequence, negotiation might be finished very quickly.

Beside the mentioned method, there are other different methods based on an assumption that only some agents, not all, make modifications of their demands (coefficient $satisfy_i$). Each agent is able to specify a manner how it modifies its demand. The first method belonging to this category might be as follows. In each round, the predefined number of agents has to modify their coefficients $satisfy_i$. These agents might be selected at random or in sequence; and this process continues until any plan is achieved. Another method for choosing agents to make modifications is based on their personal requirement. Agents with the highest requirement (at least acceptable value q_i) have to make a concession (decrease coefficient $satisfy_i$). The next possible method is based on the *risk* coefficient. The *risk* is a parameter that agents assess each proposition posted by any other agent, whether they will accept or reject it (see [5] for more detail). Each personal requirement (a minimal value of q_i) is considered as an agent's proposition, and it is evaluated by the rest of agents. The *risk* of each proposition is defined by combination of all agents' *risk* evaluation and the agent(s) with the highest *risk* proposition has/have to make concession.

Opposite to the *risk* coefficient we define an *accept* parameter. This parameter expresses the willingness of agents to accept the proposition posted by any other agent, because each proposition brings different effects (profits) to agents, and therefore their opinion about the proposition might be dissimilar. Here, agent's proposition is the minimal demand (q_i) of plans that the agent is willing to accept. Let $q_i^* = satisfy_i * \arg\max_{\forall\Omega}(q_i\,|_\Omega)$ be the minimal demand of agent A_i, and let q^*_{ji} be the highest profit that agent A_j can get within all plans, which satisfy agent A_i's minimal demand. Agent A_j assesses A_i's proposition as follows:

$$accept_{ji} = q^*_{ji} \,/\, \arg\max_{\forall\Omega}(q_j\,|_\Omega) \tag{6}$$

The assessment of A_i's proposition by all agents is defined as follows:

$$accept_i = \prod_{j=1, j\neq i}^{n} accept_{ji} \tag{7}$$

Agent(s) which has/have the smallest *accept* coefficient has to make concession toward the rest of agents.

Based on these discussions we should propose the following method for negotiation based on the coefficient of *relative satisfaction* (see Fig. 2).

Let $f_i(.)$ be a function that agent A_i uses to specify a new value of coefficient $satisfy_i$ – $f_i(.)$ might be a function of one or many variables, to simplify, only a function of one variable is considered in this paper.

Phase 0: *initialization* - Each agent defines a method for making concession - $f_i(.)$, let Θ be a set of examined plans. Initially, Θ = {∅}. Each agent starts with the same value $satisfy_i$.

Phase 1: *Personal agent decision*

- Choosing *k* randomly different plans and forwarding them to schedulers to calculate. Adding the selected plans to set Θ (*k* is a constant agreed by all agents).
- Each agent chooses all plans fulfilling a condition in Equation (5).

Phase 2: *Negotiation and selection*

- Agents broadcast to all other ones their personal values satisfyi (it might not be necessary in the first round).
- Agents choose plans satisfying condition presented in Definition 5.
- Among satisfied plans (if any), the one with the highest global quality ($Q(\Omega)$) is selected for a solution.
- Comparing the newly obtained plan and the currently best one (comparing values $Q(\Omega)$) and return to Phase 1 until all possible plans are explored (Θ = *Set_plan*) or agents agree to stop (e.g., due to the time limitation).
- If Phase 2 does not achieve any solution, then, phase 3 starts.

Phase 3: *making concessions*

- choosing agent(s) to make concessions:
  - Sequentially,
  - The highest requirement,
  - The highest risk of rejection,
  - The least chance of acceptance, etc.
- The selected agents set new values of satisfyi according to their predefined functions *$f_i(.)$*, then, add new plans satisfying the reduced requirements.
- Return to Phase 2.

Fig. 2 Negotiation and Selection based on the degree of Relative Satisfaction (NSRS)

The most important factor of the proposed negotiation scheme is the willingness of each agent to make concessions. To avoid a complication when

Phase 2 does not reach any solution and agents argue with each other to appoint which of them have to concede, agents should decide about a common mechanism of selection before starting negotiation. Agents, which are selected by using the defined mechanism, have then to decrease their demands, but according to their individual method (via function $f_i(.)$). In the simplest case, when all agents agree to decrease their coefficient $satisfy_i$ to a newly lower value, the agents could exploit the same function $f_i(.)$ (called $f_0(.)$), which might be defined as follows:

$$satisfy_j = f_0(\varepsilon) \quad (8)$$

where $satisfy_j$ is the minimal bound of the *relative satisfaction* of all agents in round j of negotiation and $\varepsilon \in$ R is a variable of function $f_0(.)$. The dependence between variable ε and the quality of obtained plan is expressed by the following theorem.

Theorem 2: Let $Q_\varepsilon(\Omega)$ be the global quality of the plan obtained when $\forall i \in [1,n]$: $satisfy_i = f_0(\varepsilon)$. If $f_0(\varepsilon)$ is a monotonously decreasing function in R, then $\forall\ \varepsilon_1 > \varepsilon_2$, $Q_{\varepsilon_1}(\Omega) \geq Q_{\varepsilon_2}(\Omega)$, and vice versa.

Proof: see [**3**].

Because the main aim of agent negotiation is to achieve a Pareto-optimal solution, the following theorem can confirm that NSRS fulfils the designed aim.

Theorem 3: NSRS converts to a Pareto-optimal plan within a set of the examined plans.

Proof: Let Ω_0 be the plan obtained by applying the NSRS protocol. If there is any plan Ω_1 among the already examined ones that dominates Ω_0, then: $\forall i \in [1,n]$: $q_i |_{\Omega_1} \geq q_i |_{\Omega_0}$ and $\exists i \in [1,n]$: $q_i |_{\Omega_1} > q_i |_{\Omega_0}$. It is clear from Equation (4) that $\forall i \in [1,n]$: $relsat_i |_{\Omega_1} \geq relsat_i |_{\Omega_0}$, and therefore plan Ω_1 has to be preferable over plan Ω_0 among the examined ones. Confrontation.

Similarly to Theorem 1, Theorem 3 considers the examined plans only, since parameters of the unexamined plans are not known, and therefore agents cannot assess their quality.

The main difference between the NSV and NSRS protocols is the manner how agents select favorite plans. NSV always achieves any solution in each round of negotiation, but NSRS might not. When agents determine too high personal demands ($satisfy_i$), their negotiation could lead to a conflict, and as a result, an agreement is not reached. To reach a compromise, NSRS forces agents (all or some) to decrease their demands. Selecting these agents is an important point of negotiation. Beside that, the willingness of agents to decrease is also an important factor, which influences the speed of the negotiation process. Clearly, the more agents decrease their demands, the more quickly negotiation can reach a final compromise; but, these features depend on each agent behavior or calculation.

Summary. In this section, several basic negotiation schemes have been presented, which are applicable to each realistic situation. Agents can flexibly switch among these schemes and choose the most appropriate one to apply to their current situation.

In fact, agents can negotiate to choose the most preferable plan, only when all parameters of the plans are known. With n agents and m alternatives there might be m^n potential plans. Getting required parameters of all these plans is usually a very hard task, and schedulers might need much more time for calculation than for the negotiation process among agents. For that reason, all negotiation protocols are suggested with the purpose to achieve any sub-optimal solution in each round, to ensure the existence of a final solution when agents stop negotiation.

One difficult problem that has not been considered in this paper is the process when schedulers search for optimal plans. Next problem is storing temporal data during negotiation. All temporal information has to be stored for later rounds, essentially in the NSRS method, when agents repeatedly modify their demands. All the mentioned topics will be dealt with in the future work.

The following sections will discuss implementation and experiments of the proposed negotiation methods.

4. IMPLEMENTATION

The proposed methods have been implemented and experimented with some simply realistic situations. Implementation is realized in C++, PVM (Parallel Virtual Machine) (with some libraries provided by [12], [13]) under Linux, and the Internet is used as a medium for communication exchanges. Each agent is implemented with all the proposed strategies, but there is one more agent (called *master*), whose task is to coordinate all agents activities, to activate negotiation, to specify some extra requirements defined by the users, e.g., deadline when agents have to finish, or some parameters that the final plan has to fulfill, etc. Communication exchanges among agents are performed via the *master* agent. This agent stores also all temporal solutions achieved during negotiation.

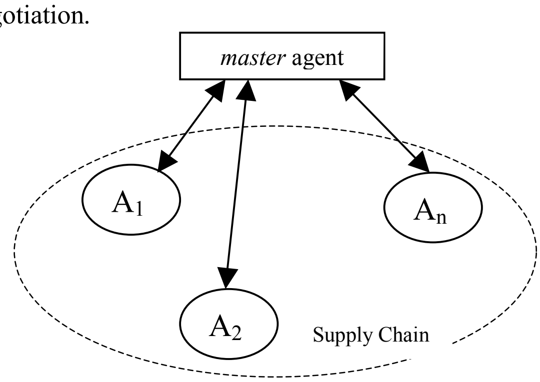

Fig. 3 Agent negotiation model

Schedulers are essentially passive members within SCM, which are invoked by the corresponding agents. Assume that each scheduler possesses all needed information about SCM and is able to calculate an optimal plan when the chosen alternative of each agent is known. To simplify, a scheduler is implemented, as one function of an agent, which is called by the agent when getting parameters of potential plans is required.

Negotiation works as follows. At the beginning, agents decide on the common negotiation protocol (NSV or NSRS). If the NSRS protocol is chosen, agents have to decide about the mechanism for selecting agents to make concessions. Negotiation is performed according to the chosen protocol. After receiving results from schedulers, agents send their choices and personal demands to the *master,* and that agent extracts the most preferable plan of the received ones by applying the rules defined in the protocol. The *master* informs all agents about the achieved results (a quality of selected plan or failure, if a solution does not exist) and coordinates agents' activities according to the defined protocol, until agents agree to stop.

5. FROM THEORY TO PRACTICE

For illustration, let us take an example consisting of three agents creating the supply chain for food distribution: *producer*, *distributor* and *seller agents*. Agents are situated in different places, with different situations and they work in the on-line mode (always in contact). The problem occurs when a customer has arrived. Agents start to create plans and search for the optimal solutions: the aim is to satisfy the customer's demand best as they can; concurrently, each agent will be contented with the received profits.

Of course, execution of one agent influences the remaining ones calculation; therefore, one agent cannot calculate all possible variants without recognizing what other agents will do. Agents could use the above mechanisms to select favorite plan and to make negotiation. Since there are many different alternatives that could be chosen for execution, the finally achieved solutions might be different, depending on a set of the examined plans.

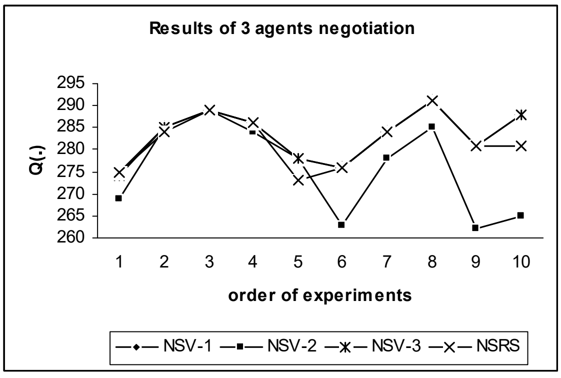

Fig. 4 Results of 3-agent negotiation

On the basis of the above negotiation protocols and real values, which were, however, generated at random, the simulation was made in our model and various results were obtained. For illustration, some simulation results are selected and shown in Fig. 4, in the case when each agent has 10 alternatives to work (q_i were generated randomly in interval [0,100]).

Remark: NSV-1, 2, 3 are the results achieved by applying the NSV protocol with corresponding coefficients $\{\beta_1, \beta_2\}$ = {1,0}, {0,1} and {0.1,0.9} for all agents. The results shown in Fig. 4 denoted by NSRS are achieved by applying the NSRS protocol, when all agents use the same function to decrease their demands at one time.

Each proposed protocol might be modified in various ways to create many different negotiation strategies, e.g., a lot of different strategies could be created by setting different values to coefficients $\{\beta_1, \beta_2\}$- within the NSV protocol, or by defining different functions $f_i(.)$- within the NSRS protocol. As a result, with the same set of data, the finally achieved solution of negotiation might be different, depending on each agent's choice. On the basis of simulated results, some conclusions could be derived, namely: using the NSV protocol, when one agent prefers its personal requirements over the global one ($\beta_2 >> \beta_1$), leads to the final compromise, which although being the Pareto-optimal plan, has lower quality than when all agents make opposite choices (all prefer the global criterion over personal ones - $\beta_1 >> \beta_2$). In the NSRS method, if the selected agents do not decrease their demands enough, the negotiation process might be very long, until any agreement is reached. Thus, if there is a selfish agent, which always prefers own requirement, the negotiation process will be very exhaustive and will lead usually to a low quality solution. The agent's willingness to make concession toward others has a very important role and a large influence on the quality of the final solution. Unfortunately, that property depends on agent's character that is usually unknown within the real world.

Each agent can express its demand in many ways (through coefficients $\{\beta_1, \beta_2\}$ or a function $f_i(.)$); consequently, negotiation might develop into a variety of scenarios, and therefore it might lead to different solutions too. Since the behavior of each human manager is very sophisticated, describing it by using standard mathematical formulations is a very hard problem. For that reason, we try to introduce many basic mathematical functions to allow describing human agent's behavior. Human agent's behavior could be approximated by combination of the predefined functions. However, each agent decides autonomously which combination is the closest one to its behavior; as a result, negotiation will be very flexible and close to realistic scenarios. Such simplification allows easier calculation of optimal strategies leading to the most appropriate solutions, which fulfill all agents' demands.

6. COMPARISON WITH OTHER APPROACHES

Both generally proposed negotiation methods always lead to the Pareto-optimal solutions, which is the main aim of agent cooperation. Such an idea has been discussed in many papers [2], [7], [8], [10], and [11]. A lot of strategists tried to propose various negotiation strategies to allow agents to achieve as high profit as possible. The most used technique in these papers is the Game theory based on an assumption that each plan with any probability could be acceptable to a final solution. On the basis of values of these probabilities and the profits of each plan, an agent calculates the plan, which brings it the highest *expected* (not real) profit. Negotiation based on such a principle usually converts to a Nash-equilibrium state (the definition of the Nash-equilibrium is introduced in [14]). That state is similar to the Pareto-optimal solution, since profit of one agent could be increased only at the cost of decreasing some other agents' prospectus. On the other hand, the necessary condition that enables applying this approach is that probabilities of the acceptance of all plans must be known before starting negotiation. Otherwise, agents have to use some learning techniques to specify these values (e.g., Q-learning, incremental learning [16]). The major disadvantage of the mentioned approach is that agents have to know the profits that they can get in each plan; that is equivalent with forcing schedulers to calculate all possible plans in advance. Such a requirement is not very practical, due to the exhaustive calculation that schedulers need to perform to achieve the desired data. On the basis of the above analysis, applying the Game theory to the chosen case of study is not considered as optimal.

Finding Pareto-optimal solutions is also used in other papers e.g., [2], which deals with a different problem – agent coalition. These papers use the same principle, that is, each agent has to make concession. To specify which agent has to concede, the parameter *excess* is defined. The parameter *excess* expresses the strength between two agents, and the stronger one can require the second one to make concession. That idea was originally introduced for the game among *n* persons, but the necessary condition of this approach is to know all the payoff functions of each agent (it is similar to the profits that agents can get in our case of study). Fulfilling that condition is too difficult, essentially when each agent uses a complicated function to calculate its expected payoff (or profit, here). Such a conclusion was mentioned not only in that paper, but also in [7], [8], and to simplify the problem solving numerous ways were proposed to approximate these payoff functions. Agents really start negotiation after calculating all *excess* parameters. The methods proposed in this paper do not require such a complicated calculation, which practically takes much more time than the negotiation process alone; and therefore agents apply more practical approach, in which only a small number of plans are examined in each round. As a result, agents always can achieve any solution; even the time available for calculation is limited.

Other papers, e.g. [1], proposed some negotiation strategies based on social mechanisms as in human societies. Such approaches, however, are very difficult for implementation, and they may be effective when agents are situated in environment where there is no agreed-upon, well-defined interaction mechanism.

7. CONCLUSION

This paper deals with the problem of agent negotiation, especially within the frame of SCM. Concerning the characteristics of SCM, two basic negotiation protocols have been proposed. Within the proposed methods, agents are able to flexibly express their personal demands, to choose different policies or methods for performing negotiation. Both methods have been implemented and experimented in various situations. To improve the use of the program, we have tried to implement in JADE and Java which are able to provide much more comfortable user-interface and other possibilities, e.g., some types of predefined templates for communication exchanges (Request, Inform, Refuse).

Acknowledgement: The paper is partially supported by APVT and VEGA grant agencies under grants No APVT 51 011602 and VEGA 2/1101/21.

REFERENCES

[1] Gasser L.: Social Knowledge and Social Action. In IJCAI93, 751-757, 1993.

[2] Shehory O. and Kraus S.: Feasible formation of coalitions among autonomous agents in non-super-additive environments. Computational Intelligence, 15(3):218--251, 1999.

[3] Dang T.T.: PhD Thesis, 2003.

[4] Durfee, E. H.: Practically Coordinating. AI Magazine, 20(1): 99-116, 1999.

[5] Wooldridge, M., Bussmann, S. and Klosterberg, M.: Production Sequencing as Negotiation. In Proceedings of the PAAM--97, 709-726, 1996.

[6] Reeves, D. M., Wellman, M. P., Grosof, B. N., and Chan, H. Y.: Automated negotiation from declarative contract descriptions. In Computational Intelligence, 18, 482–500, 2002.

[7] Koning, J. L.: Designing and testing negotiation protocols for electronic commerce applications. Forthcoming publication. 2000.

[8] Guttman R. H., Moukas A. G., and Maes P.: Agent-mediated electronic commerce: a survey. Knowledge Engineering Review, 13(2):143-152, June 1998.

[9] Jennings N. R., Sycara K., and Wooldridge M.: A Roadmap of Agent Research and Development. Journal of Autonomous Agents and Multi-Agent Systems, 1(1):7--38, 1998.

[10] Paolucci M., Shehory O., and Sycara K.: Interleaving Planning and Execution in a Multiagent Team Planning Environment. Tech. report CMU-RI-TR-00-01, Robotics Institute, Carnegie Mellon University, 2000.
[11] Lomuscio A. R., Wooldridge M. and Jennings N. R.: A Classification Scheme for Negotiation in Electronic Commerce. A European Perspective, Lecture Notes in Computer Science, Vol. 1991, Springer Verlag, 19-33, 2000.
[12] Tran D. V., Hluchy L. and Nguyen T. G.: Parallel Program Model for Distributed Systems. In Proceeding of EURO PVM/MPI – 2000, Lecture Notes in Computer Science, Vol. 1908, 250 – 257, 2000.
[13] Tran D. V., Hluchy L. and Nguyen T. G.: Parallel Program Model and Environment. In Parallel Computing PARCO -99, 697-704, 1999.
[14] Kearns M., Mansour Y., and Singh S.: Fast planning in stochastic games. In Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, 309-316, 2000.
[15] Dang T.-Tung and Frankovic B.: Agent based scheduling in production systems. In International Journal of Production Research, 3669-3679, Vol. 40, No. 15, 2002.
[16] Sutton, R.S. and Barto, A.G.. Reinforcement Learning: An Introduction. MIT Press, 1998.

BIOGRAPHY

Baltazár Frankovič is a member of IFAC/IFIP TC (technical committee), scientific society ECCAI, SSKI, J.von Neumann Society Budapest. He is a member of the Central European Academy of Sciences and Arts. He acts as a member of the Editorial Board of some current journal, like Journal of Computer and Artificial Intelligence.
His current research interests in modeling and simulation of FMS, adaptive and learning control in DEDS, MAS. His publications contain above 5books, 124 scientific papers.

Dang Thanh-Tung was born in ThaiBinh, Vietnam, in 1973. He received the Ing. (Msc) degree from the Faculty of Electrical Engineering and Informatics, at Slovak Technical University, Bratislava (1997); and PhD degree from Institute of Informatics, SAS (2003). His research interests in MAS, planning and scheduling, reasoning and knowledge management.

Ivana Budinská graduated from Slovak Technical University, Faculty of Electrical Engineering in 1987, and received the PhD degree from Institute of Informatics, SAS, in 2003. Her research interests in DEDS, control theory and MAS.