Incrementally Refined Acquaintance Model (IRAM) for Request-based Virtual Organizations Formation

Michal Pěchouček, Jan Doubek, Jiří Vokřínek, Martin Rehák
Agent Technology Group, Gerstner Laboratory
Department of Cybernetics, Czech Technical University in Prague
E-mail: pechouc@labe.felk.cvut.cz

***Abstract*—This paper presents a specific contracting algorithm that contributes to the process of distributed planning and resource allocation in competitive, semi-trusted environments. The presented contraction algorithm is based on incrementally refined acquaintance models (IRAM) of the actor that provide the right set of approximate knowledge needed for appropriate task decomposition and delegation. This paper reports on empirical evaluation of the IRAM algorithm deployment in Request-based Virtual Organizations formation domain.**

I. Introduction

The presented algorithm has been designed within the framework of the research project[1] investigating the use of multi-agent techniques supporting coordinated action in Request-based Virtual Organizations for dynamic multinational clusters of ERP/CRM value chain actors. The Request-based Virtual Organization (RBVO) is a special kind of Virtual Organization [1]. A formation of a RBVO is based on a negotiation between independent actors willing to cooperate. Individual actors are motivated to join the Virtual Organization to increase their business opportunities and to be able to participate on larger scale contracts.

The individual value chain actors act as service providers mainly for consulting, software implementation, installation and customization, training and maintenance. The motivation is to find best suitable consortium of service providers to meet customer requirements such as cost, experience in industrial domain and appropriate ERP solution, geographical location or language. In this text, we understand all the constraints and criteria aggregated in single *price* value of the service.

The domain of RBVOs organizes multi-party interaction in the environments that are:

- **non-centralized** and with flat organizational structure [R1] – the existence of a central coordinating is minimal and the information about the skills of actors, resource availability, knowledge and goals is be distributed,
- **multi-party involvement** [R2] – the final project cannot be implemented in isolation by a single actor, RBVO formation can be initiated by several actors simultaneously,
- provides **partial knowledge sharing** [R3] – the actors in the environment are motivated to keep a substantial part of their private planning knowledge and resource availability information undisclosed.

[1]PANDA: www.panda-project.com

In this paper we present a contracting algorithm that contributes to the process of distributed planning and resource allocation in competitive, semi-trusted environments. The presented contracting algorithm is based on incrementally refined acquaintance models (IRAM) – the model that the actor is maintaining about potential collaborators [2].

II. Decomposition and Delegation in RBVOs

The formation of RBVO can be represented as distributed state-space search through all the potential RBVO formations in the environment with limited information sharing.

Let us have service *providers* (actors)

$$P = \{p_1 \dots p_\rho\} \tag{1}$$

and set of available *services*

$$U = \{s_1, \dots, s_\varphi\} \tag{2}$$

Every provider is able to deliver certain services

$$S_p = \{s_1, \dots, s_{k_p}\} \subseteq U \tag{3}$$

where p is a provider and k_p is the total number of different services that p provides. The *task* is then defined as

$$t = \{s_1, \dots, s_n\} \subseteq U \tag{4}$$

Every service s_j provided by provider i has its *partial price* that is computed by particular *pricing function* that depends on pc_p - the provider's internal constraints such as availability of resources and pricing/discount strategy. The price of the individual subtasks also depends on the size of the contract that is to be in place between the requester and provider. We denote the AS_p as a batch of tasks provided by the provider p. Pricing function is defined as

$$f(s_j, AS_p, pc_p), \tag{5}$$

where $s_j \in AS_p \subset S_p$. This function is private and unique for every provider. We assume this function to be non-increasing in reference to increasing $|AS_p|$.

Every AS_p, a batch of services provided by provider p has its *aggregated price* computed according to

$$F_p(AS_p) = \sum_{j=1}^{|AS_p|} f(s_j, AS_p, pc_p) \;\; \forall s_j; \;\; s_j \in AS_p \tag{6}$$

The non-increasing assumption then causes

$$s_x \in AS_{p,1}$$
$$s_y \in AS_{p,2}$$
$$s_y = s_x$$
$$|AS_{p,1}| \leq |AS_{p,2}|$$
$$\Rightarrow f(s_x, AS_{p,1}, pc_p) \geq f(s_y, AS_{p,2}, pc_p) \qquad (7)$$

For further specification purposes we also have to specify provider's p max coverage $AS_{p,i}^{max}$ set of services of particular task t_i

$$AS_{p,t_i}^{max} = t \bigcap S_p \qquad (8)$$

this means all services $s_x \in S_p$ that provider p is able to handle in current task t.

The set of all admissible RBVOs that actually solve the task t is then defined as

$$RBVO(t) = \{\{AS_{x,t}^y\}_{x=1\ldots\rho}\}_{y=1\ldots m} \qquad (9)$$

$$\text{so that } \forall y = 1\ldots m \;\; t = \bigcup_{j=1\ldots n} AS_{j,t}^y, \qquad (10)$$

representing m different ways how the community of n providers can participate in a RBVO solving a task t. We will denote the domain of admissible RVBOs as a *deal space* ($DS_t = RBVO(t)$). The size of this space is represented by following formula

$$|DS_t| = \left(\frac{\rho\langle k_p\rangle}{\varphi}\right)^{|t|} \qquad (11)$$

where $\langle k_p \rangle$ represents average amount of services provided by the provider p.

The goal is to find optimal RBVO that minimizes the operational cost, defined as follows:

$$RBVO(t)^{optimal} = \{AS_{x,t}^z\}_{x=1\ldots n} \subset RBVO(t)$$
$$\text{so that } z = \arg\min_{y=1\ldots m} \sum_{x=1}^{\rho} F_x(AS_{x,t}^y) \qquad (12)$$

In our model we have assumed made several assumptions:

- **Fixed price** – the price of a particular subset of services is fixed during all negotiations, this means you pay the same amount of money for services A, B, C in tasks t_1 and t_2.
- **Tasks are independent** – a provider is capable of delivering same services during all negotiation even if he was contracted for some services in the previous tasks.
- **Availability of a rovider is not changing during negotiation** – the provider decides what particular services (from set S_p) he proposes for a certain task, but this availability doesn't change during iterations (it can still provide different subset of his services for another task).

III. IRAM-based Decomposition and Delegation

We have designed a straightforward decomposition mechanism that finds the most optimal decomposition given the right objective function and a complete information about provider's resource availabilities. The decomposition algorithm is polynomial and easy to construct (see [3]). Its behavior, however, worsens strongly with lower quality of information about the provider's resource availabilities stored in the requestors' acquaintance models. The most efficient approach in fully cooperative communities would be if the requestor queries all the providers and constructs the entire acquaintance model for all services provided by all actors prior to computing the optimal a contract.

As this is not possible in the environment compliant with the requirements R1 and R3, the requestor needs to approximate such knowledge with only partially available information. We are proposing *incrementally refined acquaintance model* (IRAM) algorithm for handling partial knowledge sharing and private knowledge disclosure [2]. In order to evaluate the quality of IRAM we have developed a simple *reference algorithm*. While reference algorithm is based on distributed space-search where decision making is based on fixed, pre-computed model model; IRAM incrementally builds the acquaintance model during the negotiation process.

A. Acquaintance Model

The acquaintance model can have a number of forms [4], [5]. In this particular application the acquaintance model is understood as function that predicts actor responses to a particular *call-for-proposals* (`CFP`) type of message. We represent the *acquaintance model* as a mapping from a set of $\mathcal{P}(PS)$ possible subsets asked from the provider to a 1 dimensional real-value space representing cost $\mathcal{C}$.

$$\mathcal{F}_j^{am} : \mathcal{P}(PS) \rightarrow \mathcal{C} \qquad (13)$$

Let us discuss several properties of an acquaintance model. The *fixed point* is such a mapping among the actor, the service and a particular cost that is based on exact information acquired from the communication with the specific actor. In a fixed point as_x (denoted in lower case)

$$\mathcal{F}_j^{am}(as_x) = f_j(as_x, pc_j) \qquad (14)$$

where $f_j(as_x, pc_j)$ represents the exact costs of the actor P_j providing the as_x set of services. Other mapping is provided by the approximative capability of the acquaintance model.

Provided that the fixed points of the acquaintance model are collected in a set $\Delta(\mathcal{F}_j^{am})$, we define the *size of the acquaintance model* $\delta(\mathcal{F}_j^{am})$ the amount of the fixed points in the acquaintance model as follows:

$$\delta(\mathcal{F}_j^{am}) = |\Delta(\mathcal{F}_j^{am})|. \qquad (15)$$

The maximal size of the acquaintance model is described by the following formula.

$$\delta_{max}(\mathcal{F}_j^{am}) = \sum_{i=1}^{p} \sum_{j=1}^{k_i} \binom{k_i}{j}. \tag{16}$$

Various approximation functions have been used in the acquaintance models. Linear approximation have been used in the scenarios with highly granular service provision distributions in the field of logistics planning [3]. In our model we have selected the pairwise constant approximation. The unknown price of task t in subset with size $|AS|$, equals to the closest bigger known fixed point in $|as|$

$$\mathcal{F}_j(t, |AS|, pc_j) = \mathcal{F}_j(t, |as|, pc_j) \text{ if } |AS| \lessdot |as|, \tag{17}$$

provided that the symbol $\lessdot$ represent the smallest bigger value.

The *error of the acquaintance model* - $\epsilon(\mathcal{F}_j^{am})$ - represents how well does the acquaintance model capture real capability of the providers. Error of the acquaintance model is a dual quantity to the *quality of the acquaintance model*. There can be a number of ways how the error can be related to the quality. We only require that with a monotonic increase of quality the error decreases and vice versa.

We represent the error of the acquaintance model as a sum of the differences between the real costs and the information on costs provided by the acquaintance model.

$$\epsilon(\mathcal{F}_j^{am}) = \sum_{p,i} |\mathcal{F}_p^{am}(AS_{p,i}) - f_p(AS_{p,i}, pc_j)| \tag{18}$$

As said before, the reason why we use the acquaintance models for contracting is that we are motivated by minimizing the unwanted knowledge disclosure during interaction (requirement R3). Each interaction represents disclosure of private information. By `CFP` the actors disclose their inability to perform a task as well as their intention to do so. By a response to `CFP` the agents disclose information about availability of particular resources. It is evident that with rising $\delta(\mathcal{F}_j^{am})$, the acquaintance model is more exact and thus provides better information (i.e. lower $\epsilon(\mathcal{F}_j^{am})$). Better acquaintance model managed to reduce communication (and thus private knowledge disclosure) during the negotiation between the actors. However, bigger $\delta(\mathcal{F}_j^{am})$ (and thus smaller $\epsilon(\mathcal{F}_j^{am})$) required substantial interaction during the acquaintance model construction phase where lots of unwanted information may have been disclosed.

The IRAM algorithm is balancing the size and the quality of the acquaintance models. In order to evaluate performance of the IRAM algorithm we have developed a reference algorithm that is working with a fixed acquaintance model, constructed prior negotiation. Both algorithms are based on distributed state-space search using negotiation between actors. As a negotiation protocol, we use the *competitive contract-net protocol* [6], but any protocol that enables iterative contract negotiation can be used.

B. Reference algorithm

For the comparison, we have designed an algorithm that simulates the naive approach to finding the optimal RBVO. It was constructed to be simple and to use only incoming data from providers without any further computing and price modeling. The algorithm is based on two sets of RBVOs assembled from different data. The first one is DS which is evaluated by prices requested in the first step from max coverage (Equation. 8), and stays the same through the negotiation. The second one is FP (which means assembled from fixed points $\mathcal{F}_j^{am}$, to cover needed task), so in the first step there are very few fixed points, but the number of fixed points grow with negotiation. The definition of deal space secures that $FP \subset DS$. In negotiation steps we compare best RBVOs from FP and from DS/FP.
Deal space set - due to constant state in negotiation and max coverage prices, it is evaluated by lower price bound. The prices $p(s_i)$ for particular subtasks s_i are set $p(s_i) = F_p(AS_p)/|AS_p|$
Fixed point set - is reassembled in every negotiation step again and again, because of new fixed points. It stores only real RBVOs that can be ordered.
DS/FP set - due to FP has to be also reassembled in every step. The best RBVO from this set is used for termination condition, it represents the possible best RBVO. The size of this set is decreasing with the negotiation steps.

1) Initialization: In first step of negotiation all providers are requested for $AS_{p,t}^{max}$, and the DS is constructed. The FP is also constructed from $AS_{p,t}^{max}$ but it can be $\emptyset$, thus $DS/FP = DS$.

2) Iteration phases:
– request the best RBVO from DS/FP
– construct FP
– construct DS/FP
– evaluate termination condition

3) Termination condition: The optimum is found and the algorithm will terminate if best RBVO from FP is better than the best from DS/FP

4) Partial pricing problem: But according to this case we cannot secure the convergence to optimum with aggregated price, thanks to the price difference between the particular services. The calculated price carries the averaging error caused by unknown *partial price*(see Equation 5). But when the partial price is known the algorithm will find the optimum in every run. So every provider has to respond with set of prices for particular services in asked subset AS

$$\{f(s_1, AS, pc_p), \ldots, f(s_{|AS|}, AS, pc_p)\} \tag{19}$$

This assumption partially relaxes requirement R3, but it is needed for ensuring algorithm convergence.

C. IRAM Algorithm

The run of this algorithm for one particular task t is started with the initiation phase. All providers are contacted for every

Fig. 1. Reference algorithm steps

Set $FP = \{\}$
Construct deal space DS_t.
Create AS_p^{max} according to task t for all providers from P.
Send CFP(AS_p^{max}**)** to all providers from P.
Add received set of AS_p to FP.
If $f(RBVO^{best} \in FP) \leq f(RBVO^{best} \in DS/FP)$ **then** terminate algorithm.
Send CFP($AS_{p,i}$**)**, where $\bigcup_{i=1\ldots n} AS_{p,i} = RBVO^{best} \in DS/FP$ to providers $1 \ldots n$.
Goto 5.

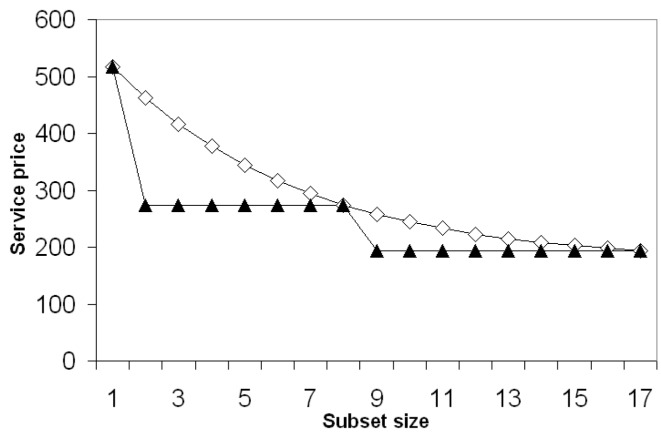

Fig. 2. IRAM model for one service according to as, fixed points in 1,8,17

single service and for maximal subset of services from task t. The IRAM model am is constructed using the closest bigger known fix-point approximation (see Equation 17). The model is represented by sets of prices for specific service and pricing function settings (see Equation 6). The price is set blank when is not known, and thus is calculated from other fixed points. The algorithm constructs the deal space and evaluates it with the prices from $\mathcal{F}_j^{am}$. The cheapest RBVO is selected and the providers are requested for appropriate services. Offered prices are then integrated into the IRAM model. The deal space is then reevaluated and the cheapest RBVO is selected. If the new RBVO is equal to the RBVO from previous step, this we understand as optimum. The phases of Iram can be seen below.

1) Initialization: It is necessary to know at least two fixed points of acquaintance model for specific service from each provider, for proper functionality of IRAM algorithm. So if the algorithm have not these from previous contracts, it obtains them in the first iteration. Due to this fact the amount of communication is considerably higher regarding to following iterations. Preferably we choose the single service data and maximum provider coverage data (Equation 8). Single service provides us data needed for proportional price reconstruction necessary due to aggregated price. And the max coverage data gives us the lowest possible prices from provider needed for approximation.

2) Iteration phases: The iteration phases represent processes that follow each other in further negotiation stage.

- Contacting the best known RBVO given by acquaintance model
- Updating IRAM model by the received responses
- Reevaluating the acquaintance model
- Sorting the deal space by total RBVO price
- Termination condition evaluation

3) Termination condition: The algorithm is iterating (contacting and updating model) till the best evaluated RBVO stays best after contacting it and updating and reevaluating the acquaintance model.

The steps of IRAM algorithm can be seen in Figure 3.

Fig. 3. IRAM algorithm steps

Set $i = 1$.
Construct deal space DS_t.
Send CFP(s_i**)** for all $s_i \in t$ to all providers from P.
Update am according to received responses.
Send CFP(AS_p^{max}**)** to all providers from P.
Update am according to received set of responses AS_p.
Select $RBVO_i^{best}$ from DS_t evaluated by am.
If $i = 1$ **then goto** 10.
If $RBVO_i^{best} = RBVO_{i-1}^{best}$ **then** terminate algorithm.
Send CFP($AS_{p,j}$**)**, where $\bigcup_{j=1\ldots n} AS_{p,j} = RBVO_i^{best}$ to providers $1 \ldots n$.
Set $i = i + 1$.
Goto 6.

D. Properties of IRAM

The presented IRAM algorithm is sound and complete. The proof of completeness of the algorithm is made through conversion of whole idea to A^* algorithm [7], where the nodes of searched space are individual RBVOs from DS, and edges represent inclusion (or exclusion) of one provider to RBVO. This representation corresponds to *Coalition Structure graph* [8].

The heuristics of A^* is then based on acquaintance model approximation, all of the nodes are priced particularly by real prices (fixed-points) and by computed prices given by acquaintance model. That is represented by

$$F(RBVO(t)) = g(RBVO(t)) + h(RBVO(t)),$$

$$\text{where } g(RBVO(t)) = \sum_{AS_i \in \Delta(\mathcal{F}_j^{am})} f(AS_i),$$

$$\text{and } h(RBVO(t)) = \sum_{AS_i \in DS/\Delta(\mathcal{F}_j^{am})} \mathcal{F}^{am}(AS_i). \tag{20}$$

The $g(RBVO(t))$ represents price of the of subsets from $RBVO(t)$ that is known from previous negotiations (the fixed-points) and $h(RBVO(t))$ is the unknown price of the subsets from $RBVO(t)$ estimated by acquaintance model. According to Equation 17 and 7 the $h(RBVO(t))$ is always equal or lower then the real price of this subset, so $h(RBVO(t)) \leq h^*(RBVO(t))$ and the Equation 20 is admissible heuristics of A^* algorithm. Since the IRAM is based on exploration of the best candidates evaluated by Equation 20 the algorithm provides the features of A^* algorithm [7].

IV. Experiments

The key contribution of the presented paper is in empirical evaluation of the presented algorithm. We will be analyzing the behavior of IRAM in relation to the reference algorithm presented above.

For presenting the contribution of IRAM we construct a market model that contains a set of four providers P, one simple *requester* that requests providers for set of 17 tasks $S = t_1 \dots t_{17}$ using IRAM and reference algorithm for comparison.

In our experiments we randomly generated 4 providers, where everyone of them was capable of delivering 14 services from total 18 services, this setting was chosen due to computation requirements. The max size of *acquaintance model* is $\delta_{max}(\mathcal{F}_j^{am}) = 65532$ possible proposals i.e. *fixed points*, and average *deal space* size in one task is $\langle |DS_t| \rangle = 8777$.

The individual pricing functions were randomly generated as follows. We generate uniform distribution set of prices $UP = (bp_1 \dots bp_s)$ in defined range $(200, 600)$ for base price bp of every service from U. Then we create one random value in range $d_j \in (0.6, 1)$ for particular provider p_j that represent the discount in price according to the number of total services asked $|AS|$. From discount is then computed margin value m_j

$$m_j = 1 + 0.05 * ((1 - d_j)/0.8) * ((1 - d_j)/0.8 + 1)/2. \quad (21)$$

The price $f_{i,j}$ of single service s_i is then derived from base price bp_i and total service asked $|AS|$ as follows.

$$f_j(s_i) = m_j bp_i d^{|AS|-1} + 0.5 m_j bp_i \quad (22)$$

Then the pricing function corresponds to Eq. 6 Every provider then responds only with this price when is asked for some services (except from the reference algorithm setting).

We have carried out two distinct sets of experiments: (*i*) studying a short-term and (*ii*) longer-term performance of IRAM. While the former set of experiments illustrates how IRAM can efficiently run negotiations among the partners with no a priori information and previous collaboration experience, while the latter shows how IRAM accumulate experience from previous negotiation rounds and uses it for efficient deal decomposition.

A. Single-Deal Negotiation

The advantage of IRAM in a new deal negotiation with new partners is in a fact that the acquaintance model is constructed only around fixed points, that are subject of the requestor's interest. Based on collaboration rejects from the providers, the requestor is updating its acquaintance model in the most relevant regions. The presented IRAM algorithm provides the ability of fast convergence in comparison with reference algorithm in single task experiment. Figure 4 shows that IRAM algorithm is able to find the optimal RBVO with the price of 1595 in 5 iterations and the reference algorithm in 76 iterations. The amount of communication sent (the number of proposals) is proportional to the number of fixed-points in the acquaintance model and thus to its size - $\delta(\mathcal{F}_j^{am})$. Figure 5 shows that the number of communicated proposals is considerably higher in the first iteration (explained in Section III-C), but when the optimum was found IRAM was operating with the acquaintance model of the size $\delta(\mathcal{F}_j^{am}) = 40$, while the reference algorithm worked with the size of the model $\delta(\mathcal{F}_j^{am}) = 95$.

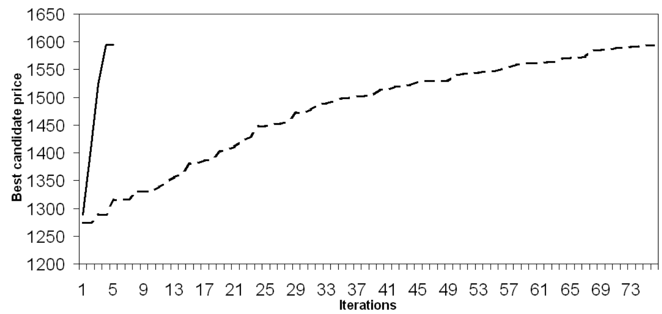

Fig. 4. Potential RBVO price according to iterations in first task (IRAM – solid line, reference algorithm – dashed line, horizontal axis – number of negotiation iterations, vertical axis – price).

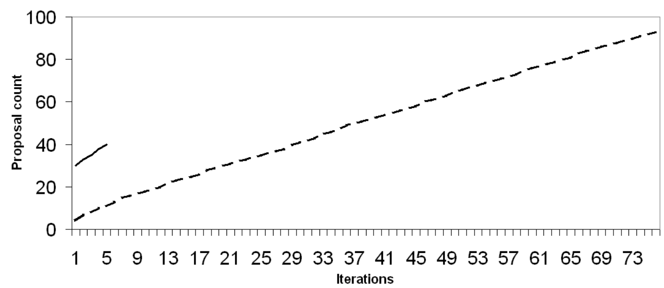

Fig. 5. Proposal count according to iterations in first task (IRAM – solid line, reference algorithm – dashed line, horizontal axis – number of negotiation iterations, vertical axis – size of the acquaintance model represented by the number of proposals).

B. Sequence of Deals Negotiation

When IRAM is used for a sequence of multiple (interrelated) deals, the size of the acquaintance model is increasing but the communication traffic needed for negotiating a single deal is substantially decreasing. Figure 6 shows that the proposal count for each subsequent task is decreasing while this is not the case for the reference algorithm that does not process the log of previous negotiation. Also the initiating communication overhead that occurs during first few tasks is made substantially smaller with higher number of communicated deals (Figure 7). After several runs, the acquaintance model covers the most important parts of the all potential deal spaces and thus the algorithm is able to find the optimal solution in minimal number of iterations for any new task (see Figure 6).

The sum of proposals sent to providers during all negotiated tasks reflects the incremental refinement by negative derivation of the function shown in Figure 8. The acquaintance model size growth is slowed down and the size of the model converge to the size adequate to defined scenario (based on number of providers, services, pricing function differences, providers overlapping and tasks generation). In our experiment setting the size of an acquaintance model after 17 tasks is $\delta(\mathcal{F}_j^{am}) = 168$, which is 0.26% of $\delta_{max}(\mathcal{F}_j^{am})$.

V. Conclusion

The presented IRAM algorithm allows formation of RBVO with respect to defined requirements, mainly non-centralized approach and minimization of private knowledge disclosure. The consecutive construction of an acquaintance model allows to refine it in the scenario with set of deals. It leads to almost immediate finding of optimal solution with the minimal communication and reasonable size of the acquaintance model. High dynamics of the environment causes devaluation of acquaintance model and it can lead to incorrect solution. In such dynamic environment, the IRAM algorithm has to start building the model from scratch for every new task. In one deal scenario, the presented IRAM algorithm still provides quick convergence to the optimum with low communication and thus low private knowledge disclosure.

Acknowledgment

The work is funded by the European Commission's FP6 programme[2] within the projects Collaborative Process Automation Support using Service Level Agreements and Intelligent Dynamic Agents in SME clusters (PANDA), IST-No.027169. The research is also partly funded by the Ministry of Education, Youth and Sports of the Czech Republic grant No. MSM 6840770038 and No. 1M0567.

References

[1] R. Roberts, A. Svirskas, and B. Matthews, "Request based virtual organisations (RBVO): An implementation scenario," in *Collaborative Networks and their Breeding Environments*, ser. IFIP, vol. 186. Springer, 2005, pp. 17–25.

[2] M. Pěchouček, V. Mařík, and J. Bárta, "Role of acquaintance models in agent's private and semi-knowledge disclosure," *Knowledge-Based Systems*, no. 19, pp. 259–271, 2006.

[3] M. Pěchouček, O. Lerch, and J. Bíba, "Iterative query-based approach to efficient task decomposition and resource allocation," in *Cooperative Information Agents X*, ser. Lecture Notes in Computer Science, M. Klusch, M. Rovatsos, and T. Payne, Eds., vol. 4149, CIA Workshop. Springer, 2006, pp. 258–272.

[4] W. Cao, C.-G. Bian, and G. Hartvigsen, "Achieving efficient cooperation in a multi-agent system: The twin-base modeling," in *Cooperative Information Agents*, ser. LNAI, P. Kandzia and M. Klusch, Eds., no. 1202. Springer-Verlag, Heidelberg, 1997, pp. 210–221.

[2] Any opinions expressed in this paper are those of the author(s)/organisation and do not necessarily reflect the views of the European Community. The Community is not liable for any use that may be made of the information contained herein.

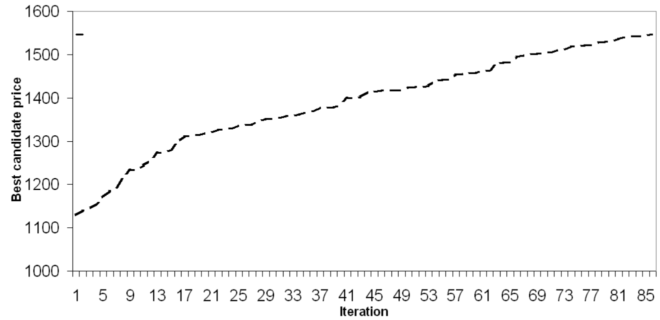

Fig. 6. Potential RBVO price according to iterations. Convergence to optimum in 10th task (IRAM – solid line, reference algorithm – dashed line, horizontal axis – number of negotiation iterations, vertical axis – price) – shows that IRAM have constructed the model appropriately and is thus capable of finding the deal in minimum iterations.

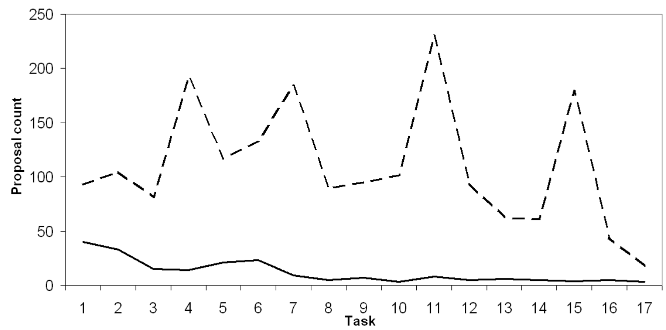

Fig. 7. Proposal count in iterations (IRAM – solid line, reference algorithm – dashed line, horizontal axis – tasks, vertical axis – number of proposals) – shows that total proposal communication by IRAM decreases with time or rather by task solved.

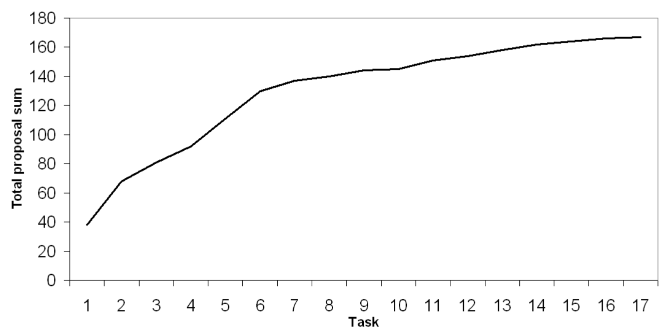

Fig. 8. Proposal sum through tasks from start of the negotiations for IRAM (horizontal axis – tasks, vertical axis – size of acquaintance model $\delta(\mathcal{F}_j^{am})$)

[5] V. Mařík, M. Pěchouček, and O. Štěpánková, "Social knowledge in multi-agent systems," in *Multi-Agent Systems and Applications*, ser. LNAI, M. Luck, V. Mařík, and O. Štěpánková, Eds. Springer-Verlag, Heidelberg, 2001.

[6] J. Vokřínek, J. Hodík, J. Bíba, J. Vybíhal, and M. Pěchouček, "Competitive contract net protocol," in *LNCS 4362 – SOFSEM 2007: Theory and Practice of Computer Science*. Berlin: Springer-Verlag, 2007, pp. 656–668.

[7] R. Dechter and J. Pearl, "Generalized best-first search strategies and the optimality of A*," *J. ACM*, vol. 32, no. 3, pp. 505–536, 1985.

[8] T. Sandholm, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. Cambridge, MA.: MIT Press, 1999, ch. Distributed Rational Decision Making, pp. 201–258.