

Multi-agent Model for Real Time Resource Allocation¹

V. Gorodetski, L. Lebedev

St. Petersburg Institute for Informatics and Automation
39, 14th Linija, St.Petersburg, 199178, Russia
E-mail:gor@mail.ias.spb.su

Article received on January 22, 1999; accepted on April 29, 1999

Abstract

A large number of important applications are reducible to combinatorial models. Almost all of them are at least exponential complexity and cannot be solved in a traditional way. In the paper, we consider an agent-based approach to solve a class of complex combinatorial problems in the area of planning and scheduling of bounded resource allocation under real time and temporal constraints. The model of the problem is formulated in terms of contract allocation over a set of contractors and specified as an auction-based competition of intelligent agents-contractors under agent-manager supervision. The paper contributions are repeatable auction-based scheme of random search of admissible decisions; knowledge-based specification of real-time and temporal constraints that is used to order of contract allocation from step to step of auction procedure; dynamic programming approach for forming strategy of bargaining by agent-contractor.

Keywords:

Multi-agent systems, combinatorial optimization, real-time scheduling, random search, auction-based coordination.

1 Introduction

Although research in the area of combinatorial problem solving have been pursued several decades, the problem itself is still of hot current interest. There exist many applications, for example, manufacturing, transportation management, shipping, workflow management, etc. that are reducible to a combinatorial problem. Many of newly arisen applications, for example, creation of virtual enterprises (Fisher *et al.*, 96), business process reengineering (Jennings *et al.*, 96), etc. are also specified formally as combinatorial problems. Almost all of them are at least of exponential complexity and cannot be solved efficiently in a traditional way, including traditional knowledge-based approaches developing during last two decades. It seems, that currently, multi-agent system (MAS) technology is the only one that is able to cope efficiently with the above applications. It became feasible due to some specific features of MAS technology, in particular, due to a distributed way of tasks solving and due to search of an approximate solution instead of optimal one.

A specific of distributed way of decision making in MAS is that each agent must make decision having a deficit of information about environment and other agents. To master an information deficit and to coordinate individual behaviors, agents use negotiation. Rules of negotiation usually are formalized as a negotiation protocol. For competitive agents the most popular negotiation protocols is one based on the metaphor of the auction (Sandholm, 96). Very often it leads to a good result in many respects.

An auction-based negotiation protocol prototype is the contract net protocol developed for centralized way of resource allocation (Smith, 80), (Smith *et al.*, 81). In fact, auction-based

¹The paper is supported by Russian Foundation of Basic Research (Grant 99-01-00733).

negotiation was the first idea used for coordination of agent behavior. Later it was investigated by a number of authors (see, for example, (Sanholm, 93), (Sandholm, 96), (Fisher *et al.*, 96), (Jennings *et al.*, 96) where it was shown that auction-based negotiation protocol is an efficient way for self-interested agent behavior coordination.

In the paper we consider a use of auction-based negotiations among agents of MAS that have to solve a complex combinatorial task of works ("contracts") allocation. Our task statement differs compared to those considered in other papers in many respects. The most close task statement was considered in (Sandholm, 93) and later investigated in (Sandholm, 96). In the framework of the Sandholm's model, each agent-contractor is self-interested, permitted to perform consistently a number of works, and mechanism of auction management is based on minimization by contractors their own marginal cost for contract execution.

A specific feature of the task statement in the paper is as follows. We aim at solving a complex task of planning and scheduling of contracts allocation which complexity is conditioned by real time, temporal, and other constraints imposed on contract execution. The second specific feature of the task is that, in general case, it is not inherently distributed. One more feature is a multi-criterion assessment of solution quality.

Auction-based model of agent behavior coordination within contract allocation task can be considered in two different senses in respect to application. On the one hand, agent contractors can be representatives of really competing entities (companies) which are self-interested and aimed at each own benefit only. In this case, auction is a mathematical model of a real world competition of companies. On the other hand, if we must solve a centralized planning and scheduling task of a large company the auction-based model may be used as a metaphor that makes it possible to implement artificial competition as a coordination mechanism of subtasks. For such case agent-contractors naturally are not self-interested and aim at solving a complex task jointly. Nevertheless, even in this case the metaphor of auction is useful. It should be noted that in between of the above cases all possible degrees of centralization-decentralization of planning tasks are lain.

In the paper we focus on the latter interpretation of the auction, i.e. it is considered as a metaphor to solve a centralized planning and scheduling task in a distributed way. In this context each session of auction to allocate all contracts makes it possible to obtain an admissible solution of a global task. But in general case coordination mechanism using metaphor of auction does not guarantee solution optimality or, at least, a good quality of it. It means that an auction-based procedure of subtasks of a task coordination can be considered as a way to generate admissible solutions. To obtain at least a "good

solution", it was proposed to repeat auction many times using randomization and on-line learning to improve decisions from step to step. Thus, we consider planning and scheduling procedures as multiple repetition of randomized auction aiming at allocation of the entire set of contracts. This procedure must possess on-line adaptation of the auction scenario for obtaining a better solution on the basis of multi-criteria evaluation its quality assessed by so called "Quality of Service" (QoS) vector.

The described point of view on the role of auction entails an increasing importance of intelligent components of MAS, in particular, of intelligent supervising of auction scenario by agent-manager (Gorodetski *et al.*, 94).

In the paper we propose MAS architecture for a class of combinatorial problem solving and focus on agent's intelligent components for decision making. A peculiarity of the considered class of applications is that to meet temporal and real-time constraints, it is necessary not only "to reason" but, in addition, to do a lot of calculations to satisfy imposed constraints and to optimize on-line behaviour of each agent-contractor (Gorodetski, 97).

The rest of the paper is organized as follows. In *Section 2* we discuss the subject of the paper research on conceptual level and outline the developed MAS architecture. The problem is specified as a task of dynamic scheduling of a "portfolio of contracts" allocation. In *Section 3* we discuss peculiarities of auction and auction management that is conditioned by the accepted way of auction interpretation. In *Section 4* the architecture of agent manager is considered and the main idea of an algorithm of on-line auction management is described. *Section 5* is devoted to agent-contractor architecture and to on-line planning algorithm that it uses. In conclusion we present shortly the main results of the paper and outline future works.

2 Conceptual Problem Statement

Let us consider a class of applications that deals with planning, scheduling and bounded resource allocation problems under real time and temporal constraints. It is supposed that every contract (task, service providing, etc.) may consist of a number of sub-contracts (sub-tasks) and be executed in diverse scenarios which consume different resources.

In conceptual terms, the problem is specified as an allocation of a set of contracts ("portfolio of contracts") to be executed over eventual contractors capable to execute specific contracts. As a sample of application we may consider planning and scheduling of an activity of a great company (plant, transportation company, service providing company, etc.) that deals with executing of real-timed interdependent contracts. A set of contracts may be executed by a number of sub-divisions (departments, job shops, etc.) of the company itself

and in such case we have to solve a complex combinatorial task of centralized planning and scheduling. In other case, the contracts to be executed may be allocated over a set of more small independent (may be, not self-interested) companies. In particular, the latter takes place for virtual enterprise creation, transportation tasks, etc.

As an example let us consider the task of planning and scheduling of ship service in a large Sea Port. We suppose that each ship or cargo holder in advance made a contract with a company which role is a moderator ("agent") between ship (cargo) and Sea Port. The ship service has to be provided within a given real-time interval.

The ship service operations have to be executed at fixed berths or by ship service companies (contractors). Each of them has to meet technological constraints (for example, specialization of contractors in respect to type of cargo, kind of work for contract, etc.), admissible duration of cargo saving in a Port depository, parameters of ship (for example, required depth along a berth, length of ship, etc.) and real-time constraints imposed by contract.

For each contract, a vector of quality of service (QoS) is mapped. In particular, QoS vector contains a payment for agent (holder of "portfolio of contracts") from ship holder, a penalty for contract execution delay or violation of other contract conditions. It should be noted that QoS vector for manager differs from one for contractors.

It is clear that a lot of applications in the planning and scheduling area can be reduced to the problem statement given above. A common feature of all such tasks is high computational complexity and necessity to use decomposition-based approach.

Let us consider conceptually parties of auction, their roles and interests. The subject of auction is a set of contracts each mapped QoS vector defined by manager. An auction participants are as follows:

- *agent-manager* that represents a "portfolio of contracts" holder; it is interested in to allocate ("to sell") all contracts as cheaper as possible (to pay for each contract execution as little as possible) ensuring that each contract execution meets constraints imposed by QoS vector of ship or cargo holder, and
- *agent-contractors*; each of them represents a company subdivision or independent company. It is capable to execute chosen contracts providing QoS vector imposed by agent-manager and interested in obtaining ("buying") contracts to be paid as many as possible and to minimize its own cost for contract execution. It is permitted for agent-contractors to exchange resources making provisional coalition.

We suppose that while managing by contract allocation, agent-manager may

- use pure market-based competition of agent-contractors,

- choose subset of contractors for bargaining for a contract up to choice of exactly one of them and
- use randomization for winner choice.

Each agent-contractor develops its strategy of behavior on-line to maximize its own total benefit under imposed constraints but sometimes it has to refuse from bargaining in favor of other contractor considering such behavior as payment of its debt. It means that agent-contractors are not pure self-interested.

On the fig.1 a general outline of MAS architecture is presented. Its peculiarity is that in addition to bargaining for contracts, there exists one more loop for agent-contractors negotiation that aims at forming provisional coalitions to exchange resources.

When participating in a "competition", each agent-contractor forms its own part of a solution and a total solution of the task is formed by joining solutions of all agents-contractors. Agent-manager enjoys a number of mechanisms that are considered below in the next section to manage by auction and to coordinate decisions made by agent-contractors to achieve its own goal that is "global" one for MAS in the large.

3 Peculiarities of Auction and Auction Management

It was written in introduction that, in the paper framework an auction is considered as a metaphor to represent a decentralized search of solution of a complex planning and scheduling task of contract allocation. This point of view on auction results in considerable distinction of auction strategy and auction management compared to those proposed in previous research and focused on idea of competition of self-interested agents.

These distinctions may be divided into two groups. The *first of them* is conditioned by a specific role of agent-manager in the course of auction since to manage auction it uses combination of market-based strategy of agent-contractors behavior management and randomized search of globally "good solution". When omitting details, the peculiarities of the first group of distinctions are as follows.

1. A multi-criterion evaluation of bids that are presented as QoS vector proposed by agent contractors in the course of auction. The QoS parameters depend on application. If we deal with scheduling of a Sea Port it may consist of such components as "<"Date of contract execution end", "Cost of service", "Probability of execution of the contract provided by stated QoS vector ">".
2. A repeatable character of auction that plays a role of a search procedure.
3. Random search. A randomization itself comprises the following components.

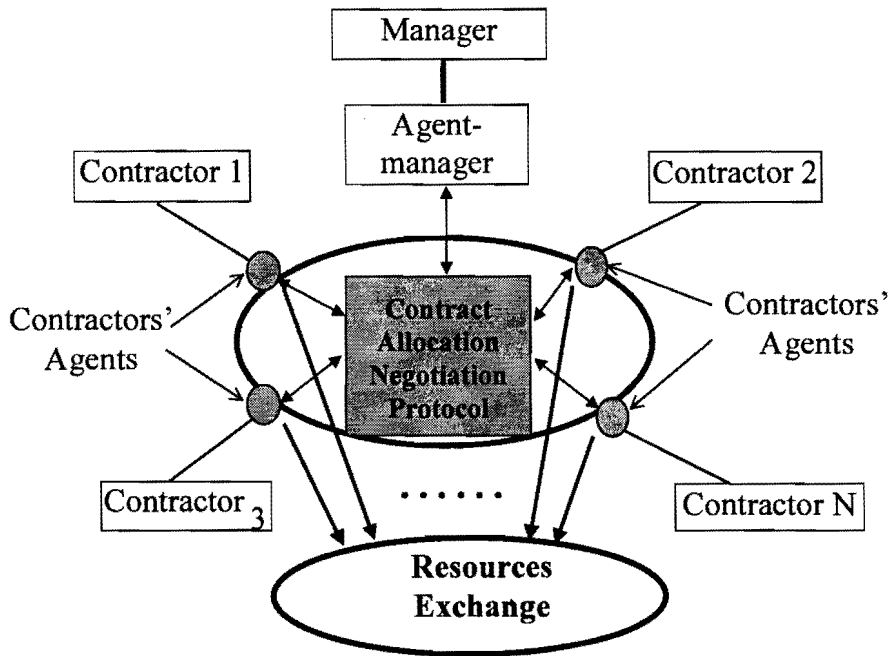


Figure 1: Agent-based architecture for planning, scheduling and resource allocation.

- Multiple repetition of auction session for total “portfolio of contract”. Each session may be based on a scenario that is being chosen randomly from given variants.
 - Randomization of a sequence of contracts presented for bidding. Each sequence had to meet imposed partial order.
 - Randomization of winner’s choice within the set of non-dominated bids (Pareto optimal bids) according to proposed QoS vectors of contractors.
4. Learning by experience based on multi-criteria rating of agent-contractors.
 5. When making decision about the best solution among generated admissible ones, agent-manager uses its own global vector of QoS that is other one than it proposes to each contractor in the course of auction. Agent-manager tries to optimize the total gain and has to take into account possible penalties to be paid to ship holders.

The second group of peculiarities of auction management is conditioned by agent-contractor decision making algorithm and they are as follows.

1. Agent-contractors are capable to optimize their strategy and current decisions on every step of auction when taking into account information about the rest of “portfolio of contracts”, about winners on previous steps of auction if it is informed about and prognosis of competitors in the progress of auction.
2. Based on randomization procedure, agent-contractor may vary scenario of contract execution utilizing resources it possess.

4 Agent-manager

Agent-manager is considered as a holder of a “contract portfolio” and is responsible for management of contract allocation to provide a required global QoS vector. Its architecture that is currently in progress of implementation is presented on the fig.2 and in general is organized like *InteRRap* architecture (Muller *et al.*, 94). It consists of three layers in accordance with the layers of abstractions of auction management, i.e.

- layer of data management (*layer I*),
- layer of behaviour management (*layer II*),
- layer of strategy of behaviour choice (*layer III*).

The upper layer (*layer III*) is responsible for making a strategic decision about main parameters of auction scenario. It is supposed that on this layer agent-manager is able

- to vary a kind of auction scenario (negotiation protocol) based on a number of available variants that may be chosen via randomization procedure or/and is based on learning during previous and/or current repeatable sessions of the “portfolio of contract” allocation;
- to vary information that is available to agent-contractors;
- to determine rules of transforming (increasing) of payment for current contract and
- to choose randomized rules of auction winner determination.

Details of procedures of layer III are currently under development.

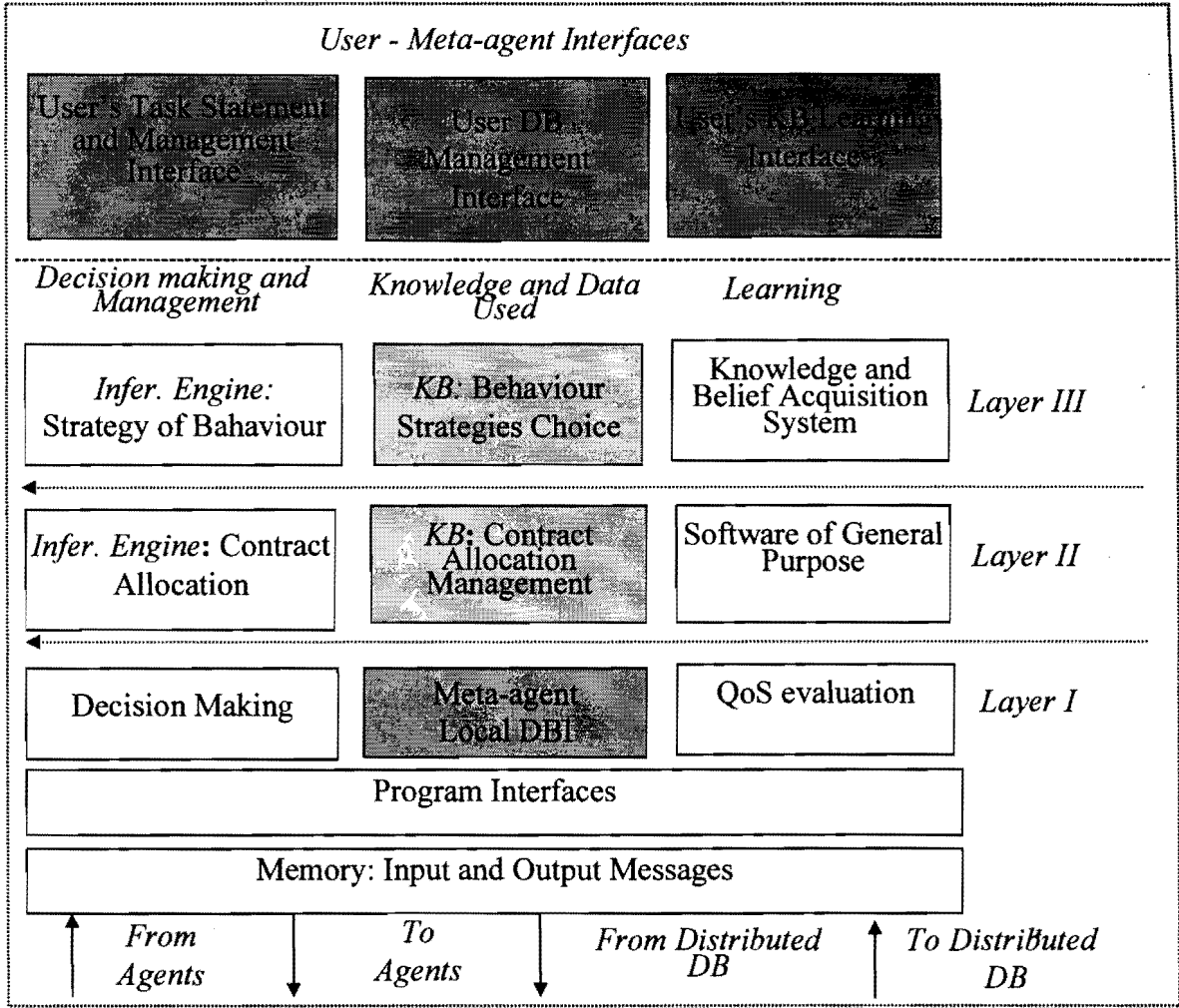


Figure 2: Meta-agent's architecture.

The next layer (*layer II*) of agent-manager architecture is responsible for real-time management of auction session in accordance with the chosen scenario. Procedures related to this layer are developed in more details compared to the previous one. On the basis of computational experiments it was obtained that the most difficult task of this layer is satisfaction of real-time and temporal constraints imposed on the set of contracts. Within developed approach, this task is solved by knowledge-based technology. Let us explain its main idea.

Each contract $C(i)$ is mapped by average or minimal duration of its execution and the real-time interval within which contract has to be executed. These constraints are specified by three parameters like it is shown in fig.3, i.e. $t_s(i)$, $t_f(i)$ – is the admissible time of begin and end of contract execution and $\tau(i)$ – is duration of a contract

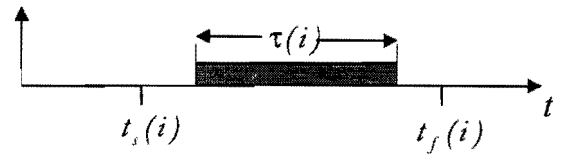


Figure 3: Real time parameters of contract execution

execution, i – is the number of a contract.

While having such information regarding all contracts, we may formalize this information in terms of two partial order relations which are as follows.

1. *Precedence relation*. We say that $C(i) \geq_{pr} C(j)$ if the following conditions are met.

- i. $t_s(i) + \tau(i) \leq t_f(j) - \tau(j)$,
- ii. $t_s(j) + \tau(j) > t_f(i) - \tau(i)$

This conditions express the fact that if contract $C(i)$ and $C(j)$ are planned to be executed by the same agent-contractor then, to meet the real-time constraint for contract $C(i)$, its execution have to be started first. This fact is illustrated on the fig.4.

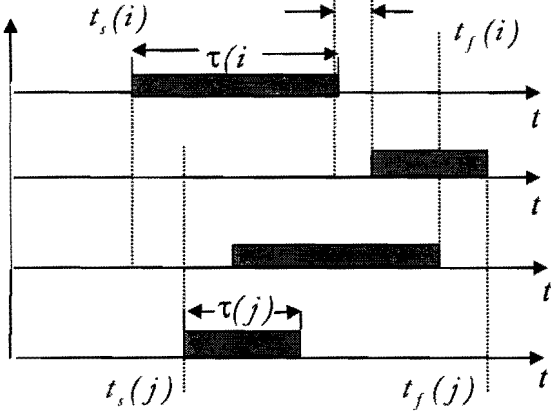


Figure 4: Illustration of the Precedence relation $C(i) \geq_{Pr} C(j)$.

2. Strict Precedence Relation. We say that $C(i) \geq_{sPr} C(j)$ if $t_s(j) \geq t_f(i)$.

This relation is a partial order one. It is illustrated on the fig .5.

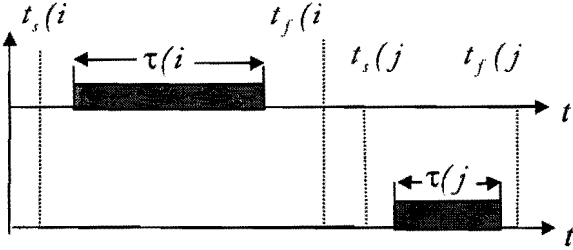


Figure 5: Illustration of the precedence $C(i) \geq_{sPr} C(j)$.

The implementation semantic of the former order relation is as follows: if contract $C(i)$ is planned for execution by a contractor then contracts $C(j)$ may become incomparable for execution by the same contractor. It is conditioned by the time of begin of execution of contract $C(i)$. For example, this situation occurs if the time $t_s(i)$ of begin of execution of contract $C(i)$ is scheduled later than $t_f(j) - \tau(j) - \tau(i)$. As for the last strict precedence relation, it persists constant because of admissible real-time intervals mapped to each of contract have empty intersection. It can be proved that the joining of the both introduced above relations is partial order relation.

It can be shown, that in addition to both above order relations there exist two more variants of relation among time intervals of contracts. The first of these relations reflects the situation when time intervals of a pair of contracts have nonempty intersection but they may be executed in any order. The second one reflects the situation when a pair of contracts cannot be executed by the same agent-contractor in any case.

We will ignore both last relations because they don't impose any order relation on contract execution.

One more kind of constraints that have to be taken into account to form a general order of contract execution is so-called temporal constraints that are conditioned by technological steps of contract execution. Actually, it is possible that a subset of contracts forms a more general contract (technology) that have to be executed in accordance with a partial ordering. For example, a ship in a Sea Port first have to be unloaded and only after this operation it may be loaded and contracts for these operations is arranged with different companies (berths).

It can be shown that while joining temporal constraints, precedence and strict precedence relations we result in the precedence relation $PR(C_i, C_j)$. Calculation of the latter from step to step of auction is a task that is solved by agent-manager and realized in the *layer II* of its architecture. $PR(C_i, C_j)$ relationship is included as a component of knowledge base. It is used for ordering of contract allocation along the auction.

Let us note that relation $PR(C_i, C_j)$ is not static and it is being transformed from step to step of contract allocation during auction procedure. Thus, the *layer II* of agent-manager architecture is responsible for forming $PR(C_i, C_j)$ relation, its transforming in progress of auction and for computations which include randomization of decision making to solve what contract have to be allocated next.

Other components of agent-manager architecture on the *layer II* are more traditional and we omit details about them.

The *layer I* of the agent-manager architecture is responsible for processing of the input and output messages during the auction sessions.

5 Agent-contractor

Agent-contractor's architecture is outlined on the fig.6. It is based on the same ideas as agent-manager architecture but differs in tasks assigned to the respective layers.

On the *layer III* of architecture the task of choice of a preferable subset of contracts to execute is computed. Based on (1) knowledge about its own available resources, (2) constraints imposed by manager on QoS vectors of each contract execution, and (3) own expense for each contract

execution, agent-contractor forms a strategy for the forthcoming step of auction using algorithm of dynamic programming.

Agent-contractor is informed about all “portfolio of contracts” and QoS vectors that are mapped to each of them. Having this information, agent-contractor is able to predict the subset of contracts that will be allocated on the current step of auction. This makes it possible for it to use dynamic programming procedure to generate a number of preferable sequences of contracts that are most beneficial for it and can be executed in time.. This is the most important task of agent-contractor that is solved on the upper layer.

On the *layer II*, a step-by-step price policy of contractor is calculated. The policy is a way to achieve its own goal in course of competition. The policy of choice depends on a number of factors for each agent. For example, each agent-contractor is mapped by a degree of risk (“risky”, “normal” or “careful”) and these factors influence on its price policy.

As well, this layer is responsible for final decision making.

One more task of agent-contractor is to exchange by resources with other agents if it haven’t needed recourses in necessary quantity. Currently this task is under research. To solve it, we intend to use an algorithm that is based on “mutual debts” idea and commitment “first to pass resources to debtor if it requests than to use resources for own needs”.

6 Conclusion

In the paper we have aimed at developing a multi-agent approach to solve large scale *combinatorial problems in the area of planning and scheduling* of a set of interdependent contracts (“contracts”) under real-time and temporal constraints by a number of contractors each having a bounded resources and a specialization. We focus on the case when the task under solution is not inherently distributed but it is a large scale task decomposed to be solved in a distributed way

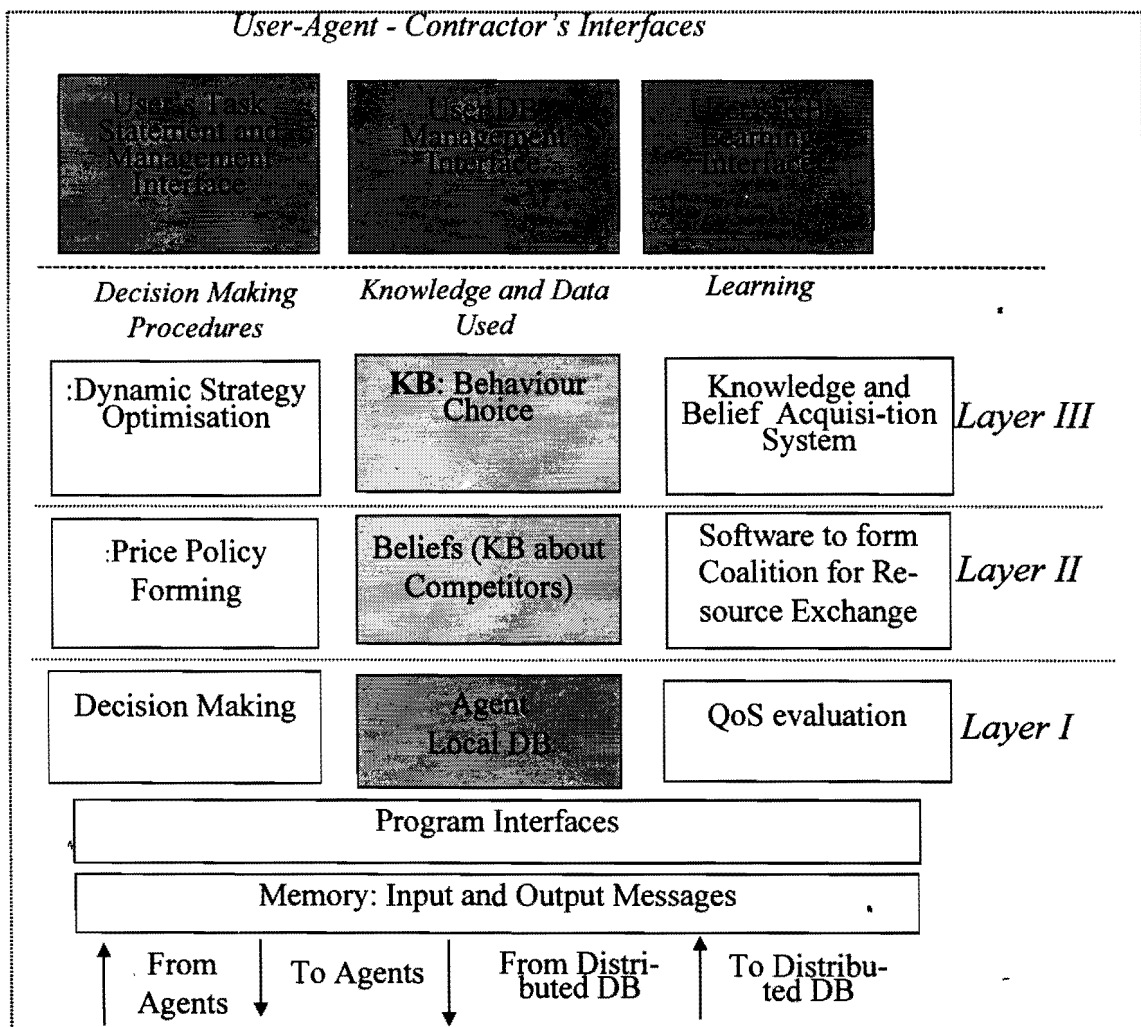


Figure 6: Agent-contractor's Architecture.

and managed by an meta-agent. In this case, a more sophisticated ways of decision making have to be used not only by agent-manager but also by agent-contractors.

The paper contributions are as follows.

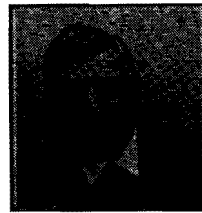
- Repeatable auction-based scheme of random search of admissible decisions.
- Knowledge-based specification of real-time and temporal constraints that is used by agent-manager to determine an order of contract allocation from step to step of auction.
- Dynamic programming approach to form strategy of bargaining of an agent-contractor.

Future research will relate to software implementation of a multi-agent system based on ideas described in the paper and theoretical and experimental research in the field of agent-manager and agent-contractor on-line and off-line learning in the framework of real time planning and scheduling tasks.

References

- Fisher V, Muller V, Heimig I., A-W.Scheer. Intelligent Agents in Virtual Enterprises. In *Proc. of the First Intern. Conference "The Practical Application of Intelligent Agents and Multi-Agent Technology"*, London, 1996, pp.205-224.
- Gorodetski V. Basic Ideas of Agent Behaviour Coordination and Auction - based Scheduling Model.. In *Proc. of the 1 Int. Workshop "Distributed Artificial Intelligence and Multi-agent Systems"*, St. Petersburg, 1997, pp.282-291.
- Gorodetski V, Lebedev A. Planning and Scheduling: Real Time Constraint Satisfaction. "*Problems of Informatics*", No. 3-4, 1994. pp. 49-55 (in Russian).
- Jennings N.R., Paratin P., Jonson M. Using Intelligent Agents to Manage Business Processes. In *Proc. of the First International Conference and Exhibition "The Practical Application of Intelligent Agents and Multi-Agent Technology"*, London, UK, 1996, pp.345- 376.
- Muller J.P., Pishel M., and Thiel M. Modelling Reactive Behaviour in Vertically Layered Agent Architectures. In: *Intelligent Agents. ECAI-94 Workshop on Agent Theories, Architecture and Languages*. Amsterdam, The Netherlands, (Eds. M.J.Wooldridge and N.R.Jennings). Proceedings. Springer Verlag: 1994, 261-276.
- Sandholm T.W. ¹An Implementation of Contract Net Protocol Based on Marginal Cost Calculations, In *Proceedings of 11th AAAI*, 1993, pp.256-262.
- Sandholm T.W. Negotiation among Self-interested Computationally Limited Agents. Dissertation for Ph.D., 1996. <http://www.cs.wustl.edu/~sandholm/dissertation.ps>
- Smith R. G. *The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver*, IEEE Transactions on Computers C-29(12), 1980, pp. 1104-1113.
- Smith R. G. and Davis R. *Framework for Cooperation in Distributed Problem Solving*, IEEE Transactions on Systems, Man and Cybernetics, SMC-11, 1981, pp.61-70.

Vladimir Ivanovitch Gorodetski, Prof. of Computer Science, Ph.D,



Doc. of Nat. Science, Chief Scientists of the St.Petersburg Institute for Informatics and Automation of the Russian Academy of Science.

He graduated the Military Air Force Engineer Academy in St.Petersburg (1960, mechanical department) and Math. and Mechanical

Department of the St.Petersburg State University (master degree in 1970), obtained his Ph.D. degree in 1967 and Doc. of Nat.Sci. degree in 1973 in the area of Space Vehicle Optimal Control. Main publications (more than 180 including 9 books and textbooks) relate to the areas of Optimal Control System Theory, Applied Statistics, Planning, Pattern Recognition and Artificial Intelligence (Data Mining, Multi-agent Systems). Current scientific interests relate to Data Mining (Statistical and Logical), Machine Learning, Bayes Inference, Consistency Maintenance for Knowledge Bases under Uncertainty, Applied Multi-Agent Systems design. Applications of interest are Real Time Planning and Scheduling, Diagnostics, Agent-based Information Security Systems. During last six years he served as Program Committee Chair and a member of a number of international conferences and workshops in the areas of Control, Artificial Intelligence, Multi-agent Systems, Open Distributed Processing, etc.

