Czech Technical University in Prague
Faculty of Electrical Engineering

DOCTORAL THESIS STATEMENT

Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics

Jiří Vokřínek

Distributed Problem Solving by Means of Agent Negotiation

Ph.D. Programme: Electrical Engineering and Information Technology
Branch of study: Artificial Intelligence and Biocybernetics

Doctoral thesis statement for obtaining the academic title of "Doctor", abbreviated to "Ph.D."

Prague, February 2011

The doctoral thesis was produced in combined manner Ph.D. study at the Department of Cybernetics of the Faculty of Electrical Engineering of the CTU in Prague

Candidate: Jiří Vokřínek
Department of Cybernetics
Faculty of Electrical Engineering of the CTU in Prague
Technická 2, 166 27 Prague 6

Supervisor: Prof. Ing. Vladimír Mařík, DrSc.
Department of Cybernetics
Faculty of Electrical Engineering of the CTU in Prague
Technická 2, 166 27 Prague 6

Opponents:

The doctoral thesis statement was distributed on: ....

The defence of the doctoral thesis will be held on ....at .... before the Board for the Defence of the Doctoral Thesis in the branch of study (to be specified) in the meeting room No. ....of the Faculty of Electrical Engineering of the CTU in Prague.

Those interested may get acquainted with the doctoral thesis concerned at the Dean Office of the Faculty of Electrical Engineering of the CTU in Prague, at the Department for Science and Research, Technická 2, Praha 6.

Prof. Ing. Vladimír Mařík, DrSc.
Chairman of the Board for the Defence of the Doctoral Thesis
in the branch of study Artificial and Biocybernetics
Faculty of Electrical Engineering of the CTU in Prague
Technická 2, 166 27 Prague 6.

1 Introduction and Related Work

Problem solving and planning in decentralized environments is a key technical challenge in numerous industrial applications, ranging from manufacturing, logistics, virtual enterprises to multi-robotics systems. The integration of the task refinement (decomposition), allocation and local planning enables to explore the planning and allocation possibilities taking into account the availability of resources. There may be several ways to decompose a task. For each such a decomposition various allocations exist and each agent can make different plans for each part of a decomposed task. The question of which decomposition is the best for a given task cannot be answered easily without allocation and local plans' evaluation. Since the number of combinations grows rapidly with the number of agents as well as the decomposition and allocation possibilities, it leads us to the non-trivial problem of multi-agent problem solving.

This work presents an abstract architecture of a multi-agent solver and a respective algorithm providing decomposition, task allocation and task delegation. The architecture is based on social welfare maximization using agent negotiation over task allocation, delegation and reallocation. The interaction mechanism ensures global optimization behavior while the individual agents apply local planning heuristics and strategies. Various features of the abstract architecture, such as computational complexity or admissibility of the underlying optimization heuristics, are analyzed.

Additionally, an approach to plan representation based on social commitments suitable for flexible replanning and plan revision purposes in dynamic non-deterministic environments is introduced. The key idea is in the representation of a distributed hierarchical plan by social commitments, as a theoretically studied formalism for representing mutual relations among intentions of collaborating agents. Reasoning about decommitment alternatives during the planning process contributes to flexibility and robustness of the resulting plan. We formally introduce and discuss three specific decommitment rules: (i) relaxation, (ii) delegation and (iii) full decommitment. The process of coordination and interaction between agents depends strongly on the ability of individual actors to perform intelligent decommitment upon specific changes in the environment. We argue that an appropriate selection, setting and preference ordering of decommitment rules contributes to robustness of the overall solution. The abstract problem solving architecture is extended for commitment allocation and the decommitment model is discussed.

The approach was verified and evaluated in simulated environments and real case studies. The experimental validation confirms the ability of the multi-agent problem solver to provide high-quality solutions, performance, stability and robustness of the system in complex scenarios. Four instances of the abstract architecture implementations are given to demonstrate the applicability of the presented approaches in a wide variety of real world problem domains – vehicle routing problems, strategic mission planning, multi-robot frontiers exploration and production planning.

1.1 Distributed Problem Solving and Planning

The problem of controlling entities in a heterogeneous distributed environment is crucial for many domains [10]. Classical centralized methods depend on one central planning system. Such a system gathers all required input data before the planning process takes place. Then the plan (a set of plans respectively) is generated using these data. This approach faces various problems. One problem is the need for private local knowledge of the actors. The other problem is the need for real-time replanning based on environments and conditions changing dynamically over time. On the other hand, in distributed methods of planning, each entity plans its own plan. Cooperation and heading towards common goals is done by various methods of negotiation.

Distributed planning and problem solving usually refer to environments where planning, solving, or coordination activity is distributed across multiple actors, processes, or sites. Distributed planning has been viewed as either (*i*) planning for activities and resources allocated among distributed agents, (*ii*) distributed (parallel) computation aimed at plan construction or (*iii*) plan merging activity. During the years of artificial intelligence (AI) research the fields of distributed problem solving (DPS), distributed planning (DP), and multi-agent planning and problem solving (MAPS) have grown significantly and became significant inherent pars of the scientific fields of distributed artificial intelligence. All three mentioned domains focus on collective effort of distributed actors to solve a joint problem. The emphasis is put on the optimization (or constrain satisfaction), planning or coordination. The distributed problem solving and planning was introduced by a statement that the DPS is a subfield of distributed AI emphasizing the collective effort of the agents to solve the common problems [10]. Compared to DPS and DP, where a lot of standard problem formulations, frameworks and methods exists (i.e. [4, 6, 7, 8, 9, 11]), MAPS is centered around the coordination and interaction of autonomous agents. The typical MAPS problem is defined as a problem with autonomy of the actors (the agents are at least partially autonomous), locality of the views (no agent has a full global view of the system, or the system is too complex for an agent to make practical use of such knowledge), and high degree of decentralization [20]. The MAPS community focuses a lot on (i) single agent knowledge representation and reasoning [3, 5, 21], (ii) multi-agent coordination and negotiation [14, 15, 18]. Also there are many implementations of multi-agent systems utilizing the concepts of autonomy, rationality, social behavior, and other features of agent-based systems [12, 16]. This work focuses mainly on multi-agent problem solving based on task allocation and local resource planning in cooperative environments.

The multi-agent system design is composed of two components: (i) local agent algorithms design that respects autonomy, individual constraints, goals and resources of agents and (ii) inter-agent interaction schemes that provide social aspects of the whole system and support (emergent) macro behavior of the multi-agent system. Agent interactions in the presented architecture are motivated by cooperative solving of a given problem. Similarly to social aspects referred to as comparative advantage in economy [17], where a group of individuals cooperates on the delivery of a service or goods at a lower opportunity cost than other groups, the agent community tries to find a solution maximizing social welfare [1]. As shown in this work, the welfare maximization can be transformed to minimization of the cost of assignments of tasks (sub-problems) to individual agents. This cost is computed locally using planning algorithms of individual agents. The overall solution cost is minimized using interactions between agents – task allocation, delegation and reallocation. The allocation of a task to an agent is represented by a social commitment that an agent undertakes [13]. This representation provides a powerful tool for task execution stability and performance in dynamic and/or uncertain environments. The commitment-based approach enables to change the solver-centric point of view to a more task-centric point of view and enhance the solver operations for heterogenous commitments in dynamic environments.

This work mainly focuses on the interaction component of the multi-agent system in the context of problem solving using task allocation and defines constraints to the local agent planning strategies. It does not directly address the local agent's point of view, but presents a commitment-based plan representation enhancing the execution stability and robustness. The presented approaches are demonstrated on a series of implementations of multi-agent systems for various domains and their evaluation and validation.

2 Aims of the Doctoral Thesis

This thesis aims to provide a unified architecture suitable for a wide variety of multi-agent problem solving domains. Such an architecture has to provide a computationally feasible high-quality solution, execution robustness and good performance and stability in heterogenous dynamic environments with a high degree of uncertainty. The goals of the thesis can be formalized as follows:

1. Define a domain independent abstract architecture of the multi-agent problem solver for distributed problem solving. Formalize the multi-agent problem and develop a distributed problem solving algorithm on top of it using the abstract architecture.

2. Analyze features of the abstract architecture and the algorithm and argue the admissibility of this approach for problem solving.

3. Enhance the abstract architecture by mechanisms for executional stability and robustness of the solution, flexible replanning and plan revisions in dynamic non-deterministic environments.

4. Evaluate the architecture and algorithms using a comparative benchmark problem and provide arguments supporting the applicability of the approach for distributed problem solving.

5. Evaluate the enhanced architecture using stress scenarios to validate the robustness and execution stability of the approach.

6. Validate the developed architecture, algorithms and mechanisms using realistic application scenarios to support the applicability of the multi-agent problem solving approach in real-world problems.

3 Working Methods

Multi-agent planning approaches are used for solving a wide variety of planning problems. As analyzed by Brafman and Domshlak [2] the multi-agent planning techniques can be beneficial for such problems where the domain sizes of individual agents are considerably smaller (e.g. in logarithmic relation to each other) than the overall size of the problem (even if the planning complexity of an individual agent is exponential) and the number of dependencies between agents is low. The distributed planning and problem solving has been analyzed by Durfee [10]. One of the related strategies discussed is a *task sharing* approach. The principle is based on passing of tasks from a busy agent to a vacant agent(s). The process can be summarized in four basic steps:

1. *Task decomposition* – the tasks of agents are decomposed into subtasks. Sharable subtasks are selected.
2. *Task allocation* – the selected tasks are assigned to the vacant agents or agents which ask for them.
3. *Task accomplishment* – each agent tries to accomplish its (sub)tasks. The tasks which need further decomposition are recursively processed and passed to other agents.
4. *Result synthesis* – the results of the tasks are returned to the allocating agent since it is aware of how to use it in the context of the higher level tasks.

From the perspective of distributed problem solving, task allocation and the result synthesis are the most crucial parts. On the other hand, from the planning perspective, the other two phases are more important. The allocation problem is usually solved by contracting and negotiation techniques which imply problems related to the resource allocation domain, e.g. cross-booking, over-booking, backtracking, and others. In the allocation phase, a hierarchy of agents is established, which may not be fixed in heterogeneous multi-agent systems.

The decomposition and delegation principle is widely used in agent-based approaches for problem solving and planning and shows great applicability to realistic problems. Taking into account Brafman and Domshlak's analysis, the Durfee's task sharing approach efficiency is tightly bound to the solver's ability to reduce the problem sizes for individual agents and keeping the dependencies between agents low.

In the domains where the optimization/planning problem can be decomposed into independent task the multi-agent approach shows its benefits. Such a task can be allocated and executed by different agents with low or no influence on each other.

3.1 Multi-Agent Solver

Based on the principles of problem decomposition and delegation described before, we can define the abstract multi-agent solver architecture as a composition of three types of agents (see Figure 1):

- *Task Agent* for pre-processing of the problem. This agent should use a domain specific heuristic, generic ordering strategy or randomized ordering method.
- *Allocation Agent* for problem decomposition into tasks and delegation of the tasks to Resource Agents. This agent maintains task allocation and result synthesis. This agent's strategies and algorithms are domain-independent.
- *Resource Agent* for individual case-specific resource planning. In case of further decomposition, the task is handed over to another Task Agent.

The multi-agent system built upon this architecture is composed of one Task Agent, one Allocation Agent and a set of Resource Agents. The fact there are multiple Resource Agents corresponds to distributed nature of the multi-agent problem. A complex hierarchical system can thus be captured by applying above mentioned architecture in recursive manner. Such representation allows for natural decomposition of the problem providing easy means for parallelization and addressing the complex problem in general (more abstract solvers are instantiated with potentially overlapping agents – e.g. several Task Agents or Allocation Agents handling various problems in parallel or single agent undertakes a role of more than one abstract agent type). In large systems, concurrent interactions may arise that need to be handled. The agents' interactions are guided by the interaction protocol defined within the scope of this work, which is based on concepts (very) similar to Smith's contract net protocol (CNP) [19].

The multi-agent solver presented in the thesis uses the principles of problem decomposition and delegation to autonomous agents that solve parts of the problem individually. The overall solution is then obtained by

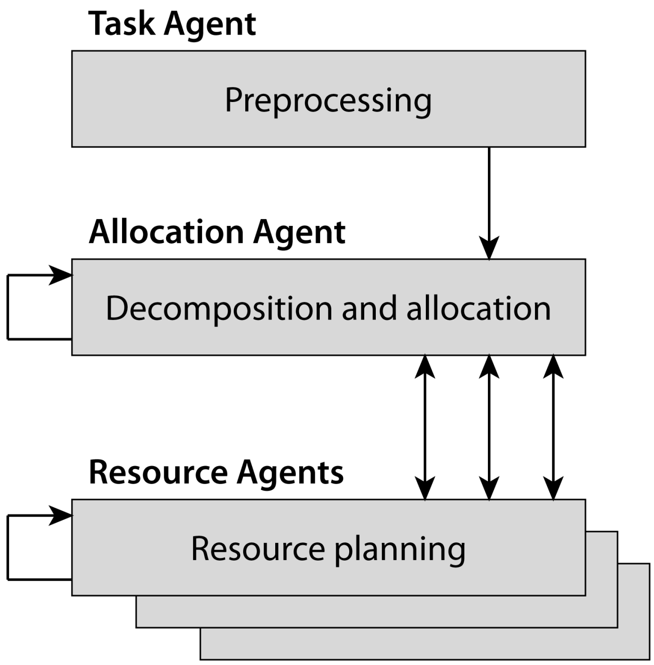


Figure 1: Abstract architecture of agent-based solver/planner.

merging the individual agents' results. The optimization based on CNP interactions in cooperative environment is usually described as utilitarian social welfare maximization [1]. So the abstract algorithm objective function can be defined as maximization of social welfare, which is a sum of individual agents utilities and can be transferred to minimization of solution cost, where cost of the assignment of particular task to an agent is computed by the respective Resource Agent that undertakes the task. The task allocation stage of the solver searches for the best suitable mapping of the tasks to the Resource Agents that minimizes defined objective function.

The abstract algorithm representing multi-agent solver contains three phases:

1. The first phase is task pre-processing provided by the Task Agent. The ordering heuristic represents case-specific sorting of the tasks to increase the solver's efficiency in the particular domain. In some cases the ordering has no influence, but in others it may provide significant improvement especially in domains with stronger task dependencies.

2. The second phase is iteration over all tasks and allocation performed by the Allocation Agent minimizing the insertion cost computed by Resource Agents. As part of this iteration, the dynamic improvement based on cooperation of Allocation Agent and all Resource Agents takes place – the improvement strategy is applied to every Resource Agent after allocation of each task (see below for the description of improvement strategies).

3. The third phase of the algorithm is the final improvement of the solution. After allocation of all tasks the improvement strategy is executed by all Resource Agents.

The improvement strategies used by the abstract algorithm are:

- *Delegate worst* – each Resource Agent identifies its worst task and tries to delegate it to another agent if the defined improvement condition holds.

- *Delegate all* – each Resource Agent tries to delegate all its tasks to another agent if the improvement condition is satisfied.

- *Reallocate all* – each Resource Agent successively removes all its tasks from the plan and allocates them again. The result of the allocation can be the same as before task removing, or a change of the position of the task in the current agent plan, or delegation to another agent.

To ensure proper function of improvement strategies we define admissible delegation and improvement terminating condition based on insertion/removal costs estimations and overall solution cost changes. Additionally, planning strategies of individual Resource Agents has to comply with defined admissability conditions.

The general computational complexity of the multi-agent solver is introduced in [2]. Using transformation of the multi-agent planning problem to the distributed constraint satisfaction problem, the worst-case time complexity of the multi-agent planning is upper-bounded by

$$f(\mathcal{I}) \times exp(comm) + exp(int),$$

where $f(\cdot)$ is the factor inducted by requesting each agent to plan while committing to a certain sequence of actions, $\mathcal{I}$ is the complexity of an individual agent's planning, $exp(comm)$ represents a factor exponential in min-max number of per-agent commitments, and an additive factor $exp(int)$ represents the interactions of agents. The consequences of this lead to interesting features of the multi-agent solver, such as that there is (i) no direct exponential dependence on the number of agents, (ii) no direct exponential dependence on the size of the planning problem or size of the joint plan, and (iii) no direct exponential dependence on the length of individual agent plans [2].

In our case the feature (ii) does not have a strong impact if the decomposition algorithm of the Allocation Agent is exponential because the size of its problem is the same as the size of the overall problem, but for the Resource Agents (and other subordinate agents in the case of a complex hierarchical structure) this feature holds. On the other hand the exponential factors are usually reduced by the polynomial heuristics – decomposition, allocation, optimization, and resource planning strategies implemented in real applications. The ordering strategy of the Task Agent does not have a strong influence on the worst-case complexity because of its additive nature and low complexity. The multi-agent solver benefits in the domains where the problem can be easily decomposed to independent tasks, and/or where the polynomial heuristics for the resource planning exist.

For the abstract algorithm we show that the improvement strategies complexity has no relation to the number of agents and the *worst case time complexity* of the abstract algorithm is upper bounded by

$$n \times log(n) + n^2 \times fr'(n) + m \times n^2 \times fr(n),$$

where n is number of Resource Agents, m is number of tasks, $fr(n')$ is the factor representing the complexity of the implemented agents' resource planning strategy for the task insertion and $fr'(n')$ is the factor representing the complexity of the implemented agents' resource planning strategy for the task removal. The complexity analysis shows the polynomial impact of the decomposition and delegation principles used by the multi-agent abstract solver. The impact of the two planning strategy factors is the following:

- the complexity of the operations of Resource Agent are multiplied by n^2,
- the influence of the number of Resource Agents is linear.

The complexity analysis shows us an important feature of agent-base solver. When using polynomial heuristics for task insertion and removal, the implemented multi-agent solver provides polynomial worst-case complexity. Together with linear computational scaling with the number of agents makes the presented abstract architecture suitable for many application areas. In real application areas, ordering heuristics can be found that result in allocation with no need of using improvement strategies (e.g. production planning system). In other cases, the planning of Resource Agents is implemented with low complexity (e.g. linear) and the improvement strategy has greater importance. The thesis presents several applications developed using the described abstract multi-agent solver.

3.2 Commitment Based Problem Solving

The thesis also introduce plan representation using social commitments to increase the execution stability of the solution provided by the solver and to strengthen the autonomy of agents. The improvement methods used by the solver are guided by the decommitment rules individually for each task in this representation. The decommitment rules are defined by Task Agent for each new task introduced to the system. Additional decommitment rules may be created by individual agents maintaining the commitments.

This approach enables to change the solver-centric point of view to the more task-centric point of view and enhance the solver operations with heterogenous commitments in dynamic environments. The formalism and algorithms for commitment-based problem solving are introduced, system architecture is defined, and execution features of this approach is analyzed.

In previous section we have put focus on the task negotiation based task allocation for problem solving. Agents undertake parts of the entire problem and create local plans for it. The plans have to be executed later and the multi-agent solver needs to address the dynamic uncertainty of the plan execution. The allocated tasks can be represented as a set of social commitments of participating agents. This commitment representation can be generalized to the distributed plan representation and provides a powerful tool for execution control and plan repairing.

The multi-agent problem we are dealing with can be informally understood as the task of solving a classical HTN (hierarchical task network) planning problem, defined by an initial partially ordered (causally connected) series of goals, by a set of admissible operators (defined by their preconditions and effects) and methods suggesting a decomposition of a goal into a lower-level planning problem. The plan can be sought for by an individual actor or in collaboration of multiple actors (sharing knowledge and resources). The product of planning is a set of partially ordered terminal actions, allocated to individual actors who agreed to implement the actions under certain circumstances. These circumstances are expressed by specific commitments including the following pieces of information:

- *Commitment condition* that may be (*i*) a specific situation in the environment (such as completion of some precondition) or (*ii*) a time interval in which the action is to be implemented no matter what the status of the environment is or (*iii*) a combination of both.
- *Decommitment conditions* specifying under which condition the actor is allowed to recommit from the commitment once the task is finished (e.g. notification) or once the task cannot be completed (e.g. a failure)

For long, multi-agent research community has been providing interesting results in the formal work in the field of agents' social commitment, as specific knowledge structures detailing agents individual and mutual commitments. We require the agents that perform intelligent planning and replanning by means of social commitments to be able to perform at least basic reasoning about the decommitment rules attached to a particular commitments. This is needed at the time of replanning, when an agent needs to decide which decommitment rule (i.e. a new commitment) to adopt, provided that conditions for more than one of them are satisfied. Similarly, when negotiating about who will accept which commitment, the agents shall be able to analyze not only properties of the goal and costs associated with the goal completion process but also the various decommitment rules when considering likelihood of the particular failure to happen. Ideally, the agent shall be able to estimate costs of each decommitment rule.

We have recognized three main types of decommitment usually used in commitments:

- *Full decommitment* decommits the original commitment if and only if the commitment goal is unrealizable (dropping the commitment).
- *Delegation* decommits the original commitment if and only if the commitment goal is unrealizable and the new commitment on the other agent's side is formed (delegates the commitment to another agent).
- *Relaxation* decommits the original commitment if and only if the commitment goal is unrealizable, the negotiated relaxation conditions hold and the relaxed commitment is formed (relaxation of the time frame or cost of the commitment).

The decommitment condition for each decommitment strategy is defined to enable flexibility of the commitment under various circumstances. During the planning process, the preference relation over the decommitments is defined as a part of the decommitment rule set.

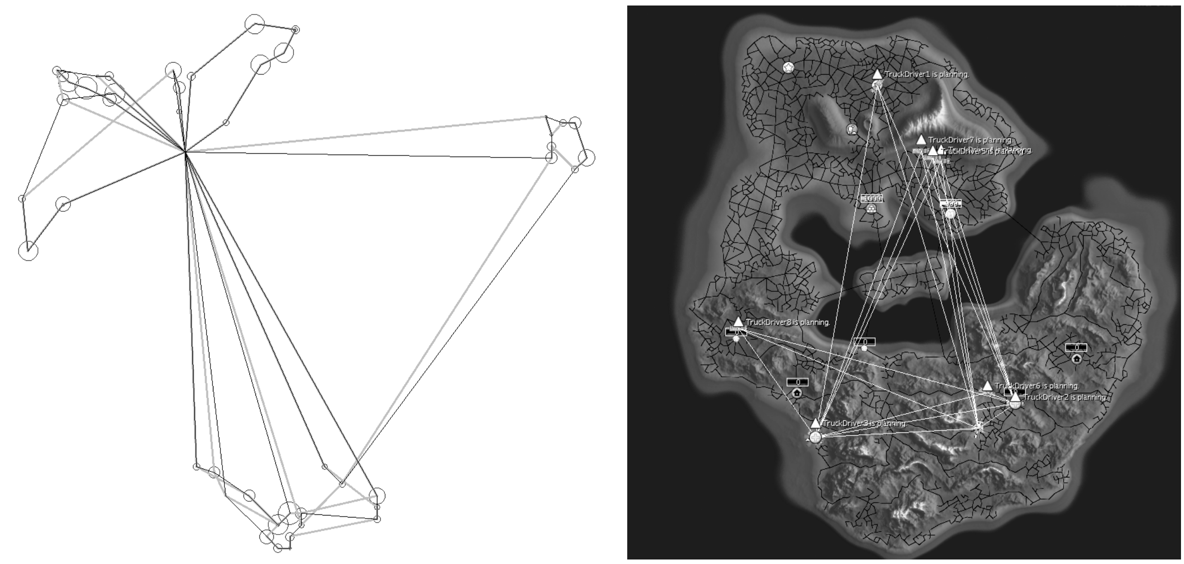
The multi-agent solver algorithms has been adapted to work with commitment based plan representation and task allocation process has been changed to commitment allocation. The committing phase is more or less the same as allocation phase of the algorithm described in precious section. After that the commitments are fixed and plan is executed. The dynamic improvement phase respects individual commitments. The particular decommitment rules are defined by the nature of the tasks. The complexity analysis and Resource Agent strategy admissability definitions holds. Using a set of properly defined decommitment rules the commitment-based solver is able to simulate the improvement strategies of the abstract solver, but the decommitment execution is fully controlled by Resource Agents.

4 Results

The applicability of the presented concepts and methods has been also validated and evaluated in a wide variety of real problem scenarios – vehicle routing problems, strategic mission planning, multi-robot frontiers exploration, and production planning. All the presented multi-agent systems share the same abstract architecture, algorithm and improvement strategies, some of them use commitment based plan representation. For each application domain a particular resource optimization heuristic has been designed and implemented by the Resource Agents. All the presented systems' implementations have low computational complexity and provide high quality solutions. The features of the multi-agent solver and its commitment based variant have been also evaluated on series of experiments. The experimental results confirm expected properties (stability, computational complexity and solution quality) of the solver. The temporal properties of decommitment rules and influence of the decommitment rules combinations on the plan execution performance and stability is also discussed.

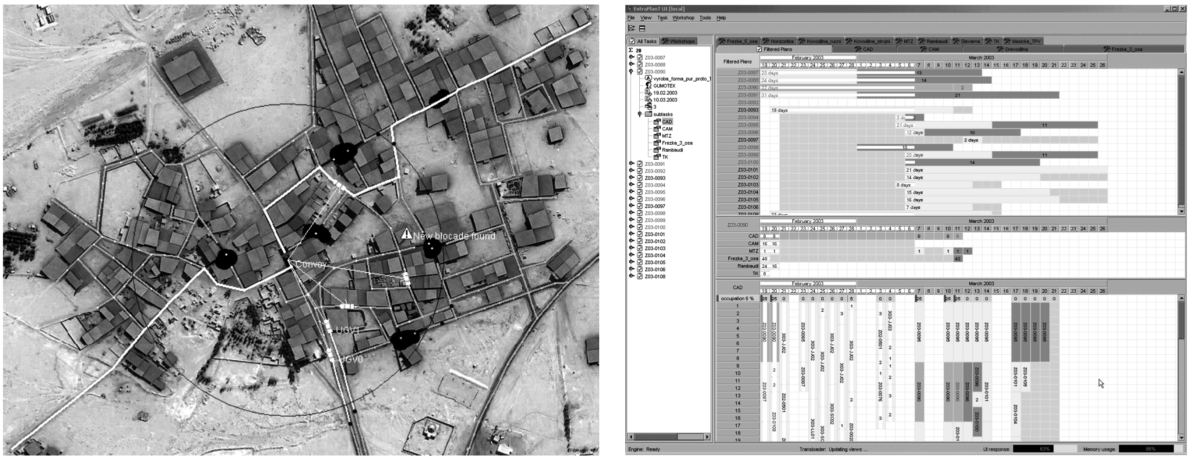
The first application targets the domain of vehicle routing problems, which are well-known optimization problems. The vehicle routing problem is NP-hard and it is defined as routing of a fleet of gasoline delivery trucks between a terminal and a number of service stations. The trucks have load capacity limitations and deliveries have to be accomplished at minimum total cost (distance traveled). The solver provides anytime solution rapidly converging to the quality of at least 81% of the optimal solution. The solver demonstrates very good applicability to the routing problem and easy adaptation to problem variants. Generalization of this approach to other problems (e.g. multiple traveling repairman problem and its variants) seems to be a promising way for further applications mainly in dynamic constrained scenarios. An example route can be seen on Figure 2–a.

The application in the domain of strategic mission planning uses the same solver architecture for distributed mission planning and allocation of particular goals to the Resource Agents. It utilizes commitment based plan representation to provide plan execution stability and robustness. The commitment-based multi-agent problem solver has been used in a system for distributed planning and coordination in dynamic non-deterministic multi-actor mixed-initiative environment for strategic mission planning. The system provides flexible planning, replanning, and task allocation. The system addresses several issues that have to be solved in order to fulfill the requirements on a system planning in dynamic non-deterministic environments. The performance, robustness, and stability of the system has been verified on extended scenario suite (see Figure 2–b). The behavior of the system in the complex interaction scenarios with focus on the planning performance and resources utilization in various settings has been evaluated.



a *b*

Figure 2: (a) An example of the VRP solver result. Circles are demands (size corresponds to amount demanded), optimal route is thin dark line and solver route is thick light line. (b) Strategic mission planning system – scenario island screenshot. There are several agents operating on island (white lines denotes negotiation activity).



a *b*

Figure 3: (a) Cooperative Frontiers Exploration – scenario screenshot. The black spots are obstacles, white thick line is actual convoy route. (b) Screenshot of multi-agent production planning system.

In the cooperative frontiers exploration domain the application is composed of three levels of planning and control. On the topmost deliberative level, the agents implement a variant of dynamic routing problem. The lower levels are responsible for detailed maneuver based path planning of the vehicles in the physical environment. Such environment provides high level of executional uncertainty and creates dynamic feedback to the routing problem solver. The problem is to find the shortest path for a convoy of vehicles through a partially known urban area. The street map of the area is a priori known, but the actual condition of the routes is not. There is a convoy moving through a city using a path-planning algorithm that incorporates the information obtained by a set of small autonomous vehicles. These vehicles explore the area ahead of the convoy – see the scenario screenshot on Figure 3–a. The Convoy Agent dynamically creates points of interests (exploration frontiers) and allocates them to the agents of autonomous vehicles. In the case of any new information is discovered new frontiers are generated and allocated (or some frontiers can be removed). The vehicle agents use a route optimization heuristic that minimizes the traveled distance similar to the vehicles routing problem described earlier.

The production planning domain is one of classical domains of multi-agent problem solving. The system implements the abstract architecture, where each resource (workshop) is represented by one Resource Agent. These agents negotiate about task assignments and cooperatively minimize the weighted lateness of planned tasks. The multi-agent system provides an overview of the utilization of resources and enables to perform a series of what-if scenario. The system provides fast response to the dynamic changes such as resource capacity adjustment or task specification update. The goal of the multi-agent system is to create a production plan for middle and long term horizon (see screenshot of the system on Figure 3–b). The production resources are represented by the Resource Agents maintaining the constraints and capabilities of individual production workshops. The Task Agent uses a classical task ordering heuristic based on weighted earliest deadline first, which significantly improves the solution and there is no need for improvement strategy execution after allocation phase of the batch of tasks. The production order decomposition and planning is provided by the Allocation Agent, which allocates the parts of the production order to the Resource Agents. In case of environment changes (e.g. an update of production times estimation, machines breakdowns, delays in production, etc.) dynamic plan improvement is executed. The solution of the solver is optimal according to the cost computed as a weighted delay penalty.

5 Conclusion

This work presents an abstract multi-agent solver architecture and an algorithm for implementing a wide variety of practical multi-agent planning and problem solving systems. The algorithm maximizing social welfare of a community of cooperative agents is introduced and analyzed. The task allocation and solution improvement using task delegation provides a powerful tool for problem solving while keeping the computational complexity within reasonable limits. We also discuss the limitations and admissability constraints of the Resource Agents' optimization heuristics that has to be designed to implement the multi-agent solver for a particular problem domain. The architecture is also applicable to dynamic problems using a dynamic anytime allocation algorithm modification. Another presented allocation algorithm modification targets heterogenous tasks and Resource Agent capabilities.

An approach to plan representation based on social commitments is introduced. An enhanced solver using commitments is suitable for flexible replanning and plan revision purposes in dynamic non-deterministic environments. The abstract problem solving architecture is extended for commitment allocation and the decommitment model. The decommitment rules definition and their influence on the plan execution robustness and stability are also presented. Each of the presented rules provides a different impact on the agent's current state. For example, relaxation helps to maintain the commitment execution, delegation effectively unblocks the agent's resources and full decommitment releases the agent's resources by dropping the commitment. We expect a combination of the decommitment rules to emerge in a self-adaptation pattern that should lead to some sort of a real-time commitment execution optimization.

The applicability of the abstract multi-agent solver is demonstrated on several real systems operating in the domains of vehicle routing problems, strategic mission planning, multi-robot frontiers exploration, and production planning. In all application areas the implemented system provides low computational complexity ($O(n^3)$) with good solution quality.

Experimental evaluation and validation of presented architectures and algorithms in benchmark settings or real application scenarios has been performed. The experiments confirmed the expected low computational complexity of all implemented scenarios and high quality of obtained solutions. The dynamics and uncertainty of the environment of a particular application scenario are properly handled by the applications utilizing the anytime dynamic allocation algorithm and/or commitment-based plan representation. The experimental validation confirms the ability of the multi-agent problem solver to provide high-quality solutions, performance, stability, and robustness of the system in complex scenarios. The discussion of the experiments results is given at the end of each experiment subsection.

5.1 Thesis Achievements

This section summarizes the contribution of the thesis to the state-of-the-art of multi-agent problem solving. The achieved improvements are the following:

1. An abstract architecture of a multi-agent solver and the respective algorithm providing decomposition, task allocation and task delegation has been introduced. The architecture is based on social welfare maximization using agent negotiation over task allocation, delegation, and reallocation. The interaction mechanism ensures global optimization behavior while the individual agents apply local planning heuristics and strategies.

2. Various features of the abstract architecture, such as computational complexity or admissibility of the underlying optimization heuristics have been analyzed. The worst case computational complexity analysis of the abstract algorithm shows that the influence of the number of Resource Agents is linear and the complexity of the local planning of the Resource Agent is multiplied by the factor n^2, where n is the total number of tasks in the system. The defined Resource Agents' strategy admissability ensures the abstract algorithm to converge even in a dynamic environment.

3. An abstract architecture and algorithm have been enhanced by introducing a social commitment plan representation. It is suitable for flexible replanning and plan revision purposes in dynamic non-deterministic environments. The definition of decommitment rules and their influence on the plan execution robustness and stability have been analyzed. We have formally introduced and discussed three specific decommitment rules: (i) relaxation, (ii) delegation and (iii) full decommitment. We argue that an appropriate selection, setting and preference ordering of the decommitment rules contributes to the robustness of the overall solution.

4. The abstract multi-agent problem solving architecture and algorithms have been evaluated in the vehicle routing problem scenario. 115 standard benchmark instances with known optimal solutions have been used for evaluation. The quality of the solution has been over 81% for all benchmark instances – a result that proves our implementation greatly outperforms a known lower-bound approximation. The time complexity of the solver on experimental instances has been upper-bounded by $O(n^3)$. We have defined the bounds for the Vehicle Agents and shown that thanks to self-organization principles the presented agent-based solver converges to the same solution both when starting from the lower bound and when starting from the upper bound number of vehicles. The solver demonstrates very good applicability to the vehicle routing problem and an easy adaptation to problem variants.

5. The architecture enriched by the social commitments aspects has been evaluated in series of experiments under stress conditions. The experiments show the impact of the execution of particular rules and prove that a complex combination of the decommitment rules provides non-trivial behavior and can potentially improve the performance of the commitments' execution in non-deterministic uncertain environments. The results prove the ability of the system to adapt to overload and thus to increase the number of successful commitments with an increasing size of the decommitment rule set and to keep high utilization of available resources.

6. The developed architecture, algorithms, and mechanisms have been validated on real-world application scenarios to support the applicability of the multi-agent problem solving approach to real problems. The experimental validation confirms the ability of the multi-agent problem solver to provide high-quality solutions, performance, stability, and robustness of the system in complex scenarios. Four instances of the abstract architecture implementations are presented to demonstrate the applicability of the presented approaches in a variety of real problem domains.

References

[1] K. J. Arrow, A. K. Sen, and K. Suzumura, editors. *Handbook of Social Choice and Welfare (Handbooks in Economics).* North Holland, August 2002.

[2] R. I. Brafman and C. Domshlak. From one to many: Planning for loosely coupled multi-agent systems. In *ICAPS*, pages 28–35, 2008.

[3] M. Bratman. *Intention, Plans, and Practical Reason.* Center for the Study of Language and Inf, City, 1999.

[4] B. J. Clement and E. H. Durfee. Top-down search for coordinating the hierarchical plans of multiple agents. In *AGENTS '99: Proceedings of the third annual conference on Autonomous Agents*, pages 252–259, New York, NY, USA, 1999. ACM.

[5] P. R. Cohen and H. J. Levesque. Intention is choice with commitment. *Artif. Intell.*, 42(2-3):213–261, 1990.

[6] K. Decker, V. Lesser, M. V. N. Prasad, and T. Wagner. MACRON: An architecture for multi-agent cooperative information gathering. In T. Finin and J. Mayfield, editors, *Proceedings of the CIKM '95 Workshop on Intelligent Information Agents*, Baltimore, Maryland, 1995.

[7] K. Decker and V. R. Lesser. Generalizing the partial global planning algorithm. *Int. J. Cooperative Inf. Syst.*, 2(2):319–346, 1992.

[8] E. Durfee and V. Lesser. Partial Global Planning: A Coordination Framework for Distributed Hypothesis Formation. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(5):1167–1183, September 1991.

[9] E. Durfee and T. Montgomery. Coordination as distributed search in a hierarchical behavior space. *Systems, Man and Cybernetics, IEEE Transactions on*, 21(6):1363 –1378, nov/dec 1991.

[10] E. H. Durfee. Distributed problem solving and planning. In G. Weiß, editor, *A Modern Approach to Distributed Artificial Intelligence*, chapter 3. The MIT Press, San Francisco, CA, 1999.

[11] K. Erol, J. Hendler, and D. S. Nau. Htn planning: Complexity and expressivity. In *In Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94*, pages 1123–1128. AAAI Press, 1994.

[12] M. Fasli. On commitments, roles, and obligations. In B. Dunin-Keplicz and E. Nawarecki, editors, *CEEMAS 2001*, volume 2296 of *LNAI*, pages 93–102. Springer-Verlag Berlin Heidelberg, 2002.

[13] A. Komenda, M. Pěchouček, J. Bíba, and J. Vokřínek. Planning and re-planning in multi-actors scenarios by means of social commitments. In *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT/ABC 2008)*, volume 3, pages 39–45. IEEE, october 2008.

[14] S. Kraus, K. P. Sycara, and A. Evenchik. Reaching agreements through argumentation: A logical model and implementation. *Artif. Intell.*, 104(1-2):1–69, 1998.

[15] S. Kraus, J. Wilkenfeld, and G. Zlotkin. Multiagent negotiation under time constraints. Technical report, College Park, MD, USA, 1992.

[16] A. Mallya, P. Yolum, and M. Singh. Resolving commitments among autonomous agents. In F. Dignum, editor, *Advances in Agent Communication. International Workshop on Agent Communication Languages, ACL 2003*, volume 2922 of *LNCS*, pages 166–82. Springer-Verlag, Berlin, Germany, 2003.

[17] O'Sullivan and S. M. Sheffrin. *Economics: Principles in Action.* Addison Wesley Longman, 2002.

[18] A. Rubinstein and M. J. Osborne. *Bargaining and Markets.* Emerald Group Publishing Limited, 1990.

[19] R. G. Smith. The contract net protocol: High level communication and control in a distributed problem solver. *In IEEE Transactions on Computers*, C-29(12):1104–1113, 1980.

[20] M. Wooldridge. *An Introduction to MultiAgent Systems.* John Wiley & Sons, Inc., New York, NY, USA, 2002.

[21] M. Wooldridge and N. Jennings. Cooperative problem solving. *Journal of Logics and Computation*, 9(4):563–594, 1999.

Selected Related Publications

This section summarizes the author's selected publications related to the content of the thesis. The authorship is given in parentheses for each publication.

Articles in journals and book chapters:

1. J. Vokřínek, A. Komenda, and M. Pěchouček. Abstract architecture for task-oriented multi-agent problem solving. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 41(1):31 –40, 2011. (80%)

2. A. Komenda, J. Vokřínek, and M. Pěchouček. Plan Representation and Execution in Multi-actor Scenarios by means of Social Commitments. *Web Intelligence and Agent Systems: An International Journal*, accepted for publication in 2011. (∼20%)

3. J. Vokřínek, J. Bíba, J. Hodík, and J. Vybíhal. The RBVO formation protocol. *International Journal of Agent-Oriented Software Engineering*, 3(2/3):135–162, 2009. INDERSCIENCE ENTERPRISES LTD, World Trade Center BLDG. (80%)

4. J. Vokřínek, J. Hodík, M. Pěchouček, P. Bečvář, and J. Pospíšil. ExtraPlanT as a Multi-Agent System for Extra-Enterprise Collaboration, In *Encyclopedia of Networked and Virtual Organizations - vol. I A-F*, pages 593–600. Information science Reference, 2008. (33%)

5. J. Hodík, P. Bečvář, M. Pěchouček, J. Vokřínek, and J. Pospíšil. Explantech and extraplant: multi-agent technology for production planning, simulation and extra-enterprise collaboration. *International Journal of Computer Systems Science and Engineering*, 20(5):357–367, 2005. (20%)

6. M. Pěchouček, J. Vokřínek, and P. Bečvář. Explantech: Multiagent support for manufacturing decision making. *IEEE Intelligent Systems*, 20(1):67–74, 2005. (33%)

7. M. Pěchouček, A. Říha, J. Vokřínek, V. Mařík, and V. Pražma. Explantech: Applying multi-agent systems in production planning. *International Journal of Production Research*, 40(15):3681–3692, 2002. (20%)

In proceedings:

8. J. Vokřínek, A. Komenda, and M. Pěchouček. Agents towards vehicle routing problems. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1 - Volume 1*, AAMAS '10, pages 773–780, Richland, SC, 2010. International Foundation for Autonomous Agents and Multiagent Systems. (80%)

9. J. Vokřínek, A. Komenda, and M. Pěchouček. Cooperative agent navigation in partially unknown urban environments. In *PCAR '10: The Third International Symposium on Practical Cognitive Agents and Robots. Proceedings of the AAMAS-10 Workshops.*, 2010. (40%)

10. J. Vokřínek, A. Komenda, and M. Pěchouček. Decommitting in multi-agent execution in non-deterministic environment: experimental approach. In C. Sierra, C. Castelfranchi, K. S. Decker, and J. S. Sichman, editors, *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009), Budapest, Hungary, May 10-15, 2009, Volume 2*, pages 977–984. IFAAMAS, 2009. (35%)

11. J. Vokřínek, A. Komenda, and M. Pěchouček. Relaxation of social commitments in multi-agent dynamic environment. In *ICAART 2009 - Proceedings of the International Conference on Agents and Artificial Intelligence*, pages 520–525. INSTICC Press, 19-21 January 2009. (55%)

12. A. Komenda, J. Vokřínek, M. Pěchouček, G. Wickler, J. Dalton, and A. Tate. I-globe: Distributed planning and coordination of mixed-initiative activities. In *KSCO '09: Knowledge Systems for Coalition Operations 2009*, Chilworth Manor, Southampton, UK, Mar-Apr 2009. (18%)

13. G. Wickler, A. Komenda, M. Pěchouček, A. Tate, and J. Vokřínek. Multi-agent planning with decommitment. In *KSCO '09: Knowledge Systems for Coalition Operations 2009*, Chilworth Manor, Southampton, UK, Mar-Apr 2009. (10%)

14. A. Komenda, J. Vokrínek, M. Pechoucek, G. Wickler, J. Dalton, and A. Tate. Distributed planning and coordination in non-deterministic environments (demo). In C. Sierra, C. Castelfranchi, K. S. Decker, and J. S. Sichman, editors, *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009), Budapest, Hungary, May 10-15, 2009, Volume 2*, pages 1401–1402. IFAAMAS, 2009. (35%)

15. A. Komenda, M. Pěchouček, J. Bíba, and J. Vokřínek. Planning and re-planning in multi-actors scenarios by means of social commitments. In *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT/ABC 2008)*, volume 3, pages 39–45. IEEE, October 2008. (5%)

16. 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*, pages 656–668, Berlin, 2007. Springer-Verlag. (35%)

17. J. Vokřínek, J. Bíba, J. Hodík, J. Vybíhal, and P. Volf. RBVO formation protocol. In *2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2007)*, pages 454–457, 2007. (35%)

18. M. Pěchouček, J. Vokřínek, P. Bečvář, J. Hodík and J. Pospíšil. ExPlanTech: Multi-agent Framework for Production Planning, Simulation and Supply Chain Management. In *Multiagent System Technologies*, Berlin, 2004. Springer-Verlag. (20%)

19. A. Říha, M. Pěchouček, J. Vokřínek and V. Mařík. Exploitation of Agent-Based Technology in Production Planning. *Multi-Agent-Systems and Applications*, Berlin, 2002. Springer-Verlag. (25%)

20. V. Mařík, M. Pěchouček, J. Vokřínek and A. Říha. Application of Agent Technologies in Extended Enterprise Production Planning. *Lecture Notes in Computer Science*, Proceedings of First EurAsian Conference, Berlin, 2002. Springer-Verlag. (25%)

Response and Reviews

This section summarizes responses to the related author's publications according the Web of Science[1] (self citations are not listed).

E. Adam, T. Berger, Y. Sallez, and D. Trentesaux. Role-based manufacturing control in a holonic multi-agent system. *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*, 49(5):1455–1468, 2011. CITE: 7

O. Lopez-Ortega and K. Lopez de la Cruz. A communication protocol among agents for exchanging data in the extended enterprise. *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY*, 51(1-4):383–396, NOV 2010. CITE: 20

S. Cochrane, R. Young, K. Case, J. Harding, J. Gao, S. Dani, and D. Baxter. Manufacturing knowledge verification in design support systems. *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*, 47(12):3179–3204, 2009. CITE: 6

P. Forget, S. D'Amours, J.-M. Frayret, and J. Gaudreault. Study of the performance of multi-behaviour agents for supply chain planning. *COMPUTERS IN INDUSTRY*, 60(9):698–708, DEC 2009. CITE: 6

F. Bellifemine, G. Caire, A. Poggi, and G. Rimassa. JADE: A software framework for developing multi-agent applications. Lessons learned. *INFORMATION AND SOFTWARE TECHNOLOGY*, 50(1-2):10–21, JAN 2008. CITE: 5

V. Marik and D. McFarlane. Industrial adoption of agent-based technologies. *IEEE INTELLIGENT SYSTEMS*, 20(1), 27-35 JAN-FEB 2005. CITE: 18, 19

T. Holvoet and P. Valckenaers. Exploiting the environment for coordinating agent intentions. In *Environments for Multi-Agent Systems III*, volume 4389 of *LECTURE NOTES IN ARTIFICIAL INTELLIGENCE*, pages 51–66, 2007. CITE: 6

B. Kadar, A. Pfeiffer, L. Monostori. Building agent-based systems in a discrete-event simulation environment. In *LECTURE NOTES IN ARTIFICIAL INTELLIGENCE*,volume of *MULTI-AGENT SYSTEMS AND APPLICATIONS IV*, 2005. CITE: 6

M. Obitko and V. Marik. Integrating transportation ontologies using semantic web languages. In *LECTURE NOTES IN ARTIFICIAL INTELLIGENCE*, 2005. CITE: 6

P. Novak, M. Rollo, J. Hodik, T. Vlcek. Communication security in multi-agent systems. In *LECTURE NOTES IN ARTIFICIAL INTELLIGENCE*,volume of *MULTI-AGENT SYSTEMS AND APPLICATIONS III*, 2003. CITE: 7

L. Lhotska, O. Stepankova. Agent architecture for smart adaptive systems. In *TRANSACTIONS OF THE INSTITUTE OF MEASUREMENT AND CONTROL*, 26(3), pages 245–260, 2004. European Symposium on Intelligent Technologies Hybrid Systems and their Implementation on Smart Adaptive Systems, Albuifeira, PORTUGAL, SEP 19-21, 2002. CITE: 7

J. S. Kinnebrew and G. Biswas. Efficient Allocation of Hierarchically-Decomposable Tasks in a Sensor Web Contract Net. In *IEEE/WIC/ACM International Conferences on Web Intelligence (WI)/Intelligent Agent Technologies (IAT), Milan, Italy, SEP 15-18, 2009.* CITE: 16

J. Trullas-Ledesma and L. Ribas-Xirgo. Transforming SME manufacturing plants into evolvable systems through agents. In *Proceedings of 3rd IEEE International Conference on Digital Ecosystems and Technologies*, pages 136–141, 2009. CITE: 18

S. A. Katriou, G. Fragidis, I. Ignatiadis, E. Tolias, and A. Koumpis. Waste Management Using Request-Based Virtual Organizations. In *IFIP Advances in Information and Communication Technology*, pages 537–544, 2009. CITE: 17

P. Kadera and P. Tichy. Plan, Commit, Execute Protocol in Multi-agent Systems. In *Proceedings of 4th International Conference on Industrial Application of Holonic and Multi-Agent Systems*, volume 5696 of *Lecture*

[1] accessible through http://www.isiwebofknowledge.com/

Notes in Computer Science, pages 155–164, 2009. CITE: 16

R. Mahajan and A. Gupta. Scalable Contract Net Based Resource Allocation Strategies for Grids. In *Proceedings of 9th International Conference on Parallel and Distributed Computing, Applications and Technologies*, pages 25–32, 2008. CITE: 16

P. Forget, S. D'Amours, and J.-M. Frayret. Multi-behavior agent model for planning in supply chains: An application to the lumber industry. In *Proceedings of 7th IFIP International Conference on Information Technology for Balanced Automation Systems in Manufacturing and Services*, 24(5):664–679, OCT 2008. CITE: 6

A. M. D. Shr, A. Liu, and P. P. Chen. Load balancing among photolithography machines in semiconductor manufacturing. In *2006 3rd International IEEE Conference Intelligent Systems, Vols 1 and 2*, pages 314–319, 2006. CITE: 6

P. Golinska, M. Fertsch, J. M. Gomez, and J. Oleskow. The concept of closed-loop supply chain integration through agents-based system. In *Information Technologies in Environmental Engineering*, Environmental Science and Engineering-Environmental Engineering, pages 189–202, 2007. CITE: 7

M. Obitko and V. Marik. Transparent ontological integration of multi-agent systems. In *2006 International Conference on Systems, Man and Cybernetics*, IEEE International Conference on Systems, Man and Cybernetics, pages 2488–2492, 2006. CITE: 6

A. Liu, P. P. Chen, A. M. D. Shr, and Y.-R. Cheng. Using a Multiagent Scheduling system for dedicated machine constraint in semiconductor manufacturing. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 3262–3267, 2006. CITE: 6

L. Monostori, J. Vancza, and S. R. T. Kumara. Agent-based systems for manufacturing. *CIRP ANNALS-MANUFACTURING TECHNOLOGY*, 55(2):697–720, 2006. CITE: 6, 19

H. Helaakoski, P. Iskanius, I. Peltomaa, and J. Kipina. Agile business model in the steel product industry sector. In *IEEE International Conference on Management of Innovation and Technology, Vols 1 and 2*, pages 1010–1014, 2006. CITE: 6

P. Forget, S. D'Amours, and J.-M. Frayret. Collaborative Event Management in supply chains: An agent-based approach. In *Information Technology for Balanced Manufacturing Systems*, pages 89–98, 2006. CITE: 6

A. Svirskas, I. Ignatiadis, B. Roberts, and M. Wilson. Virtual Organization management using Web Service Choreography and Software Agents. In *Network-Centric Collaboration and Supporting Frameworks*, pages 535–542, 2006. CITE: 7

V. Marik, P. Vrba, and M. Fletcher. Agent-based simulation: Mast case study. In *Emerging solutions for future manufacturink systems*, 2004. CITE: 19

J. Trullas-Ledesma and L. Ribas-Xirgo. Transforming SME manufacturing plants into evolvable systems through agents. In *3rd IEEE International Conference on Digital Ecosystems and Technologies*, pages 136–141, 2009. CITE: 18

A. O. Aydin, S. Semra. Intelligent agent technologies promises in emerging arena: mass customisation. In *Proceedings of the 6th Conference on 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases*, volume 6, 2007. CITE: 19

Résumé

Plánování a řešení problémů v decentralizovaném prostředí je klíčový problém v mnoha průmyslových aplikacích, počínaje výrobou, logistikou, virtuálními organizacemi a konče multi-robotickými systémy. Integrace procesu dekompozice úkolů, jejich alokace na agenty a lokálního plánování umožňuje využít metody plánování a alokace s důrazem na dostupnost zdrojů. Každý úkol lze obecně dekomponovat několika způsoby. Pro každou dekompozici existuje mnoho možných alokací a jednotliví agenti mohou vygenerovat rozdílné plány pro každou část úkolu. Otázku zvolení vhodné dekompozice pro daný úkol nelze snadno zodpovědět bez znalosti konkrétní alokace a vyhodnocení lokálních plánů. Jelikož počet možných kombinací značně stoupá s počtem agentů jakožto i s počtem možných dekompozic a alokací dostáváme se k netriviálnímu problému multi-agetního řešení problémů.

V této práci představujeme abstraktní architekturu multi-agentího solveru a odpovídající algoritmy zajišťující dekompozici úloh, jejich alokaci a delegaci. Architektura je založena na principech maximalizace sociálního užitku pomocí agentního vyjednávání o alokaci, delegaci a realokaci úkolů. Interakční mechanismy zaručují optimalizaci v globálním měřítku, přičemž jednotliví agenti používají lokální plánovací heuristiky a strategie. Práce také analyzuje vlastnosti této abstraktní architektury jako je výpočetní složitost nebo přípustnost použitých optimalizačních heuristik.

V další části představujeme reprezentaci plánů pomocí sociálních závazků, která je vhodná zejména pro flexibilní přeplánování a opravy plánů v dynamickém nedeterministickém prostředí. Nosnou myšlenkou je reprezentace distribuovaného hierarchického plánu pomocí sociálních závazků za použití formální teorie pro zachycení vzájemných relací mezi záměry a cíli spolupracujících agentů. Uvažování nad takovými závazky a alternativami jejich možného plnění či rušení přispívá v plánovacím procesu k tvorbě flexibilních a robustních plánů. V práci formálně definujeme a analyzujeme tři základní vyvazovací pravidla: (i) relaxaci, (ii) delegaci a (iii) úplné zrušení závazku. Koordinační a interakční proces mezi agenty je silně vázán na schopnost jednotlivých agentů inteligentně použít vyvazovací pravidla na základě konkrétních změn v prostředí. Ukazujeme, že správný výběr, nastavení a řazení vyvazovacích pravidel přispívá k robustnosti celkového řešení. Výše uvedenou abstraktní architekturu pro agentní řešení problémů jsme obohatili o možnost práce se sociálními závazky a diskutujeme možný vyvazovací model.

V závěru verifikujeme a vyhodnocujeme představené přístupy v simulovaném prostředí i na konkrétních příkladech z reálného světa. Experimentální ověření potvrzuje možnosti multi-agentího řešení problémů poskytnout kvalitní řešení, výkonnost, stabilitu a robustnost systému v komplexních scénářích. Abychom demonstrovali aplikovatelnost předložených přístupů v širokém spektru problémů reálného světa, předkládáme čtyři implementace abstraktní architektury pro konkrétní domény – vehicle routing problémy, strategické plánování misí, multi-robotický průzkum okrajových bodů a plánování výroby.