Decision issues for multiple heterogeneous vehicles in uncertain environments

M. Barbier[1], **H. Cao**[2], **S. Lacroix**[2], **C. Lesire**[1], **F. Teichteil-Königsbuch**[1] **and C. Tessier**[1]

[1] *ONERA Centre de Toulouse – DCSD*
2 avenue Édouard Belin, B.P. 4025
F-31055 TOULOUSE CEDEX 4

[2] *LAAS/CNRS*
University of Toulouse
7 avenue du Colonel Roche
F-31077 TOULOUSE Cedex 4

Abstract

This paper focuses on on-going work in the "Action" project related to the decisional processes that are required to allow teams of cooperating heterogeneous vehicles to achieve target localization and tracking missions. Autonomy is made possible thanks to embedded functions such as data fusion, planning and supervision, organized within each vehicle so that they can cooperate together. Two essential characteristics are taken into account: the uncertainty on the environment models and on the vehicle action models (motions, perception and communication) and the constraints on the inter-vehicle communications, that cannot be permanently maintained. The paper presents the overall objectives of the Action project, analyzes the raised problems, and sketches the global approach chosen for the planning and supervision processes.

Keywords

Multi-vehicle missions, Decisional autonomy, Distributed planning and supervision.

1 Introduction

A lot of research work has been devoted to various types of uninhabited ground, aerial, sea and space vehicles, for both military and civil applications, aiming at endowing them with autonomous abilities to achieve operational missions. The fact that the acronym "UxV" for such vehicles (Uninhabited or Unmanned Vehicle) is being replaced by "AxV" (Autonomous Vehicle) highlights the evolution of research towards *decisional autonomy*, which is intended to reduce the number of remote operators, to change their roles and to decrease their workload.

The robotics and AI literature abounds in contributions targeted to vehicle autonomy, but recently multiple cooperating autonomous vehicles have raised an increasing interest. Multiple vehicles bring obvious benefits from an operational point of view: they increase the mission profit while offering more robustness with respect to vehicle failures. Deploying multiple *heterogeneous* vehicles yields even better properties: the motion and perception complementarities of various UxVs allow synergies and extend the operational abilities, be it for monitoring, surveillance or exploration missions. But such systems require complex cooperation schemes, especially when communication constraints among the vehicles prevent planning and supervision to rely on centralized solutions: these processes have to be distributed among the vehicles according to cooperation schemes that maintain a consistent behavior for the overall system.

This paper presents on-going work on the planning and supervision framework for a team of autonomous heterogeneous vehicles cooperating to perform target detection and tracking missions.

1.1 Context

This work is supported by the Action project, funded by the DGA, the French Defence procurement agency, and supervised by the LRBA, the expertise centre for missile navigation)[1]. The project started in January 2007 and should last 7 years. The project team is composed of researchers and engineers from Onera, the French Aerospace Lab and Laas, a CNRS research laboratory.

The main goal of the Action project is to develop and experiment means to endow a team of heterogeneous vehicles to autonomously cooperate in order to localize entities evolving in the operation theatre. An essential requirement is to be as independent as possible from external global localization means – namely GPS. Scientific work in the project is set up around two groups of functions:

- data fusion functions: data processing in order to localize the vehicles and the targets, and to gather information on the environment;
- decision making functions: selection of the vehicle actions to perform and supervision of their execution.

The developments will be assessed through demonstrations based on a set of scenarios that are representative of operational contexts, that define missions driven by target localization (scenarios are akin to *search and rescue* missions). Six scenarios have been defined in both aeroterrestrial contexts (that involve AAVs[2] and AGVs[3]) and aeromaritime contexts (that involve an AAV, an ASV[4] and an AUV[5]). The scenarios are defined with incremental cooperation complexity:

1. cooperation within a two-vehicle team: an AAV-AGV team in the aeroterrestrial context and an AAV-AUV team in the aeromaritime context;
2. information sharing among three vehicles: one AAV and two AGVs (aeroterrestrial), and one AAV, one AUV and one ASV (aeromaritime);
3. coordination of two heterogeneous AAV-AGV teams (aeroterrestrial);
4. flotilla management: cooperation of four AAV-AGV-AGV teams (aeroterrestrial).

[1] As this project is multidisciplinary, other DGA expertise centres are involved in the project supervision: CEP (Paris) for information processing, ETAS (Angers) for land systems, CTSN (Toulon) for naval systems and GESMA (Brest) for sub-marine fighting.

[2] Autonomous Aerial Vehicles

[3] Autonomous Ground Vehicles

[4] Autonomous Surface Vehicle

[5] Autonomous Underwater Vehicle

The scenarios involve experimental vehicles owned by the project partners (AAVs and AGVs) or lent by the DGA (ASV and AUV). All vehicles can move autonomously in their environment: the Action project focuses on a hierarchical collaboration of vehicles for mission completion.

1.2 Outline

The following section briefly analyzes the problem and highlights the requirements on the decision making functions, and on their organization. Section 3 sketches the choices made for data fusion among the vehicles. Although these functions are not decisional processes, they provide the basis on which the vehicles plan their actions and cooperate. Section 4 presents the two approaches to the planning problem we are currently developing: an approach that mixes decision-theoretic planning with Hierarchical Tasks Networks, and an approach that is centered on contract-net task allocation schemes. Finally section 5 focuses on the way decisional processes and task execution are supervised.

2 Problem description and analysis

All the project scenarios have the following characteristics:

- the goal is to localize one or several targets, human foes (aeroterrestrial scenarios) or mines and wrecks (aeromaritime scenarios) and to track them (aeroterrestrial scenarios);
- besides mission preparation, each scenario involves two phases: a detection phase that is akin to an exploration mission (even if prior models of the environment are available) and a localization phase that includes target tracking (aeroterrestrial scenarios);
- the vehicles are heterogeneous as they have different motion and perception capabilities;
- communications between two vehicles and the operator are constrained by geometric criteria;
- the models of the environment and of the vehicle actions are uncertain (in particular, GPS localization is not available throughout the mission, which calls for environment-based localization techniques – see section 3).

The operator's role is to specify the mission by defining the operation theatre and to monitor the mission execution. Consequently they must be regularly informed on the progress of the mission (areas explored so far, vehicle positions...)[6]

From the decisional point of view, the most important characteristics are the communication constraints and the uncertainty of the models.

Communication constraints result in the fact that no centralized approach can be considered. This calls for the distribution of the various processes and the explicit consideration of communications in the actions to plan and execute.

[6]The issues raised by the possibility for the operator to *intervene* in the mission are not in the scope of the project.

Model uncertainty, and especially environment model uncertainty, does not allow a priori defined complete plans to be executed as the vehicles must regularly update their plans according to the evolution of their knowledge and to the asynchronous occurrence of disruptive events. Consequently perception actions for knowledge update have to be planned and executed as other actions.

Therefore the following actions have to be planned, scheduled and distributed among the vehicles:

- motion actions,
- communication actions (between vehicles and with the operator),
- perception actions and
- plan computation itself.

Decision capabilities are ensured by task planning on the one hand and by supervision on the other hand. Supervision plays a central role: besides controling the proper execution of the planned actions, supervision analyzes the vehicle state through situation assessment in order to trigger replanning. Figure 1 shows the data flows between the various components embedded on board each vehicle. Note that communications among vehicles are defined for each component.

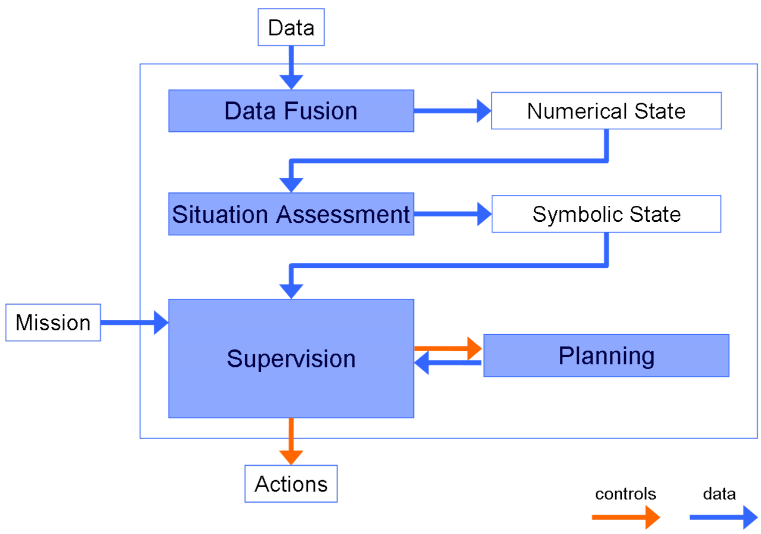

Figure 1: Data and control flows within the main processes embedded in one vehicle.

The vehicle state knowledge is the basis of the decisional architecture: the data fusion function updates the numerical state whereas the situation assessment function elaborates the symbolic state. The multivehicle cooperation problem can be defined as an information gathering problem: "what state to estimate in which agent and to communicate to which agent?".

3 Data fusion

Data fusion encompasses all the processes that integrate data from the environment and the vehicle states in order to *(i)* estimate the localization of the vehicles and targets and *(ii)* to build models of the environment that are exploited by the decisional processes. These data are mainly acquired by the vehicles with their on-board sensors, but prior information on the environment is also considered.

3.1 Vehicle localization

Since the first outcome of the missions is the target localization, vehicle localization is of major importance. In the absence of external absolute localization means or of precise a priori maps, the only way to reduce errors on the position estimates brought by the vehicles sensors (*e.g.* inertial sensing, odometry for AGVs...) and to maintain a consistent position estimate is to rely on environment features (landmarks) that are mapped as the vehicles evolves. The fact that the mapping and the localization problems are intimately tied together has lead to the development of Simultaneous Localisation And Mapping (SLAM) approaches so as to solve them concurrently in a unified manner (see [2] for a survey).

To tackle the problem in a distributed multivehicle context, we have extended the notion of hierarchical SLAM introduced in [7]: each vehicle builds a collection of local landmark maps whose spatial organization is defined by a graph of uncertain frame transformations (figure 2). Various "events" can lead to the evolution of the graph, that correspond to the establishment of cycle[7]: when a vehicle matches landmakrs in a previously built map, when a GPS fix occurs, or when an absolute position estimate is produced by matching detected features with geo-localized features contained in an a priori map. But loop closures can also be detected in the overall graph defined by all the vehicles individual graphs, *e.g.* when one vehicle localizes another one, or when submaps built independently by two different vehicles are matched [12].

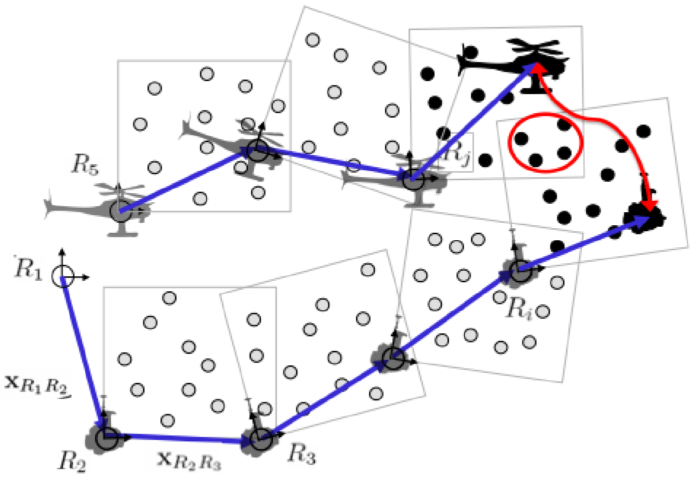

Figure 2: Hierarchical SLAM approach in a multivehicle context. Each vehicle maintains a sequence of submaps of landmarks, the spatial organization between the submaps and among the vehicle is ensured by a higher level spatial graph of frame transformations. Here, the red arrow represents an inter-vehicle loop closure, obtained by the matching of landmarks mapped by the two vehicles.

The important fact is that the approach is *distributed*: each vehicle builds and maintain submaps

[7]literally, a "loop-closing".

and the associated graph, and when communications are available, only few data exchanges are required to detect inter-vehicle loop closures.

3.2 Environment models

Landmark maps are required to maintain consistent localization estimates but only represent a small subset of the environment properties. Other information on the environment is required for the decision processes to assess the current situation and to plan the vehicle activities (figure 3):

- a traversability map is used to evaluate the feasibility and cost of motions,
- a 3D model is exploited to plan environment perception tasks, target detection tasks and inter-vehicle communication tasks[8] and
- context information is exploited to estimate the probability of target locations and to predict their evolution.

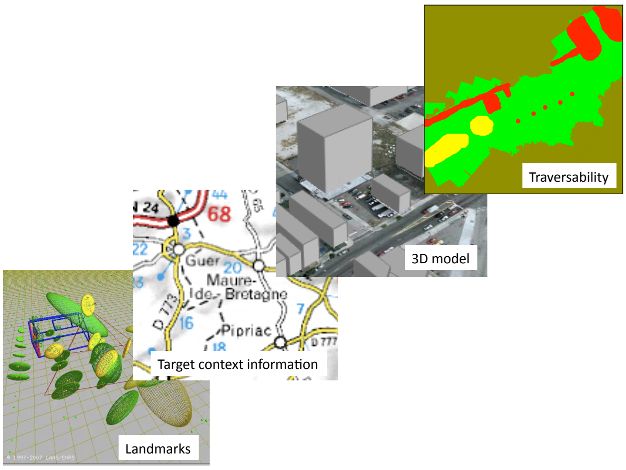

Figure 3: The four environment models required by the aeroterrestrial scenarios.

As a consequence, the environment is represented by a collection of layers, as in a Geographic Information System. To ensure that they are spatially consistent, each of this layer is decomposed into local submaps, according to the spatial structure defined by the SLAM approach.

3.3 Perception models

Besides motion actions, the vehicle perception actions have to be planned. Therefore models are defined for each vehicle perception ability: target detection, target localization, vehicle localization, plus a model of each environment modeling functionality. Given the nature of the sensors and the environment models, these models are stochastic, and express the quality or quantity of information brought by each perception action.

[8]A conservative model of visual inter-visibility is used as a communication model.

4 Planning

In this multiagent localization problem, many team decisions must be taken in order to generate the vehicle motion paths in the environment and to synchronize individual decisions and actions. In other words the entire autonomous system must automatically produce geometrical and communication decisions according to events that may occur during the mission. Many questions arise concerning the different possible ways to tackle multiagent autonomous localization problems:

- should decisions be produced in a reactive way as new events occur or should they be robust to new events and therefore conditionally depend on new events that may occur?
- should some decisions be produced off-line during mission preparation and some others be generated on-line?
- in case of on-line replanning, should we assume that there is a leader vehicle that will generate a new team plan for the whole team or will individual vehicles replan for themselves?
- should decisions be optimal at the expense of costly computations or may suboptimal but efficient decisions be acceptable in some cases?
- what is the abstraction level of planning models? more precisely, will decision and environment variables be all discrete, or all continuous, or both continuous and discrete?
- are we going to handle uncertain action effects or uncertain observations in the planning models, knowing that such models lead to very costly computations?

In fact, there are no definitive answers to the previous questions, and many options are possible and even complement one other. We are considering the development of a planning framework along two directions: by mixing HTNs and MDPs on the one hand, and using a task allocation scheme on the other hand.

4.1 DEC-MDPs & HTNs

This section describes the proactive multiagent planning method we implement. Before going into further details, here is an overview of its main characteristics:

- we mix purely reactive planning algorithms, that assume that the environment is locally deterministic and that replan on-line in case of unpredicted action consequences, and proactive planning methods, that produce conditional action plans (either off-line or on-line);
- a proactive planning algorithm generates a conditional team plan during mission preparation; this conditional plan is automatically translated into decentralized state machines that are executed in each vehicle;
- since all events cannot be taken into account by the proactive method, decentralized on-line (reactive or proactive) planning techniques are used to generate new plans if some events occur that were not considered or known during mission preparation;

- since communications between the vehicles are not always possible, the on-line decision repair methods previously mentioned must be decentralized, i.e. individually run by each vehicle knowing its local environment.

Our approach aims at automatically generating decentralized hierarchical supervisors during mission preparation, but using such proactive techniques on-line to replace parts of some vehicle supervisors will be considered later on.

A supervisor can be viewed as a fixed conditional plan that is hand-coded in a vehicle decisional architecture. It controls the vehicle high-level behavior according to the current state, exactly as a conditional plan (or *policy*) produced by some proactive planning algorithm. Consequently we focus on the automated off-line generation of vehicle supervisors by means of proactive planning algorithms.

4.1.1 Decentralized Markov Decision Processes

Markov Decision Processes [10] (MDPs) are a popular model for (monoagent) planning under uncertainty with quantified probabilistic effects on actions. The fundamental assumption of this model is that the probability of the next state, knowing the performed action and the current state, only depends on the current state but not on the entire state history. Note that this property can always be satisfied if "history" state variables are added in the state space.

The optimization of a criterion, generally defined as the mean of discounted cumulated rewards over all probabilistic infinite-step trajectories, leads to the automated generation of an optimal conditional plan. The latter, named *policy*, can be translated into a finite state automaton, which is the supervisor formalism.

Extensions of this monoagent model to multiagent frameworks have been extensively studied. Decentralized Markov Decision Processes [3, 4] (DEC-MDPs) represent multiagent planning problems with probabilistic effects on the agents' actions. While the multiagent policy is centrally optimized, its execution can be performed in a decentralized way by each agent. Nevertheless the complexity of DEC-MDP optimization is NEXP-complete for more than three agents, which makes this model generally useless for realistic multiagent planning problems. Most authors simplify the model by assuming action transitions dependency and no communication between the agents [3].

Yet, recent advances in hierarchical MDPs, as described in the next paragraph, have proved to improve MDP optimization performance by many orders of magnitude in terms of computation time and memory usage. Consequently we are currently extending this research to hierarchical DEC-MDPs in order to solve realistic probabilistic multiagent planning problems.

4.1.2 Hierarchical Task Networks

Hierarchical Task Networks [6] are a hierarchical planning model where actions are organized as a hierarchical action tree. Each action can be decomposed in simpler actions that can be executed either in parallel or in sequence. A leaf of the action tree is named an *elementary action* because it is not further decomposed. Other actions are named *tasks*. Contrary to non-hierarchical planning frameworks, HTN algorithms use the action hierarchy to ease the search of state trajectories leading to goal states. For instance, a sequence of actions directly encoded in the HTN spares the planner the reconstruction of the action sequence.

Originally formalized for deterministic planning problems, HTNs have been successfully extended to probabilistic planning in [8]. By adding probabilistic effects on elementary actions, HTN action trees have been used in conjunction with forward probabilistic planners to prune useless actions and state trajectories. As a result, the performance of these forward probabilistic planners have been increased by many orders of magnitude.

Our work consists in extending probabilistic HTNs to multiagent probabilistic planning problems. It seems to us action hierarchies can be wittingly used to model team tasks or pre-compiled communication strategies. In other words the HTN formalism should help the optimization of DEC-MDPs by adding knowledge about team and communication strategies in order to suppress expensive computations.

Finally the HTN model is a natural formalism to define structured planning problems where some behaviors are required by the human operator: for instance, exploring a given area with an AGV and na AAV requires the AAV to globally search the area for some obstacles, then the AGV to plan a path between two ground points within a sub-area extracted by the AAV. Combined with probabilistic reasoning, the HTN model provides a human-supervised formalism to optimize structured team strategies.

4.2 Task allocation based planning

We are also considering the planning issues from the task allocation point of view. The main idea is to solve the team planning problem by individual planning and by negotiation among vehicles to coordinate and to cooperatively enhance individual plans, hence defining a global distributed team plan[9].

This choice is motivated by the fact that the scenarios raise complex team planning problems that can hardly be solved in an optimal manner with a centralized approach. The fact that the models are incomplete and uncertain indeed calls for the ability to *dynamically replan* vehicle tasks, which cannot be dealt with in a centralized manner because of communication constraints.

Using a task allocation scheme is somewhat akin to an optimization problem: the team starts with a non optimal plan divided into individual plans. During the execution, the vehicles enhance their plans through negotiation, by buying or selling their tasks depending on the evolution of their knowledge of the environment and of their plan execution. An essential property of such a scheme is its distributed nature: vehicles negotiate only when they can communicate and negotiation may occur only within a subset of the deployed vehicles.

However existing task allocation based architectures are not straightforwardly applicable to our scenarios, mainly because of the communication constraints. Some approaches deal with this issue, either considering communications as a constraint to satisfy or a utility to optimize. In [9], the vehicles have to evolve while maintaining a MANET network. This is a strong constraint that produces team configurations in which every vehicle must stay close to the others, which leads

[9]Note that this approach is only relevant for the aeroterrestrial scenarios, in which both the AAVs and the AGVs can participate in the target detection and environment exploration tasks – whereas in aeromaritime scenarios, each type of vehicle has a fixed predefined role.

to non optimal covering or exploration schemes. In [1, 11], a utility is associated with communications: the actions (namely motions) of each vehicle are rewarded if they allow this vehicle to communicate with other vehicles, and penalized otherwise. Hence there is no guarantee on the communications, whereas the operational context requires the satisfaction of communication constraints – such as regular communications between the vehicles and the operator for mission monitoring purposes.

We propose a negotiation-based architecture that explicitly considers communications as one of the possible vehicle actions – the others being environment perception and motions. Its main properties are the following:

- A vehicle plan is represented by a *temporal* HTN-like plan: an AND node between two tasks can encode a precedence order (HTN-like plans have already been considered in a task allocation scheme in [13], but without considering temporal constraints).

- Communication tasks are explicitly modeled: besides the communication constraints, their outcome, *i.e.* the information they allow to exchange, is modeled. Therefore various communication tasks are defined: to enhance the knowledge of one vehicle about the environment, to refine vehicle or target position estimates, to ensure the synchronization of the plan execution between vehicles, and to set up a negotiation between vehicles. Communications then become a task that is negotiated and inserted in the vehicle plans, similarly to motion and perception tasks.

Within each vehicle, the various processes to allow the development of a task allocation scheme are organized into the following components: an *individual planner* evaluates the cost and utility of inserting tasks, a *negotiation component* instantiates the negotiations for task allocation and opportunistic cooperation, an *execution component* monitors the task execution, including the coordinations with other vehicles, and a *decisional component* is in charge of triggering negotiations according to the situation assessment – and in particular the current execution status.

5 Supervision

Supervision is a central component of the architecture as it controls the execution of elementary actions (motion, perception, communication), receives action plans – or policies – from the planning processes, and reacts to incoming events. We can distinguish two main topics:

1. The synthesis of a supervisor from the off-line planning processes: as discussed before, the planning process could provide off-line a plan or a policy for the whole team to accomplish the mission; while distributed (i.e. the vehicle actions are identified in the plan), the plan needs first to be made hierarchical, so that we could define an agent hierarchy within the team, and second connected to the supervisor of each vehicle.

2. The handling of disruptive events: even with a robust policy-based planner, we will not avoid disruptive events during mission execution; the supervisor must be able to handle disruptive events so that the mission could be performed.

5.1 Off-line supervisor synthesis

The off-line planning process provides an off-line global policy, which is made hierarchical through the HTN decomposition of the mission tasks. The actions of this policy are already allocated to specific vehicles of the team. We can then allocate local plans to the vehicles but we need more organization to handle authority into the team, for instance to tackle communication or replanning processes. Consequently a team hierarchy organized as a set of sub-teams has to be considered.

The work proposed in [5] computes this hierarchy using a global plan described by a Petri net. The same kind of abstraction could be performed using the HTN description of the mission tasks. The objective is then to synthetize, from a HTN-based policy, a set of hierarchical and synchronized state machines that will be distributed on the vehicles.

Moreover, these individual state machines must be connected into the vehicle supervisors. This connection should consider:

- the possibility to translate state machines into each supervisor formalism (e.g., automata, Petri nets or procedures);
- the link with action execution, in terms of methods the supervisor has to call, and returns of these methods, in terms of action success (or failure) and resulting states.

5.2 Handling disruptive events

With a pro-active planner producing a robust policy most of the events are already handled by the synthetized supervisor. However some disruptive events that were unknown (i.e. not considered in the planning process), or misknown (their a priori models were false) may have serious consequences on the mission.

These events, dealt with as *exceptions* during execution, are thrown up in the supervisor hierarchy (that corresponds to an agent hierarchy) until a way to catch it is found, i.e. a set of actions to perform to go on with the mission.

These actions may be local (e.g., finding a new trajectory to reach next point without impacting the sub-team plan), involve several agents into a "reactive" task (e.g., localization of an AGV by an AAV), or need a replanning of (sub-)team actions.

6 Discussion

We presented first insights on the development of decisional processes to allow the deployment of teams of heterogeneous vehicles. Uncertainties in the environment and communication constraints raise a wide spectrum of difficulties, and in particular voids the a priori definition of complete action plans: a good balance between planning and reacting has to be set, depending on the situation at hand. Supervision plays here a crucial role, as it controls the planning and communication abilities of the vehicles.

Work is only at a preliminary stage, and further developments will refine the definition of solutions. In parallel, experimental scenarios with incremental complexity are defined and experimented: actual demonstrations constitute a big challenge and will drive our developments.

References

[1] N. Atay and O. B. Bayazit. Emergent task allocation for mobile robots. In *Robotics: Science and Systems*, 2007.

[2] T. Bailey and H. Durrant-Whyte. Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms. *IEEE Robotics and Automation Magazine*, 13(2):99–110, June 2006.

[3] Raphen Becker, Shlomo Zilberstein, Victor Lesser, and Claudia V. Goldman. Transition-independent Decentralized Markov Decision Processes. In *AAMAS'03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 41–48, New York, NY, USA, 2003. ACM.

[4] Daniel S. Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of Decentralized Control of Markov Decision Process. In *Mathematics of Operations Research*, 27(4):819–840, 2002.

[5] Olivier Bonnet-Torrès and Catherine Tessier. A formal petri-net based model for team monitoring. In V. Dignum, editor, *Multi-Agent Systems: Semantics and Dynamics of Organizational Models*, chapter 23. IGI, 2009.

[6] Kutluhan Erol, James Hendler, and Dana S. Nau. Semantics for hierarchical task-network planning. Technical report, College Park, MD, USA, 1994.

[7] C. Estrada, J. Neira, and J.D. Tardós. Hierarchical SLAM: real-time accurate mapping of large environments. *IEEE Transactions on Robotics*, 21(4):588–596, Aug. 2005.

[8] Ugur Kuter and Dana S. Nau. Using domain-configurable search control for probabilistic planning. In *AAAI*, Pittsburgh, Pennsylvania, USA, July 2005.

[9] A. R. Mosteo., L. Montano, and M. G. Lagoudakis. Multi-robot routing under limited communication range. *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on Robotics and Automation*, pages 1531–1536, May 2008.

[10] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1994.

[11] N. Rooker and A. Birk. Multi-robot exploration under the constraints of wireless networking. *Control Engineering Practice*, 15(4):435 – 445, 2007.

[12] T. Vidal-Calleja, C. Berger, and S. Lacroix. Event-driven loop closure in multi-robot mappinp. In *Submitted to International Conference on Intelligent Robots and Systems, St Louis, MO (USA)*, Oct. 2009.

[13] R. Zlot. *An Auction-Based Approach to Complex Task Allocation for Multirobot Teams*. PhD thesis, Carnegie Mellon University, 2005.