Complex Tasks Allocation for Multi Robot Teams under Communication Constraints

Hung CAO, Simon LACROIX, Félix INGRAND, Rachid ALAMI
CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France
Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ;
F-31077 Toulouse, France

Abstract

The Multirobot Task Allocation (MRTA) paradigm is widely used in multirobot cooperation schemes, e.g. for observation, surveillance and tracking missions. Market-based approaches have yielded effective distributed solutions for such missions, showing the ability to manage heterogeneous, dynamic and robust robot teams. Two major challenges remain however poorly tackled: the management of inter-robot and inter-task communication constraints, and the use of a rich task formalism to model complex missions.
This paper presents our investigations to treat these two aspects. The inter-robot and inter-task communication constraints are explicitly handled in the task allocation process, through simple geometric models and thanks to temporal scheduling skills. Rich tasks are allocated using a tree-based task formalism that allows to treat complex missions with task ordering. Current work has shown it to be able to handle more complex tasks and to give better solution than MRTA systems working on simple task structures. In our work we will try to go further in this investigation.

Keywords

Complex Task allocation ; Hierarchical Task; Communication constraints; Temporal Planning.

1 Introduction

Research on cooperative multirobot systems has now been very active for a while [1, 21, 10, 6]. Multirobot planning problems are generally hard, many of them being NP-hard, and therefore hardly tractable. Many efforts have been put on proposing either centralized or distributed approaches to overcome the computational burden and to make multirobot planning possible. But besides planning, setting up multirobot systems requires coordinated execution and cooperative supervision: work is required to bring theoretic planning results into real robotic platforms, evolving in actual environments and achieving complex missions. These challenges are compounded by requirements that include operation in dynamic environments, imperfect information on the environment and robot states, and unreliable or limited communications.

The limit on communication's range is an essential constraint to consider: the assumption of permanent communication only holds when the mission is restricted within a small geographical area, and not for teams evolving in large and complex environments, in which obstacles can hinder communications. As communications are required to cooperate, it is an activity that has to be explicitly planned, considering the geometric and temporal constraints that define their feasibility. A task formalism that handles geometric and temporal reasoning in planning is therefore required. Such a formalism should be expressive enough to model the ordering between tasks and elementary temporal features like duration of the actions, while being simple enough to prevent heavy computational burden.

In this paper, we look into solutions to the multirobot task allocation that explicitly handle the range and line-of-sight communication constraints between robots. Unlike approaches that consider communications as a hard constraint to cope with, we consider communications as a parameter in the task allocation process. The robots reason on communications through geometric and temporal reasoning to decide where, when and with whom to meet, according to the needs of negotiation and task execution synchronization. We also investigate the use of a hierarchical task formalism that fits the constraints imposed by the reasoning on communications, and study its impact of the solution quality of the multirobot plan.

The next section reviews the MRTA approaches that are related to the two challenges we consider. Section 3 introduces and formulates our exploration problem, and our overall approach is sketched in section 4. Sections 5 and 6 are respectively dedicated to the *task allocator* and the *individual scheduler* with which each robot is endowed.

2 Related Work

A wide spectrum of approaches has emerged for multirobot system. At one end of the spectrum, fully centralized approaches employ a single agent to coordinate the entire team. In theory, this agent can produce optimal solutions by gathering all relevant information and afterwards planning for the entire team. But centralized approaches are rarely tractable for large teams, suffer from a single point of failure, have high communication demands, and are usually sluggish to respond to unpredicted events.

At the other end of the spectrum, in fully-distributed systems, robots rely only on local knowledge. Such approaches are typically very fast, flexible to change, and robust to failures. But they can produce highly suboptimal solutions since good local solutions are hardly likely to yield a good global solution. Applications where large teams carry out relatively simple tasks with no strict requirements for efficiency are best served by fully distributed coordination schemes.

A vast majority of coordination approaches mix centralized and distributed elements, and thus reside in the middle of the spectrum. Market-based task allocation approaches fall into this hybrid category, and, if designed well, can opportunistically adapt to dynamic conditions to produce more centralized or more distributed solutions.

2.1 Multirobot Task Allocation

Multirobot task allocation (MRTA) has been investigated extensively in the multirobot planning branch. This problem addresses the question of finding the task-to-robot assignments subject to resources constraints that optimize global cost or utility objectives. It can be viewed as a constraint optimization problem with the objective of maximizing rewards from various goals. If we assume that rewards vary linearly with time, the problem can be formulated as a mixed integer linear programming (MILP) problem incorporating scheduling, task allocation, and path planning constraints.

Among the numerous methods that solve this problem, market-based approaches is a very promising one. In these approaches, each distributed agent computes a cost for completing a task, and broadcasts the bid for that task. The auctioneer robot decides the best available bid, and the winning bidder attempts to perform the task won. They effectively meet the practical demands of robot teams, while producing efficient solutions by capturing the respective strengths of both distributed and centralized approaches. First, they can distribute much of the planning and execution over the team and thereby retain the benefits of distributed approaches, including robustness, flexibility, and speed. They also have elements of centralized systems to produce better solutions: auctions concisely gather information about the team and distribute resources in a team-aware context. Following the contract-net protocol [26], several variations of this method has been derived [17, 11, 4, 8].

Another important approach relies on a behavior-based architecture. ALLIANCE [20] is a behavior-based architecture where robots use motivational behaviors such as robot impatience and robot acquiescence. These behaviors motivate robots to perform tasks that cannot be done by other robots, and give up the tasks they cannot perform efficiently. BLE [29] is another behavior-based architecture which uses continuous monitoring of tasks among robots and the best fit robot is assigned to each task. A detailed analysis and comparison of these methods can be found at [12].

[15] proposes a taxonomy of multirobot task allocation problems, depending on whether the robots are capable of performing a single task (ST) or multiple tasks (MT) at a time, whether goals require exactly a single (SR) or multiple (MR) robots to be achieved, and whether task allocations are instantaneous (IA) or include time-extended scheduling (TA). While this taxonomy ignores heterogeneity, it is useful for comparing our work, which falls into the ST-MR-TA category, with existing systems. Market-based planners such as MURDOCH [11] and TraderBots [8] do not permit sub-teams to work on a task and so fall into the ST-SR-IA and ST-SR-TA categories respectively. Neither of these systems provide error bounds on the solution quality. When a single robot is allocated to each task, interleaving task allocation and scheduling is sufficient and the problems do not need to be considered simultaneously. Recent work by [25] extends the TraderBots system to include time varying rewards and heterogeneous robots.

2.2 Communication constraints

Constraints on communications, mainly limitations on communication range, are a reality one can not ignore, and an important obstacle for better cooperation. They are difficult to consider because they imply geometric reasoning, and handling them concurrently with symbolic reasoning is an issue. Various attempts aimed at including these constraints into the planning process, in centralized or distributed manners.

The simplest approaches opportunistically take advantage of network connectivity when available [5], but do not explicitly avoid network splits dues to communication link loss between two team members. In further work, a behavioral approach has been proposed, where connectivity maintenance is addressed, but is not guaranteed [28].

Other approaches rely on assumptions about the signal decay function [27] or the line-of-sight view [16, 14, 2]. However, it is known [7] that such models can badly misrepresent the real behavior of the signal. A further possibility is to place task generation besides task allocation. In exploration missions for example, goals may be decided as a result of cost functions that depend on signal quality [22, 3]. This is difficult to carry over to more general applications, where tasks are provided by external sources and, thus, cannot be created based on system preferences. A solution to this problem is to allow robots to autonomously explore beyond communication range limits. This can be implemented in terms of clustering behavior, in which groups of robots stay close together as they explore the environment [22].

Communication constraints could be considered as inviolable and many methods have been proposed in that purpose. In [9], this problem is formalized and solved as distributed constraint satisfaction problems (disCSP) which are an extension of classical constraint satisfaction problems (CSP). This method is fully distributed and has been tested in the exploration of an unknown environment with maintenance of connectivity between all the members of the team. Another significant distributed method uses task allocation paradigm, in [19], several algorithms for task allocation when network connectivity is an inviolable constraint are presented. This connectivity is maintained by measuring signal strength and the mobility controller is a spring-damper system. Any link in the network spanning tree that falls below a threshold becomes a *virtual spring* that exerts attracting forces between the robots that are about to cause network splits. The mobility controller has a higher priority than the task allocation process, and forces it to allocate only tasks that do not cause potential connectivity loss and also to allocate tasks solely to *robot clusters*. A robot cluster is created when there is a weak link between two robots or two clusters who are going to form a new cluster.

While several of these approaches have been proved successful in maintaining team connectivity during missions, they are usually limited by the constraint of having to keep team members within communication range. Some investigations [23] have been done on the multirobot rendezvous – this work however concern only teams of two robots.

2.3 Complex Task formalism

Task allocation has been shown to be working efficiently on *toy* missions where no hard task decomposition skill is required, *e.g.* multirobot routing or multirobot exploration. The mission is, most of the time, described by a sequence of independent tasks that have to be allocated to different robots. These two sequentiality and independence assumptions do not fit to more complex problems.

We make the following distinction between simple and complex tasks in the context of MRTA: simple tasks are tasks that can be performed in a straightforward, prescriptive manner by executing a direct action or low-level behavior. Navigating from one point to another *along a defined trajectory*, moving a block or opening a door are examples of simple tasks. In contrast, complex tasks require higher-level decision making and can have many potential ways to be achieved. For example, searching a collapsed building during a disaster response scenario may be done in many different ways, depending on the capabilities, state, and other commitments of the robots or the team. Complex tasks generally require the execution of more than one simple task. While both simple and complex tasks can often be decomposed into smaller sub-components, it is the nature of these elements with respect to task allocation that define the type of task.

In contrast to simple and compound tasks, modeling complex tasks at multiple abstraction levels is necessary for a complete task allocation algorithm. If a complex task is treated as a set of simple sub-tasks (one of its full decompositions), this implies that a particular decomposition has been predetermined and the allocation algorithm does not have the flexibility to change this decomposition. If complex tasks are modeled as atomic units, then any allocation involving division of the multirobot-allocatable sub-tasks among multiple robots or sub-teams are ignored. In both cases, most of the possible allocations are removed from the search space, some of which may be preferable to those that are achievable by the allocation algorithm. Given a task set that includes complex tasks, each combination of the possible ways to decompose the complex tasks essentially generates a new simple task allocation problem with the simple sub-tasks resulting from each full decomposition as input. Thus the size of the search space for a complex task allocation problem is exponentially larger than the size of the search space for a simple task allocation problem.

In many domains, tasks are temporally constrained with respect to each other. They may be partially ordered or may need to start or finish within a common time frame. During assignment, robots can incorporate the cost of meeting constraints into their bids [18]. More realistically, they must coordinate during execution to reschedule and accommodate team and task changes since the initial allocation [17, 13].

The systems M+ [4] and DEMiR-CF [24] consider partially ordered plans. However, the bid is only on the set of the current executable tasks, *e.g.* free of all precedence constraints. The whole plan is incrementally distributed to the team during execution via an iterative process of allocation.

Gerkey [11] has used in the MURDOCH system a task formalism where each task is a tree of other tasks. The parent task is responsible of allocating and monitoring the children achievement. Another approach using tree-based formalism is the work of Zlot in TraderBOTs [30], which investigates more aspects than Gerkey's work. On the one hand, it allows the individual decomposition of each robot during the bidding step on a task tree. On the other hand, it permits the re-allocation of tasks (*subcontracting*). This work has shown that complex task allocation can yield better solutions than an MRTA system working on simple task structure.

3 Problem Formulation

We consider a set of m robots $R = \{r_1, .., r_m\}$, not necessarily homogeneous, that are endowed with perception, navigation and communication capabilities. "Perception" here is to be understood in a generic sense: it can for instance be observing for target detection. The overall mission is a surveillance mission, which comes to observe the whole environment, while satisfying given communications constraints – *e.g.* regularly inform a base station about the status of each vehicle.

The environment is represented by a set of models related to the various robot capabilities: a model is dedicated to evaluate the possibility of communications tasks, an other one is dedicated to evaluate the utility of perception tasks, and a third one is dedicated to evaluate the possibility, duration and cost of navigation tasks (*traversability model*). Each map is a represented as a grid, associated to a quad-tree representation to speed up computations. The grid resolution depends on the information it represents and the robot capabilities: for instance, a ground rover navigation model exploits a one meter resolution traversability model to navigate, a coarser resolution model to define its perception task, while a flying vehicle may exploit a much larger resolution grid for perception tasks. All these models are supposed as known.

Navigation model Defined upon the traversability model, the navigation model for a robot allows to quickly estimate a coarse trajectory to move from one location to an other and the associated cost and time. The trajectory is composed of nodes in the graph associated to the traversability model. It is needed to allow a reasoning about the meeting point between two and more robots.

$$cost(r_i, P_{start}, P_{end}) \in [0..1]$$
$$time(r_i, P_{start}, P_{end}) \in [0..1]$$

where P_i denotes the position (x_i, y_i) of a robot.

Communication model Communications are limited in range, and must satisfy line-of-sight visibility to be established. A grid that encodes elevations (3D model, akin to a digital terrain map) is exploited to assess the feasibility of communications between any pair of positions defined in the environment. Two kind of communications are considered: between two robots, and between one robot and a fixed given base station.

$$combase(r_i, P_i) \in \{0, 1\}$$
$$com(r_i, r_j, P_i, P_j) \in \{0, 1\}$$

Perception model The perception model we consider is related to an observation mission. For a robot located at $P_{(x,y)}$, it provides the list of areas (grid cells) that can be perceived – the 3D model used to assess communications is exploited for that purpose.

Robot abilities are represented by their action model. Each action model $A_r = t_1, t_2, .., t_j$ is composed of a set of possible tasks $t_i \in T$ that the considered robot can achieve. Each task t_i has a duration d_{t_i}, and requires either one or more robots to be carried out. If it is a multi-robot task (or *joint* task), it specifies the communication constraints the participants have to respect: it can be a permanent communication, or a communication at the beginning and the end of the task. This is called the *communication modality* of the joint task.

Each complex (compound) task is represented by a tree of another tasks with *AND*/*OR* branching. An *AND* branching means the all children tasks have to be achieved whereas *OR* specifies that only one of the children tasks has to be done. An *AND* branching can contain ordering constraints between children tasks.

Besides the reasoning about the communication for negotiation and synchronization, there are hard communication constraints that each robot must satisfy during the mission execution, that are specified by the user at the beginning of the mission – for instance, establish a communication with the base at given time intervals, to keep the user periodically informed of the robot evolution. Such constraints are initially set in each robot plan.

Given a mission and its decomposition, the planning system should deliver a valid plan for each robot, that respects the initial hard communication constraints. In particular, this plan should contain all the communication actions related to the synchronization between two related tasks achieved by two different robots. For the moment, we do not consider task decomposition, and suppose that this process is done by an available component. A difficult issue for the planner is to define how the communication will be organized between members of the team.

4 Approach Overview

Our approach is centered on the market-based multi-robot task allocation component, which is coupled with an individual planner and an individual scheduler. Besides, a global mission planner

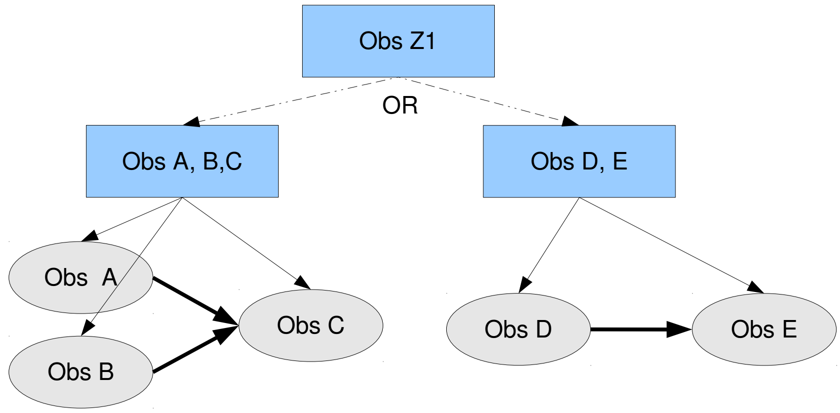

Figure 1: Tree representation of a complex task. The observation task of the zone Z can be decomposed in two tasks: either observe the three locations A,B,C or the two locations D,E. Precedence constraints are denoted by thick arrows: observing A,B,C is decomposed into a partially ordered plan whereas observing D,E requires to achieve the observation of the location E after D.

is responsible for decomposing the mission specification in a plan tree, without assignment of robot to different tasks in the plan. This a priori decomposition takes into account the whole team with its members and their capabilities – we do not consider the design of this component here, because we focus on the task allocation aspects.

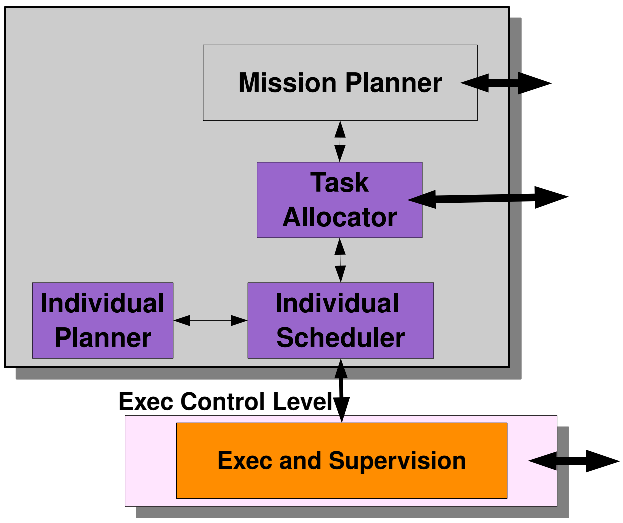

Figure 2: The different components of the planning system.

The role of the *individual scheduler* is to arrange in an optimal manner (time and plan's cost) different atomic tasks of the robots. It is exploited by each robot during the initial mission planning process and during the mission achievement, and reasons about the current plan to insert new tasks in an optimal way. Rather than attempt to ensure the optimality of the computed schedules, we instead employ a search procedure with the goal of quickly find finding acceptable solutions. The scheduler exploits the *individual planner* to decompose the task. This individual planner is an symbolic planner, supported by various refiners for geometric reasoning about navigation, communication and perception.

We do not consider communications as inviolable constraints that the team has to fulfill during the mission, neither as a utility or in an opportunistic way. The communication in our system is treated in three different ways :

- As a task to which an utility is associated. Each robot in the system is encouraged to communicate as often as possible with other robots to maximize its profits. The utility is dynamic and depend a set of parameters *e.g.* the probability of failure for a robot in the team.
- As hard constraints :
  - For the synchronization between two dependent tasks carried out by two different robots.
  - For communications with the base station.

The planning system work in both offline and on-line fashion. Initially, the mission description produced by the mission-planner component is given to the system. This decomposition does not specify any assignment. This assignment-free decomposition is allocated to the team via the allocation process until the whole decomposition is allocated and each robot has a complete plan ready to schedule for execution. The on-line planning does not differ much, but generally concerns only a sub-team because of communication limitations. The tasks to be allocated on-line are of different nature: mission-specific tasks that the current responsible robot is not able or not willing to achieve anymore, communication tasks automatically generated by each robot to maintain the network connectivity, communication tasks needed to plan execution of a particular robot (communication relay to another robot or to the base station). The activation of the overall approach is described by a iterative process composed of the following five steps :

1. The mission is decomposed by Mission-Planner (MP) (within a central station or one robot, possibly with the help of a user)
2. This global plan is distributed to the robot team through task allocation (MRTA) process.
3. Each robot starts to refine and execute its own plan.
4. Robots continuously monitors their plan (cost, validity), its status and listen requests from operator and neighbouring robots : repair plan by adding, removing or redistributing tasks (Individual Scheduler)
5. Loop of negotiation for adding tasks to allocation process / re-distributing task in plan (for plan enhancement).

5 Task Allocator

The task allocator has to deal with complex task structures that requires extensive computation for both the bidders and auctioneers. For a bidder, the computation consists in valuating bids on the tree and possibly re-decomposing these bid's nodes. For an auctioneer, the winner determination algorithm has to search for the optimal assignment in a large solution space because of the significant number of bids. Additionally to the computation, this number of bids can make the communication demand grow quickly. Furthermore, the task allocator must also handle the communication constraints related to the allocation of two ordering dependent tasks to different robots.

5.1 Bidding rule & Tree valuation

A bidding rule refers to which nodes a bidder can bid on simultaneously at each allocation round. Giving bidders a high freedom in choosing which tasks to bid on leads to improved solution quality but require significant amount of computation (for each additional node considered by the bidder, the valuation mechanism has to determine the local cost and possibly re-decompose it if it is abstract and of lower cost than the given decomposition). The time required for task valuation is a concern even for simple task auctions, as determining the local cost of a particular task or a set of task can involve *NP-hard* planning and scheduling problems. Bid valuation in a task tree is much more sophisticated than for a simple tasks and therefore has greater computational demands. Costs may be computed not only for individual tasks but for combinations of tasks.

In our system, when each robot receives the task tree to bid on, the bidding language allows bidders to bid on any set of nodes in the tree, despite of possible large communication and computation demand. To overcome this, we propose a valuation algorithm that on the one hand quickly estimates the cost and the re-decomposition of each node in the tree. The assumption is given a method to compute the cost of a task *individual planner*, the cost of a node is just the cost of all primitive descendant subtasks of that node. On the other hand, we propose a method of selecting which tasks to bid on, based on comparing approximated lower bounds for the tasks with their reserve price. The lower bound is computed by heuristic search on the navigation cost of the task.

5.2 Winner determination algorithm

The winner determination algorithm must first derive the optimal assignment of the task tree nodes to the bidders. It also has to generate synchronization communication tasks related to the precedence constraints between two tasks affected to two different robots. As we have seen, the communication is handled in three different ways. Whereas hard constraints are handled by the individual scheduler and the communication as utility is handled by a communication task generation process, the synchronization communication is directly managed inside the winner determination algorithm.

A synchronization is needed when bids on two ordered tasks t_1, t_2 come from different bidders r_1,r_2. If r_1 is unable to announce the end t_1 to r_2 so that the latter starts the execution of t_2, our algorithm generates the resolution request to concerned robots that will generate a synchronization act and send back the new plan with its associated cost. The resolution plan could either imply r_1 to go near r_2 or r_2 to go near r_1 or both r_1 and r_2 make the move. This is decided via a iterative negotiation process done by the *individual scheduler*.

Our winner determination algorithm is composed of the following two steps :

1. Via a dynamic programming algorithm, an optimal tree is computed and the corresponding assignment of robot to tree node is given. The algorithm clears the tree from the bottom to the top. At each level, it chooses the best bid for the node. If the robot r_i bid is chosen for the node n_j, all the previous allocation of descendant nodes of n_j are removed and these nodes go to r_i.

2. After the first step, an initial assignment is done but there are potential inconsistencies because no synchronization is able to synchronize two dependant tasks assigned to two different robots. The algorithm walks the tree and determines these inconsistencies. It sends resolution requests and waits for the resulting plans. Once all new bids are received, a new winner determination is locally done in the tree and the consistent assignment is found.

6 Scheduler

The role of the individual scheduler is to arrange in an optimal and consistent manner (make-span and plan's cost) different atomic tasks of the robots plan tree. Two data structures are used to represent this arrangement: a constrained timeline (communication constraints and end-location constraint) and an associated STN (Simple Temporal Network) for representing temporal ordering between different tasks. Given a list of tasks to perform, a robot must determine the most efficient total ordering in which to perform them while ensuring that any inter-task, inter-robot, communication and end-location constraints are met. A schedule may specify the start and completion times of the tasks, or simply the execution sequence.

The *marginal cost* of adding a plan needed by the *tree valuation* process is computed by the scheduler through an iterative algorithm that we designed to be fast but sub-optimal. The algorithm starts with an empty timeline with initial constraints. At each addition of a new plan tree, the algorithms divides this plan to temporal independent sub-plans (tree branches) and insert them *in an optimal manner* in the timeline. The final solution is not optimal because we do not consider every atomic task but task blocks. We believe that this is fast enough and good enough in terms of solution quality – which however remains to be demonstrated.

7 Conclusion

We presented in this paper our investigations on two aspects of a multi-robot planning based on distributed market-based task allocation. The first one concerns the explicit management of communications with range-limited and line-of-sight constraints. For that purpose, we designed a system capable of simple temporal and geometric reasoning and derived a new task allocation algorithm to handle communication in three possible ways: as hard rendez-vous constraint, as a task with associated utility and cost, and finally as a synchronization act needed in the execution of two dependent tasks by different robots.

The second investigation focuses on the use of complex task formalism, firstly to satisfy the demand of communication management in terms of plan semantics and secondly to handle more complex missions with better multi-robot plan quality. For that purpose, we have designed fast task allocation and scheduling algorithms without sacrificing the solution quality. No proof of the approach features are yet established: the solutions currently remain at the "on-going work" status, and our objectives is to terminate the implementation and evaluate it in the context of surveillance and tracking scenarios.

Acknowledgments This work has been partially supported by the DGA founded Action project (`http://action.onera.fr`).

References

[1] Tamio ARAI, Enrico PAGELLO, and Lynne PARKER. Editorial: Advances in multi robot systems. *IEEE Transactions on Robotics and Automation*, 18:655, 2002.

[2] Ronald C. ARKIN and Jonathan DIAZ. Line-of-sight constrained exploration for reactive multiagent robotic teams. In *in AMC 7th International Workshop on Advanced Motion Control*, page 455461, 2002.

[3] Nuzhet ATAY and O. Burçhan BAYAZIT. Emergent task allocation for mobile robots. In *Robotics: Science and Systems*, 2007.

[4] Sylvia BOTELHO and Rachid ALAMI. M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement. *IEEE International Conference on Robotics and Automation*, pages 1234–1239, 1999.

[5] Wolfram BURGARD, Mark MOORS, Cyrills STACHNISS, and Frank SCHNEIDER. Coordinated multi-robot exploration. *IEEE Transactions on Robotics*, 21:376–386, 2005.

[6] Uny Y. CAO, Alex FUKUNAGA, and Andrew KAHNG. Cooperative mobile robotics: Antecedents and directions. Technical Report 950049, UCLA Computer Science Department, March0, 1995.

[7] Imrich CHLAMTAC, Marco CONTI, and Jennifer LIU. Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks*, 1(1):13–64, 2003.

[8] Bernardine M. DIAS. *TraderBots: A New Paradigm for Robust and Efficient Multirobot Coordination in Dynamic Environments*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, January 2004.

[9] Arnaud DONIEC, Noury BOURAQADI, Michael DEFORT, Van Tuan LE, and Serge STINCKWICH. Distributed constraint reasoning applied to multi-robot exploration. In *ICTAI '09: Proceedings of the 2009 21st IEEE International Conference on Tools with Artificial Intelligence*, pages 159–166, Washington, DC, USA, 2009. IEEE Computer Society.

[10] Alessandro FARINELLI, Luca IOCCHI, and Daniele NARDI. Multi robot systems: A classification focused on coordination. *IEEE Transactions Systems, Man, Cybernnetics*, 34:2015, 2004.

[11] Brian GERKEY and Maja MATARIĆ. Sold!: Auction methods for multi-robot coordination. *ITRA*, 18(5):758–768, October 2002.

[12] Brian GERKEY and Maja MATARIĆ. A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, 9:939–954, 2004.

[13] Dani GOLDBERG, Vincent CICIRELLO, Bernardine DIAS, Reid SIMMONS, Stephen SMITH, Trey SMITH, and Anthony STENTZ. A distributed layered architecture for mobile robot coordination: Application to space exploration. *Proceedings of the 3rd International NASA Workshop on Planning and Scheduling for Space*, 2002.

[14] Andrew HOWARD, Maja MATARIĆ, and Sukhatme SUKHATME. An incremental self-deployment algorithm for mobile sensor networks. *Auton. Robots*, 13(2):113–126, 2002.

[15] Chris JONES, Dylan SHELL, Maja MATARIĆ, and Brian GERKEY. Principled approaches to the design of multi-robot systems. In *In Proc. of the Workshop on Networked Robotics, IEEE/RSJ IROS*, 2004.

[16] Nidhi KALRA, David FERGUSON, and Anthony STENTZ. A generalized framework for solving tightly-coupled multirobot planning problems. In *International Conference on Robotics and Automation (ICRA)*, April 2007.

[17] Thomas LEMAIRE, Rachid ALAMI, and Simon LACROIX. A distributed tasks allocation scheme in multi-uav context. In *ICRA*, April 2004.

[18] Douglas McKENZIE. Collaborative tasking of tightly constrained multi-robot missions. In *In Multi-Robot Systems: From Swarms to Intelligent Automata: Proceedings of the 2003 International Workshop on Multi-Robot Systems*, 2004.

[19] 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.

[20] Lynne PARKER. Alliance: An architecture for fault tolerant multi robot cooperation. *IEEE Transactions on Robotics And Automation*, 14:220, 1998.

[21] Lynne PARKER. Current research in multirobot systems. *Artificial Life and Robotics*, pages 1–5, 2003.

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

[23] Nicholas ROY and Gregory DUDEK. Collaborative robot exploration and rendezvous: Algorithms, performance bounds and observations. *Auton. Robots*, 11(2):117–136, 2001.

[24] S. SARIEL, T. R. BALCH, and N. ERDOGAN. Incremental multi-robot task selection for resource constrained and interrelated tasks. In *IROS*, pages 2314–2319, 2007.

[25] Jeff SCHNEIDER, David APFELBAUM, Drew BAGNELL, and Reid SIMMONS. Learning opportunity costs in multi-robot market based planners. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA*, 2005.

[26] Reid SMITH. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29:1104, 1981.

[27] Jose VAZQUEZ and Chris MALCOLM. Distributed multirobot exploration maintaining a mobile network. In *In Proceedings of Second IEEE International Conference on Intelligent Systems*, pages 113–118, 2004.

[28] Alan WAGNER and ARKIN Ronald C. Multi-robot communication-sensitive reconnaissance. In *ICRA*, pages 4674–4681, 2004.

[29] B. B. WERGER and MATARIĆ M. J. Broadcast of local eligibility for multi-target observation. pages 347–356. Springer-Verlag, 2000.

[30] Robert M. ZLOT and Anthony STENTZ. Complex task allocation for multiple robots. In *Proceedings of the International Conference on Robotics and Automation*, pages 1515 – 1522. IEEE, April 2005.